BIOINFORMATICS ORIGINAL PAPER Vol. 21 no. 22 2005, pages 4155–4161
doi:10.1093/bioinformatics/bti638

Gene expression

DNA microarray data imputation and significance analysis of differential expression

Rebecka Jörnsten[1,*], Hui-Yu Wang[2], William J. Welsh[3] and Ming Ouyang[3,*]

[1]Department of Statistics, Rutgers, the State University of New Jersey, New Brunswick, NJ 08903, USA, [2]9 Stoecker Road, Holmdel, NJ 07733, USA and [3]Department of Pharmacology, Robert Wood Johnson Medical School, and Informatics Institute, University of Medicine and Dentistry of New Jersey, Piscataway, NJ 08854, USA

Received on September 24, 2004; revised on August 2, 2005; accepted on August 17, 2005
Advance Access publication August 23, 2005

ABSTRACT

Motivation: Significance analysis of differential expression in DNA microarray data is an important task. Much of the current research is focused on developing improved tests and software tools. The task is difficult not only owing to the high dimensionality of the data (number of genes), but also because of the often non-negligible presence of missing values. There is thus a great need to reliably impute these missing values prior to the statistical analyses. Many imputation methods have been developed for DNA microarray data, but their impact on statistical analyses has not been well studied. In this work we examine how missing values and their imputation affect significance analysis of differential expression.

Results: We develop a new imputation method (LinCmb) that is superior to the widely used methods in terms of normalized root mean squared error. Its estimates are the convex combinations of the estimates of existing methods. We find that LinCmb adapts to the structure of the data: If the data are heterogeneous or if there are few missing values, LinCmb puts more weight on local imputation methods; if the data are homogeneous or if there are many missing values, LinCmb puts more weight on global imputation methods. Thus, LinCmb is a useful tool to understand the merits of different imputation methods. We also demonstrate that missing values affect significance analysis. Two datasets, different amounts of missing values, different imputation methods, the standard *t*-test and the regularized *t*-test and ANOVA are employed in the simulations. We conclude that good imputation alleviates the impact of missing values and should be an integral part of microarray data analysis. The most competitive methods are LinCmb, GMC and BPCA. Popular imputation schemes such as SVD, row mean, and KNN all exhibit high variance and poor performance. The regularized *t*-test is less affected by missing values than the standard *t*-test.

Availability: Matlab code is available on request from the authors.

Contact: rebecka@stat.rutgers.edu; ouyangmi@umdnj.edu

1 INTRODUCTION

The DNA microarray technology is a method for probing the expression of large numbers of genes simultaneously. Thousands of DNA probes are arranged in a 2D array, typically on glass slides. The total pool of mRNA from experimentally manipulated cells or tissues are used to generate cDNAs, which are labeled using fluorescent nucleotides. The labeled cDNAs are allowed to bind (hybridize) to the DNA probes on the slide. The intensity of the hybridized signal is related to the amount of mRNA that was originally present in the cells or tissues.

There are several sources of missing values. First, the spots on the slides are miniscule and they are packed very tightly. A tiny imperfection, a smudge or a speck of dust will corrupt the signals at a number of spots. After the array images are scanned and digitized, the problematic spots are manually flagged as missing. Second, there is background noise in the scanned image, and it is customary to subtract the background intensity from the spot intensity. For various technical reasons, such as bleed-over from neighboring spots and hybridization failures, the background intensity can be higher than that of the spot, and background subtraction produces negative expression levels. Those negative numbers are treated as missing. Some investigators use quality filtering: a spot is flagged and treated as missing if the spot intensity is less than, for example, 1.5-fold of the background.

Basically two types of microarray are in current use; they can be categorized by how the DNA probes are immobilized on the slide: the *in situ* synthesized Affymetrix GeneChips and the spotted cDNA (or oligonucleotide) microarrays. In a GeneChip (Lipshutz *et al*., 1999), 11–20 probe pairs are used to interrogate a gene. Although the probe-pair multiplicity is not intended to prevent missing values, it virtually precludes them in GeneChip data. Spotted cDNA microarrays (Brown and Botstein, 1999) usually allocate one spot per gene. Some do have double to quadruple spots for a gene, but they are the exception rather than the norm. The loss at a spot usually translates to the loss of information for a gene. Thus, the present work is concerned with imputation of missing values in spotted cDNA microarray data. Note that what we mean by *missing* is different from the *absent* flag in GeneChip data. The *present* and *absent* flags generated by proprietary Affymetrix software indicate whether the targets are detectable, whereas by *missing* we mean the data are corrupted, and it is infeasible to determine whether or at what quantities the targets are present.

Microarray data can be represented as a matrix A. The rows correspond to the genes, the columns correspond to the experiments and the entry $A_{i,j}$ is the expression level of gene i in experiment j. Most spotted microarray experiments use the two-dye protocol: The control is labeled with Cy3 (green), the treatment is labeled with Cy5 (red) and the data in the matrix are the log-ratios of treatment versus control. A simple imputation is to fill the missing values with

*To whom correspondence should be addressed.

zeros, effectively declaring that the treatment does not alter gene expression. Another simple method uses the row means (ROW) for imputation. There are numerous non-trivial imputation methods: Troyanskaya *et al*. (2001) studied K nearest neighbor imputation (KNN) and singular value decomposition based imputation (SVD); Oba *et al*. (2003) described a method based on Bayesian principal components analysis (BPCA); Zhou *et al*. (2003) used Bayesian variable selection and both linear and non-linear regression for imputation; Bø *et al*. (2004) described a method based on the least squares principle; Kim *et al*. (2005) studied local least squares imputation. For time-series data, Bar-Joseph *et al*. (2003) described a model-based spline fitting method and Schliep *et al*. (2003) used hidden Markov models for imputation. All these papers focused on how close the imputed values were to the true values, in terms of normalized root mean squared error (RMSE).

We take the view that the goal of imputation is to improve the results of subsequent data analysis, such as cluster analysis (Eisen *et al*., 1998) and significance analysis of differential expression (Cui and Churchill, 2003). Therefore, the focus of our work is the impact of missing values and their imputation on data analysis. Previously, we (Ouyang *et al*., 2004) described an imputation method based on Gaussian mixture clustering (GMC) and model averaging that has smaller RMSE than KNN and SVD, and improves the subsequent cluster analysis by reducing the number of misclustered genes.

In the present study, we describe a new method (LinCmb) whose estimates are the convex combinations of the estimates by ROW, KNN, SVD, BPCA and GMC. The rationale is that each method takes a specific approach to data imputation, and is therefore associated with a particular type of systematic error. By combining the different estimates, some of the errors may cancel out, and we can borrow strength across methods to adapt to the data structure.

Using cDNA microarray data from a study on human liver and liver cancer (Chen *et al*., 2002) and a drug study (Pan *et al*., 2004) for simulations, we demonstrate that LinCmb has smaller RMSE than all the constituent methods. Furthermore, we use the full data with no missing values to construct 'gold standards' of differentially expressed genes by the standard *t*-test, the regularized *t*-test (Baldi and Long, 2001) or ANOVA. We then compare the false positive rate (FPR) of the analyses in the presence of missing values. We find that the standard *t*-test is less robust and is affected by missing values to a larger extent than the regularized *t*-test. If the amount of missing values is small (1%), imputation does not significantly improve the results and may in fact increase FPR. However, if there are many missing values (4% or 7%), good imputation reduces FPR. LinCmb helps shed light on the differences of performance among the various imputation methods. LinCmb, GMC and BPCA are all competitive in terms of FPR performance, and among them GMC is the simplest and easiest to compute. Widely used methods, such as ROW, SVD and KNN, perform poorly in comparison. Our conclusion is that imputation should be an integral part of microarray data analysis.

2 DATA AND METHODS

2.1 Microarray data

The human liver and liver cancer microarray data (Chen *et al*., 2002) were downloaded from Stanford Microarray Database (Gollub *et al*., 2003). Each of the 207 arrays probes 23 000 transcripts. The smallest percent of missing values in an array is 3.49%, the largest 54.1% and the median 26.5%. Out of the 23 000 transcripts, only 285 have no missing values. Many of the missing values correspond to genes that are not expressed in liver or liver cancer. The proportion of missing values that arise from blemishes on the chips is <5% for most of the arrays. We selected 20 arrays of liver samples and 20 arrays of liver cancer samples with the fewest missing values for further analysis. The resulting complete data consist of 6511 probes on 40 arrays. The drug data (Pan *et al*., 2004) probe gene expression in spinal cord in response to injury and treatment by several anti-inflammatory drugs. Each array probes 4967 transcripts. There are seven experimental conditions (two baselines and five drugs). Three replicates are available for each experimental condition. We extracted a subset of the data with no missing values consisting of 1664 probes on 21 arrays ($7 * 3$).

To emulate the random nature of blemishes on the chips, we adapt the model of missing at random: in the simulations, 1, 4 and 7% of the data values are randomly and independently marked as missing, their values are imputed and the imputed values are compared with the true values. The 1–7% range is based on realistic expectation of quality control. If a chip has >10% missing entries owing to blemishes, the remaining data may not be reliable for meaningful analysis.

2.2 Imputation methods

Microarray data are represented as a matrix. The rows correspond to the genes, and the columns correspond to the samples. ROW uses the row means as the estimates of the missing values. KNN and SVD were described by Troyanskaya *et al*. (2001), and C++ code of KNN is available; we implement both methods in Matlab. BPCA was described by Oba *et al*. (2003), and Matlab code is available. GMC (Ouyang *et al*., 2004) takes the approach of model averaging. The microarray data are clustered into $1, 2, \ldots, T$-component Gaussian mixtures (T is usually <10) by the classification expectation–maximization algorithm (Banfield and Raftery, 1993); then the missing values are estimated by the expectation–maximization algorithm (Dempster *et al*., 1977); for each missing value, the estimate by GMC is the simple average of the T estimates.

The new method LinCmb takes a regression approach called model stacking (Hastie *et al*., 2001). Let M be the true values of the missing data, let R, K, S and B be the estimates of M by ROW, KNN, SVD and BPCA, respectively, and let $G_1, \ldots, G_5$ be the estimates of M by Gaussian mixtures of $1, \ldots, 5$ components. Least-squares regression is used to determine the constants r, k, s, b and $g_1, \ldots, g_5$ in

$$M' = rR + kK + sS + bB + \sum_{i=1}^{5} g_i G_i, \tag{1}$$

subject to the constraints

$$r, k, s, b, g_1, \ldots, g_5 \geqslant 0, \tag{2}$$

and

$$r + k + s + b + \sum_{i=1}^{5} g_i = 1. \tag{3}$$

The values of $r, k, s, b, g_1, \ldots, g_5$ are estimated as follows. LinCmb is given a matrix with missing values. Let p be the proportion of missing data. LinCmb first uses KNN to estimate the missing values and obtains a completed matrix A. Then LinCmb performs a loop of 30 iterations. In each iteration, LinCmb uses the missing probability $p/(1-p)$ to generate 'fake' missing entries in A whose true values, $\tilde{M}$, are known to LinCmb. If a fake-missing entry coincides with a real missing value, it is *not* treated as fake missing. The expected fake-missing rate among the originally non-missing entries is $(p/(1-p)) \cdot (1-p) = p$. The constituent methods are used to estimate the fake-missing entries. Since LinCmb knows $\tilde{M}$, it can then perform the least-squares regression to obtain a vector $(r, k, s, b, g_1, \ldots, g_5)$. Let $(\bar{r}, \bar{k}, \bar{s}, \bar{b}, \bar{g_1}, \ldots, \bar{g_5})$ be the mean of these 30 vectors from the loop.

The LinCmb imputation is defined as

$$\bar{r}R + \bar{k}K + \bar{s}S + \bar{b}B + \sum_{i=1}^{5} \bar{g}_i G_i. \quad (4)$$

The parameters of KNN and SVD can be set via procedures similar to the one above. However, for KNN we simply set K, the number of nearest neighbors, at 16 for all datasets and all missing probabilities. The optimal values of K do vary somewhat depending on data and missing probabilities, and they are in the range from 10 to 20. The difference in imputation accuracy between the optimal values and 16 is very small. For SVD we set the number of singular vectors at the optimal value (in this case at 2).

Let M' be an estimate of M. The accuracy of M' is measured by normalized RMSE:

$$\text{RMSE} = \sqrt{\frac{\text{mean}\{(M-M')^2\}}{\text{mean}\{M^2\}}}, \quad (5)$$

where M^2, for example, is componentwise. Oba *et al*. (2003) used a different definition: $\text{RMSE} = \sqrt{\text{mean}\{(M-M')^2\}/\text{variance}\{M\}}$, but when microarray data as here are normalized (Quackenbush, 2002) before imputation, the expected value of mean$\{M\}$ is zero. Thus these two RMSEs should be very close.

2.3 Statistical tests

To cope with the large variations in microarray data, the regularized t-test (Baldi and Long, 2001) calculates the weighted average of the gene-specific variance and the mean variance of the gene's 'neighborhood'. The mean neighborhood variance appears in the prior of a Bayesian formulation. We use a neighborhood of 101 genes consisting of 50 each with mean expression levels immediately above and below the gene under consideration, and the gene itself. Let s^2 be the gene-specific variance, let σ_0^2 be the mean neighborhood variance, let n be the number of replicates and let ν_0 be a positive integer. Then the Bayesian adjusted variance of the gene is

$$\sigma^2 = \frac{\nu_0\sigma_0^2 + (n-1)s^2}{\nu_0 + n - 2}. \quad (6)$$

The adjusted variance σ^2 is used in a gene-specific t-test with $\nu_0 + n - 2$ degrees of freedom. Baldi and Long (2001) suggested that $\nu_0 + n$ be 10 when n is very small, but there was no guideline with respect to missing values. We use the following way to accommodate them. When the number of replicates n' of a gene is less than n owing to missing values, a ν_0' such that $\nu_0' + n'$ is equal to $\nu_0 + n$ is used to calculate σ^2. Another issue is that the value 10 for $\nu_0 + n$ is meant for experiments with very few replicates of samples from controlled genetic compositions, whereas the liver and liver cancer data are from 20 independent samples. Thus, we use 25 for $\nu_0 + n$ here, so that there is a (5:20) Bayesian modulation in σ^2.

The regularized t-test provides a nominal P-value of differential expression. We then use the Benjamini and Hochberg (1995) adjustment to control the false discovery rate in multiple testing. We designate 0.001 as the thresholds on the adjusted P-values for significance of differential expression in the liver and liver cancer data, resulting in 1452 significant genes. These genes are treated as the 'gold standard'. For the purpose of comparison, we also use the standard t-test to calculate nominal P-values and then apply the BH adjustment. The number of significant genes in the gold standard of the standard t-test is 1416. For the drug data, we apply ANOVA and the BH adjustment. The number of significant genes in the gold standard is 378 at the adjusted P-value of 0.05.

2.4 The simulation scheme

The liver and liver cancer data are two 6511×20 matrices. The 6511×40 entries are randomly and independently marked as missing with probabilities 0.01, 0.04 and 0.07.

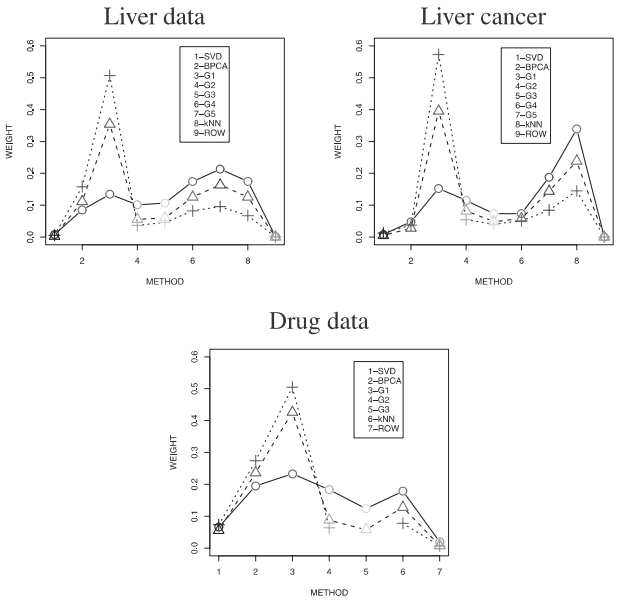

Fig. 1. Mean method weights assigned by LinCmb for missing probabilities 0.01 (open circle, solid line), 0.04 (triangle, dashed line) and 0.07 (cross, dotted line).

First, the data with missing values are subjected to the regularized t-test where $\nu_0' + n'$ is set at 25 regardless of how many missing values a gene may have. The genes are sorted by their BH adjusted P-values, and the sorted list is compared with the gold standard. We compare two sets of gene lists: one at false negative rate (FNR) 0%, and the other at FNR 5%. We calculate the false positive rate (FPR) of the regularized t-test in the presence of missing values at these two FNR levels:

$$\text{FPR} = \frac{\text{false positive}}{\text{true positive} + \text{false positive}}. \quad (7)$$

FPRs of the standard t-test are also calculated, where missing values result in reduced degrees of freedom.

Second, the missing values in the data are imputed by ROW, KNN, SVD, BPCA, $\text{GMC}_1, \ldots, \text{GMC}_5$ and LinCmb. The RMSE of each method is calculated. The data with imputed values are subjected to both the regularized t-test where $\nu_0 + n$ is 25, and the standard t-test where the degrees of freedom are 19. We then calculate FPRs for each of the imputation methods.

The simulation is repeated 200 times for each missing probability. A similar simulation is performed on the drug data with ANOVA.

3 RESULTS

3.1 Parameters for GMC and LinCmb

Gaussian mixtures require the number of components to be fixed; we used mixtures of $1, \ldots, 5$ components for the liver and liver cancer data. With more mixtures the components often contain too few data points to reliably determine their probability density functions. For the drug data, we were only able to use up to three components (two for missing probability 0.07).

LinCmb is the convex combination of the constituent methods. Figure 1 depicts the means of these weights from 200 randomized runs on liver, liver cancer and drug data. As the number of missing values increases, the weights of local methods (e.g. KNN and GMC_5) decrease, and those of global methods (e.g. BPCA and GMC_1) increase. For the heterogeneous datasets (liver cancer

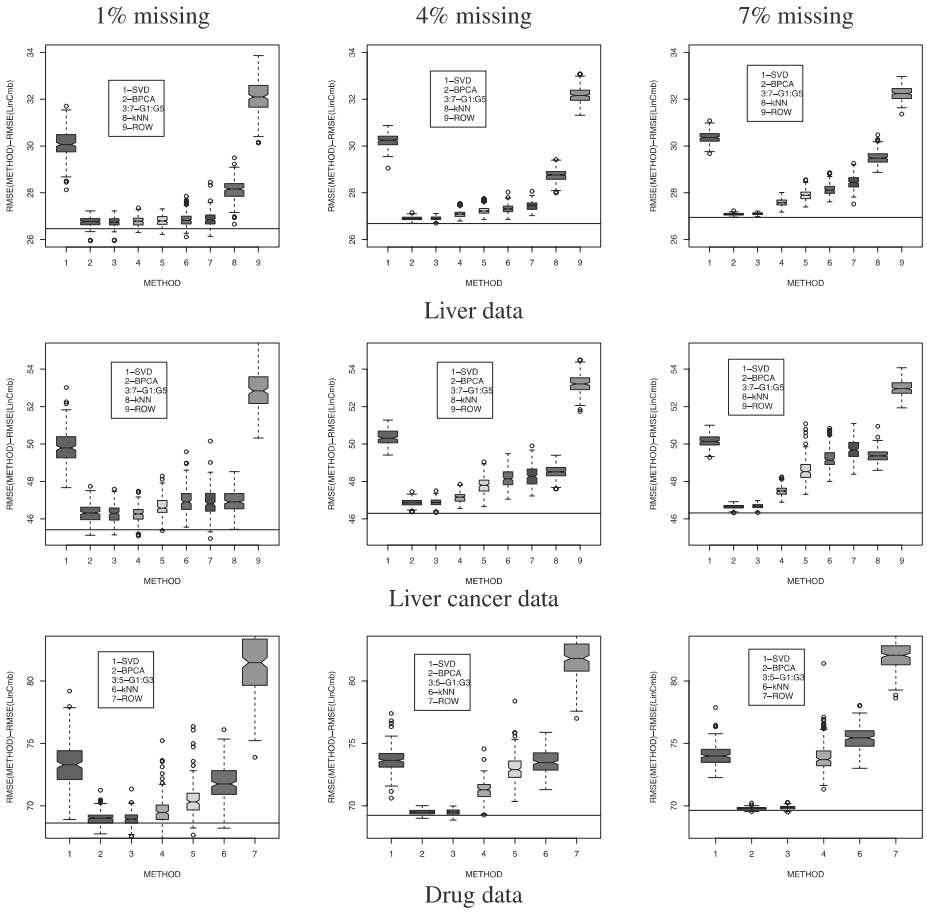

Fig. 2. Imputation RMSE differences for missing probabilities 0.01 (left column), 0.04 (middle column) and 0.07 (right column) for the liver (first row), liver cancer (middle row) and drug (third row) data. The boxplots show pairwise differences [RMSE(method)−RMSE(LinCmb)] (multiplied by 100) across 200 randomized runs. The median RMSE(LinCmb) is added to all entries and is shown with a horizontal line. Figures in each row are drawn to the same scale.

and drug), larger weights are assigned to local methods when local information is available (1% missing). To reduce clutter in the figure only mean results are shown. The results are highly reproducible across randomized runs.

3.2 Root mean squared error

Imputation accuracy is measured by normalized root mean squared error [RMSE; Equation (5)]. Figure 2 depicts boxplots of RMSE (200 randomized runs) for liver, liver cancer and drug data. To facilitate the comparison, we show the pairwise difference of RMSE(LinCmb) and the RMSE of the constituent methods, i.e. RMSE(method)−RMSE(LinCmb). The median of RMSE(LinCmb) is added to all entries to make the multiple figures directly comparable. We find that LinCmb has lower RMSE than all of the constituent methods. We test the significance of the RMSE improvement and obtain P-values $<10^{-12}$ for all pairwise comparisons, all datasets and all missing probabilities.

There are notable connections between Figures 1 and 2. First, ROW and SVD have the worst RMSE, and thus the smallest weights in the convex combination. Second, among the GMC_is, GMC_1 has the best RMSE and the largest weight. Third, BPCA and GMC_1 have about the same RMSE, but GMC_1 is assigned larger weights. The phenomenon is most pronounced for the liver cancer data with missing probability 0.07; BPCA has the best RMSE, and yet its contribution to LinCmb is very limited. BPCA and GMC_1 predictions are highly correlated; however, upon close examination of the simulation data, we find that BPCA predictions are more variable. Thus, in the least-squares regression, larger weights are assigned to GMC_1.

To explain how the LinCmb weights are distributed we classify imputation methods as local and global. KNN is clearly local; GMC_i with $i > 1$ are local; the more Gaussian components that are fit to the data, the more local the method is. The global methods are SVD, its robust counterpart BPCA and GMC_1. ROW is local if the row mean is taken across columns of the same experimental condition (as for the liver data), and can loosely be thought of as global if imputation is done across all columns (as for the drug data, with few replicate samples). As the missing probability increases, the weights shift from local to global methods (Fig. 1). When there are few missing values, local methods have high weights, because there is sufficient local information for imputation. When there are many missing values, global methods dominate, because the availability of reliable local information is limited. In other words, LinCmb adapts to the amounts of missing values in the data, and this

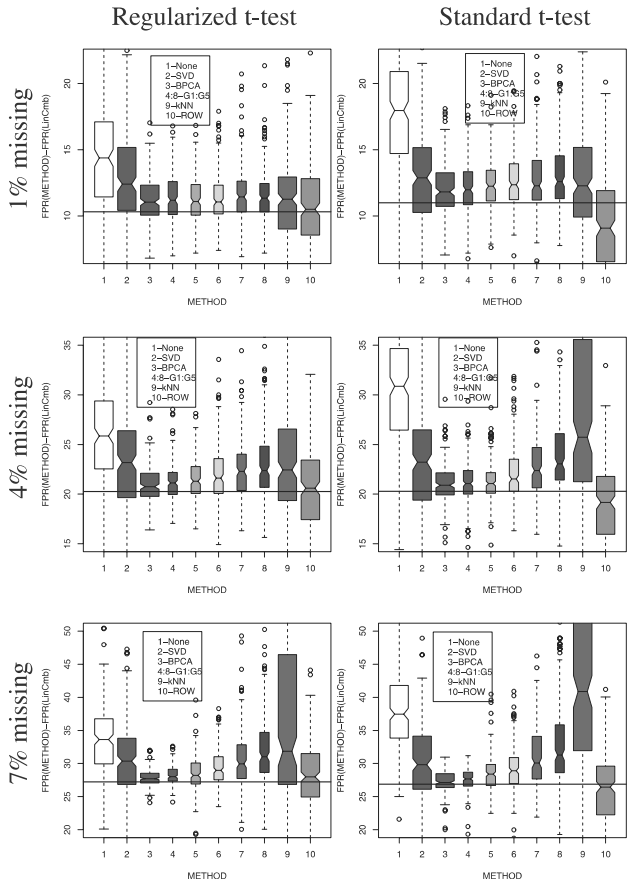

Fig. 3. FPR (multiplied by 100), at FNR = 0%, for the liver data. The boxplots show FPR(method)−FPR(LinCmb) across 200 randomized runs; the median of FPR(LinCmb) is added to each entry and marked with a horizontal line.

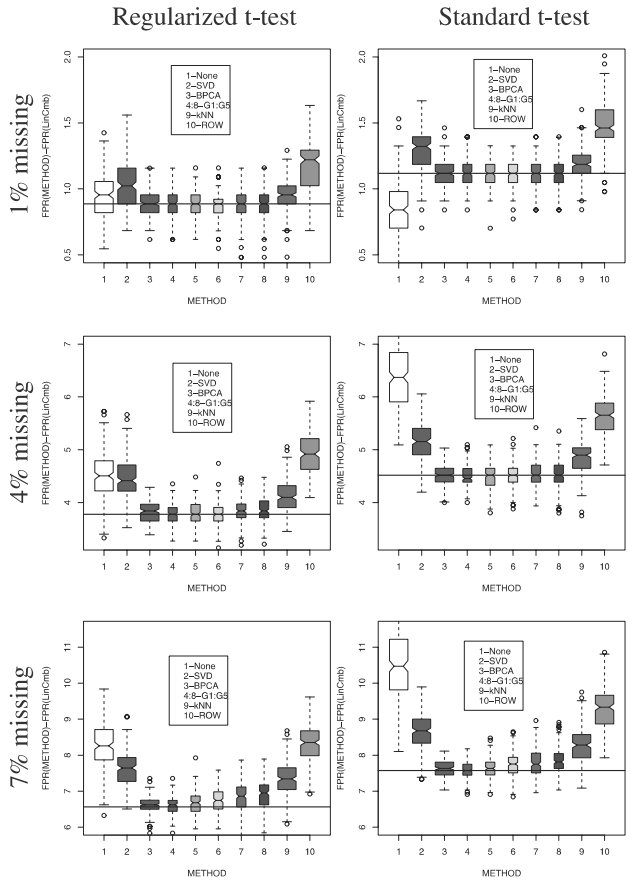

Fig. 4. FPR (multiplied by 100), at FNR = 5%, for the liver data. See Figure 3 for explanations.

explains its excellent RMSE. Note that local methods show good RMSE performance when few missing values are present, but their performance quickly deteriorates as missingness increases (Fig. 2).

3.3 False positive rates

Missing values are replaced by imputed values, and imputed data are analyzed by the two *t*-tests or ANOVA, as if there are no missing values at all. We also process the data without imputation (referred to as 'None' in the figures and tables). This involves reduced degrees of freedom for the standard *t*-test and ANOVA, and an increase of ν_0' to maintain $\nu_0' + n' = 25$ for the regularized *t*-test. Some examples of FPR/FNR values for the liver data are shown in Table 1. Genes are selected at adjusted *P*-value of 0.001, and compared with the gold standard. None gives the lowest FPR and highest FNR; an expected result that reflects the loss of power associated with a reduced data size owing to missingness. ROW has the next highest FPR and lowest FNR. ROW reduces the variance within experimental conditions, and if borderline (non)significant genes contain missing values, ROW will almost guarantee their significance. However, compared with other methods, ROW's loss in FPR is 1–2%, whereas its gain in FNR is an order of smaller magnitude. KNN has both high FPR and FNR, suggesting that it is not competitive despite its popularity (similar results were obtained for SVD). BPCA and LinCmb appear to control both FPR and FNR.

Since the number of significant genes, FPR and FNR all vary across methods, direct comparison of methods is difficult. We therefore compare methods at two fixed FNR levels: 0% and 5% (Section 2.4).

Figures 3, 4 and 5 show the FPR when data are imputed by various methods. We depict differences in FPR between LinCmb and its constituent methods (similar to Fig. 2): FPR(method)−FPR(LinCmb); the median of FPR(LinCmb) is added to all entries to make the figures directly comparable. Figure 3 shows the FPR differences for regularized *t*-test and standard *t*-test for the liver data set at FNR = 0%; Figure 4 shows the results at FNR = 5%. A comparison between the two figures shows that the regularized *t*-test has a smaller FPR at FNR = 5% than the standard *t*-test, whereas the FPRs are comparable at FNR = 0%. At FNR = 0%, ROW is the most competitive method in terms of FPR, with LinCmb as a close competitor. Excluding ROW, LinCmb is significantly better than all other methods when 1–4% data are missing, and significantly better or comparable to ROW at 7% missing. ROW performance deteriorates drastically at FNR = 5%. 5%. In fact, the relative FPR results at FNR = 5% mimic the RMSE results (Fig. 2). If the amount of missing values is small (1%), imputation does not greatly improve the results compared with no imputation (Fig. 4, first row). However, when there are

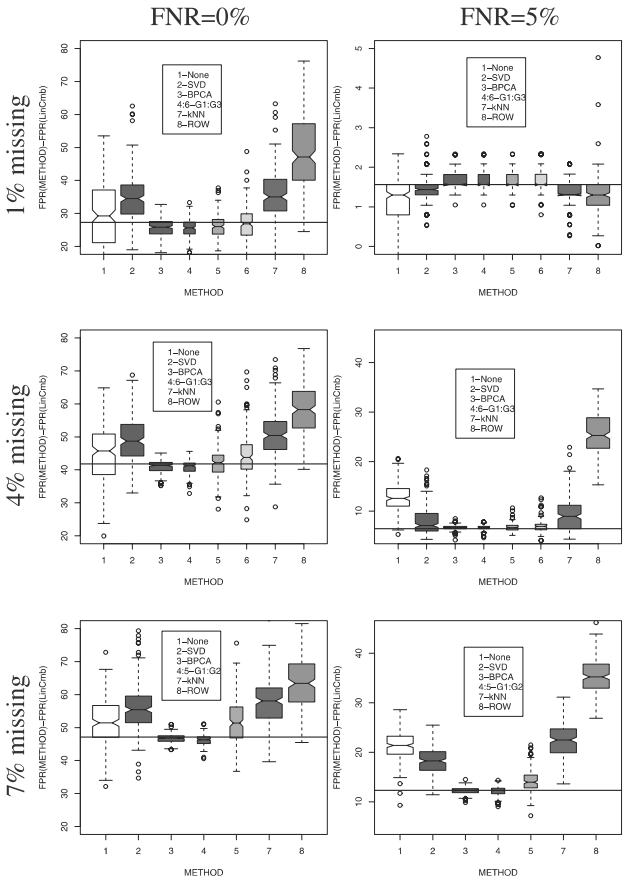

Fig. 5. FPR (multiplied by 100), at FNR = 0% (left column) and FNR = 5% (right column) of significance analysis for the drug dataset.

many missing values (4 and 7%), good imputation reduces FPR. When 1% are missing, None dominates all other imputation schemes if the standard *t*-test is used. With the regularized *t*-test None, SVD, KNN and ROW are all significantly worse than the rest (pairwise comparisons). At 4% missing, for both tests, None, SVD, KNN and ROW are all significantly worse than the other methods. At 7% missing LinCmb and GMC are significantly better than the rest.

Combining the results in Table 1, and Figures 3 and 4, we conclude that, for the liver data, LinCmb is consistently competitive at both FNR levels, especially when there are many missing values. BPCA and GMC_1 are competitive at FNR = 5%, though BPCA performance deteriorates at 4–7% missing (significantly worse than LinCmb). BPCA and GMC_1 performances at FNR = 0% are significantly worse than LinCmb.

Figure 5 shows the results on the drug data, which have few replicates per condition. Thus ROW imputes across multiple conditions and is more of a global method. ROW, KNN and SVD performance is consistently poor. With only 1% missing, None has one of the lowest FPR. With 4 and 7% missing, we see that at FNR = 0% (left column of Fig. 5) BPCA and GMC_1 are the most competitive, closely followed by LinCmb. At FNR = 5% and 1% missing, the FPR are all low (within four genes of each other). At 4% LinCmb is significantly better than the rest, and at 7% LinCmb, BPCA and GMC_1 are all comparable.

Table 1. Comparison of significant gene lists to the gold standard

	0.01	0.04	0.07
Standard *t*-test			
None	2.01/2.34	3.20/9.57	3.97/17.15
ROW	3.91/0.55	9.40/1.37	14.03/1.64
KNN	3.50/0.56	8.36/1.59	12.37/2.15
BPCA	3.44/0.51	8.02/1.44	11.80/1.91
LinCmb	3.44/0.48	8.01/1.42	11.70/1.91
Regularized *t*-test			
None	1.57/1.68	4.77/4.98	6.96/7.58
ROW	2.57/0.81	8.45/1.64	13.24/1.85
KNN	2.15/0.79	7.27/1.79	11.45/2.23
BPCA	2.08/0.76	6.92/1.66	10.82/2.01
LinCmb	2.09/0.75	6.86/1.66	10.73/2.02

Genes selected at FDR 0.001. FPR/FNR mean values (200 runs) for the liver data.

4 DISCUSSION

The liver and liver cancer microarray data provide an interesting contrast in imputation difficulty. Using the best methods available to us, imputation RMSE of liver data is <0.27, whereas that of liver cancer data is >0.46. This difference is probably because of the molecular homogeneity in the liver samples and the lack thereof in the cancer samples. The drug data are noisy with few replicates, and thus they are very difficult to impute.

When RMSE is considered (Fig. 2), LinCmb is the best method, followed by BPCA (Oba *et al*., 2003) and GMC_i (Ouyang *et al*., 2004). Popular methods such as KNN, SVD (Troyanskaya *et al*., 2001), and ROW all perform poorly in comparison. LinCmb is a convex combination of the other methods and uses both local and global information. The more missing data are present, the more weights are put on the global models, and vice versa. Prior to using the convex combination, we studied the simple method average, and the results were inferior to those presented here (data not shown). If we relax the convex constraint, the results are slightly better (data not shown); however, the weights of the unconstrained linear combination are difficult to interpret as some of them are negative. We also used median models instead of mean models from the interior iterations of LinCmb with insignificant changes to the results.

BPCA can be construed as a regularized version of SVD. The regularization improved the performance of the method drastically. This relationship is similar to the one between the regularized *t*-test and the standard *t*-test. In both cases, the Bayesian components help to cope with the large variations inherent in microarray data. In general, the regularized *t*-test (Baldi and Long, 2001) has smaller FPR than the standard *t*-test (Fig. 4).

LinCmb, GMC_1 and BPCA are all competitive in terms of FPR, with LinCmb having a slight edge when there are many missing values. If the standard *t*-test is used, and if few values are missing, not imputing may in fact lower the FPR. Future work will center on extending LinCmb to locally adapt to the rate of missingness. The decision on whether or not, or how to impute a missing value on a gene-by-gene basis will depend on the amount of missing values for that gene, and the amount of missing values in the neighborhood of that gene. We stress that the performance of widely-used methods, such as ROW, KNN and SVD, is very poor. Superior performance

in terms of RMSE does not guarantee a superior performance in terms of FPR. Thus the effect on significance analysis needs to be taken into account when comparing and developing imputation methods. Our results suggest that the effect of imputation on significance analysis is of 'second order', compared with the first order effect on RMSE. Still, the effect of imputation on significance analysis is notable, and using a robust and well-chosen imputation strategy is highly advisable.

The focus of this work was on understanding the impact of missing values on significance analysis, delineating the relative merits of local and global imputation approaches for different data, and comparing various imputation methods directly. In its present implementation, LinCmb is not computationally competitive with simple schemes such as GMC. When imputing the liver and liver cancer data with 7% missing values (Matlab scripts, Linux, Pentium 2.8 GHz, 1G RAM), the running time is, ROW 1 s, KNN 36 s, SVD 10 s, BPCA 13 min, and 16 and 60 s for GMC_1 and GMC_5, respectively. For the same task, LinCmb takes 8.5 h because it calls all the constituent methods 30 times in a loop to estimate its parameters (there was no significant improvement if we increased the number of iterations to 100). However, computational complexity is not as critical a factor as accuracy for imputation (Sehgal *et al*., 2005); it can argued that 8.5 h is negligible when compared with the time and effort that scientists put into microarray experiments and subsequent analysis. To use LinCmb in practice, we recommend removing BPCA from model stacking. Without BPCA, the running time of LinCmb is comparable to that of BPCA, and since the weight assigned to BPCA in model stacking is small, the performance is not significantly affected (data not shown).

ACKNOWLEDGEMENTS

M.O. is partially supported by USDA grant 2003-05414. R.J. is supported by NSF grant DMS0306360. M.O. and W.J.W. are partially supported by the NIH-NLM for an Integrated Advanced Information Management Systems (IAIMS) grant G08 LM06230-03A1 and the New Jersey Commission on Higher Education for a High-Technology Workforce Excellence grant 801020-09.

Conflict of Interest: none declared.

REFERENCES

Baldi,P. and Long,A. (2001) A Bayesian framework for the analysis of microarray expression data: regularized *t*-test and statistical inferences of gene changes. *Bioinformatics*, **17**, 509–519.

Banfield,J.D. and Raftery,A.E. (1993) Model-based Gaussian and non-Gaussian clustering. *Biometrics*, **49**, 803–821.

Bar-Joseph,Z. *et al*. (2003) Continuous representations of time-series gene expression data. *J. Comput. Biol*., **10**, 341–356.

Benjamini,Y. and Hochberg,Y. (1995) Controlling the false discovery rate: a pratical and powerful approach to multiple testing. *J. R. Stat. Soc. B*., **57**, 289–300.

Bø,T. *et al*. (2004) LSimpute: accurate estimation of missing values in microarray data with least squares methods. *Nucleic Acids Res*., **32**, e34.

Brown,P. and Botstein,D. (1999) Exploring the new world of the genome with DNA microarrays. *Nat. Genet*., **21**, 33–37.

Chen,X. *et al*. (2002) Gene expression patterns in human liver cancers. *Mol. Biol. Cell*., **13**, 1929–1939.

Cui,X. and Churchill,G.A. (2003) Statistical tests for differential expression in cDNA microarray experiments. *Genome Biol*., **4**, 210.

Dempster,A.P. *et al*. (1977) Maximum likelihood from incomplete data via the EM algorithm (with discussion). *J. R. Stat. Soc. B*, **39**, 1–38.

Eisen,M. *et al*. (1998) Cluster analysis and display of genome-wide expression patterns. *Proc. Natl Acad. Sci. USA*, **95**, 14863–14868.

Gollub,J. *et al*. (2003) The Stanford Microarray Database: data access and quality assessment tools. *Nucleic Acids Res*., **31**, 94–96.

Hastie,T., Tibshirani,R. and Friedman,J. (2001) *The elements of statistical learning*, Springer-Verlag, NY.

Kim,H. *et al*. (2005) Missing value estimation for DNA microarray gene expression data: local least squares imputation. *Bioinformatics*, **21**, 187–198.

Lipshutz,R. *et al*. (1999) High density synthetic oligonucleotide arrays. *Nat. Genet*., **21**, 20–24.

Oba,S. *et al*. (2003) A Bayesian missing value estimation method for gene expression profile data. *Bioinformatics*, **19**, 2088–2096.

Ouyang,M. *et al*. (2004) Gaussian mixture clustering and imputation of microarray data. *Bioinformatics*, **20**, 917–923.

Pan,J.Z. *et al*. (2004) Screening anti-inflammatory compounds in injured spinal cord with microarrays: a comparison of bioinformatics analysis approaches. *Physiol. Genomics*, **17**, 201–214.

Quackenbush,J. (2002) Microarray data normalization and transformation. *Nat. Genet*., **32**, 496–501.

Schliep,A. *et al*. (2003) Using hidden Markov models to analyze gene expression time course data. *Bioinformatics*, **19**, i255–i263.

Sehgal,M.S. *et al*. (2005) Collateral missing value imputation: a new robust missing value estimation algorithm for microarray data. *Bioinformatics*, **21**, 2417–2423.

Troyanskaya,O. *et al*. (2001) Missing value estimation methods for DNA microarrays. *Bioinformatics*, **17**, 520–525.

Zhou,X. *et al*. (2003) Missing-value estimation using linear and non-linear regression with Bayesian gene selection. *Bioinformatics*, **19**, 2302–2307.