Stat Biosci
DOI 10.1007/s12561-013-9081-1

A Two-Stage Procedure for the Removal of Batch Effects in Microarray Studies

Marco Giordan

Received: 10 April 2012 / Accepted: 29 January 2013
© International Chinese Statistical Association 2013

Abstract The presence of different batches is routinely observed in microarray studies and is well known that non-biological variability potentially confounding biological differences is commonly related to such batches. The removal of these undesired effects for a non-biased inference is often accomplished either with normalization methods that do not take into account all the available information, or with models that rely on strong parametric assumptions. We have developed a new method for the batch effects removal, named *ber*, which is based on a two-stage procedure for the estimation of location and scale parameters. Batch effects and biological differences are estimated using a regression approach and bagging, therefore mild distributional assumptions are required. We have compared *ber* with other commonly employed methods and we have shown that *ber* can bring to a higher power in detecting differentially expressed genes. The application of *ber* to a real microarray study led to interpretable biological results. The method is implemented in the R package ber, available through CRAN repository.

Keywords High dimensional data · Normalization · Gene expression profiling · Bagging

1 Introduction

Systematic variations in microarray data are commonly observed when sets of chips belong to different sources or are assayed under different settings. Broadly speaking, we can define a batch as a set of chips generated under similar conditions. The deviations from the biological signal of interest arising from these different conditions are known as *batch effects*. While it is important to aggregate data from many batches

M. Giordan (✉)
Department for Woman and Child's Health, University of Padua, Via Giustiniani 3, Padova, Italy
e-mail: giordan@stat.unipd.it

Published online: 05 February 2013

to get more reliable and powerful analyses, studies that ignore the batch effects can be biased. For example, in the comparison of two or more groups batch effects can easily generate false positives and false negatives, or produce misclassification errors. Zilliox and Irizarry [23] and McCall et al. [15] considered the problem of the batch effects for the determination of the genes expressed or unexpressed in a specific tissue or cell type.

There are many approaches to handle batch effects. In a straightforward manner the batches can be considered as a covariate in a standard linear model. This strategy, for example, is a suggestion in the package limma (available through bioconductor). Usually in fact when biological or clinical covariates are available it is of interest to estimate their influence on the main aspect being investigated. However, batch effects are undesired non-biological experimental effects that should not exist. For a non-biased inference many proposals suggest to estimate and remove them.

The methods for the removal of the batch effects can be classified in two main groups: methods working on normalized data and methods embedding the removal of the batch effects in the normalization process.

Many currently employed normalization methods such as RMA [9] or VSN [8] do not take into account that data are gathered together from different batches. The following approaches are applied on data normalized with such methods with the purpose to eliminate the batch effects. *Mean centering* is a common method to remove batch effects after normalization. For each gene and each subject the means of the expression levels in the batches are subtracted from the normalized values. This method is implemented in the pamr package (available through CRAN, www.r-project.org). In dchip software (http://biosun1.harvard.edu/complab/dchip) the normalized values in each batch are mean centered and then scaled to obtain standard deviations equal to one (*standardization*). Benito et al. [2] used a distance weighted discrimination method (*DWD*) to remove batch effects. Johnson et al. [11] developed an Empirical Bayes method (*combat*) for removing location and scale batch effects (see the bioconductor package sva). For merging two or more gene expression studies [19] proposed the use of a block linear model and of a procedure based on clustering (*XPN*). When reference samples are available ratio-based methods can be developed (see [13]).

In the second group of methods estimation and removal of batch effects are embedded into the normalization procedure. McCall et al. [14] proposed *fRMA*, an algorithm that allows to combine data for downstream analysis. Their method is based on a reference distribution and a model that uses random effects to explain the variability in probe effects across batches. Mecham et al. [16] proposed a general normalization framework where all the variables available for the study are employed. They developed *SNM*, an iterative algorithm for the simultaneous fitting of biological and study-specific adjustment variables.

The above mentioned methods show great differences in the adopted assumptions and in the amount of employed information. *Mean centering*, *standardization* and *DWD* do not assume a particular shape for the distribution of the gene expression levels, however, they use only the information about the batches and they are not able to use other information from biological covariates. *XPN* is based on a linear model and assumes that the error terms follow a Normal distribution, but the authors did not model information from biological covariates. *Combat* can manage either information on batches and from biological covariates, but, however, makes strong parametric

assumptions (Normal and Inverse-Gamma distribution are used and graphical checks are suggested to assess the fit of the model).

In this paper we propose a method for the removal of location and scale batch effects from normalized data. Our purpose is to introduce a method based on weak model assumptions that is able to manage all the biological information available. We named such method *ber* (batch effects removal). We develop a two stage procedure where location effects are estimated at the first stage while batch scale effects are estimated at second stage. In both stages we use linear models accounting for batches and other biological variables of interest. We shall show that *ber* is effective in microarray context where the number of variables is huge and the sample size is low. In Sect. 2 we introduce the method. Section 3 is devoted to a simulation study for comparing *ber* with other commonly employed methods. In Sects. 4 and 5 we apply *ber* to real microarray studies. Some final considerations are given in Sect. 6.

2 Model and Method

The method proposed in this paper is based upon an extension of the model used by [11] for the estimation and removal of the batch effects. While the assumed models are similar, for the estimation of the parameters we propose a completely different method. Two procedures are described: one is suited for supervised analyses, the other for unsupervised analyses.

Johnson et al. [11] related the expression level of a gene in a subject to an overall mean expression for that gene, the influence of the biological covariates of interest and location/scale batch effects. Normally distributed error terms were adopted to model the variability not explained by such factors. They used an Empirical Bayes method for the estimation of the parameters. We propose instead a two stage procedure. This will give a computational advantage assuring at the same time an effective method in HDLSS (High Dimension Low Sample Size) contexts.

Let us denote with g the number of genes, with n the number of subjects and with m_b the number of batches. Let Y be the $n \times g$ matrix with the observed expression levels $Y(i, j)$, ($i = 1, \ldots, n$ and $j = 1, \ldots, g$). Denote with X_b an $n \times m_b$ matrix where the element (i, l) is equal to one if subject i belongs to batch l and zero otherwise ($l = 1, \ldots, m_b$), and with B_b the corresponding $m_b \times g$ matrix of parameters. Similarly we denote with X_c the $n \times m_c$ design matrix (without the column for the intercept) for modeling the effects of the biological covariates and with B_c the $m_c \times g$ matrix with the corresponding parameters. We want to fit the linear model

$$Y = XB + E \tag{1}$$

where $X = [X_b \; X_c]$ is a full rank matrix, $B' = [B_b' \; B_c']$ and E is a matrix of error terms with zero expectation.

A least squares estimate of this multivariate linear model is

$$\hat{B} = X^+ Y \tag{2}$$

where X^+ is the generalized Moore–Penrose inverse of X. Such estimate corresponds to the matrix with minimum norm among those minimizing the residuals sum of

squares (see [1]). A formal proof can be obtained using the singular value decomposition of a matrix. $\hat{B}$ can be partitioned as B in $\hat{B}' = [\hat{B}_b' \ \hat{B}_c']$.

At second stage we want to estimate the scale batch effects. Once $\hat{B}$ has been obtained we can consider the residuals given by the matrix $\hat{E} = Y - X\hat{B}$. These residuals have zero expectation because $X\hat{B}$ is a correct estimator of XB. However, a batch effect affecting the variability of these residuals could still be present. To estimate the scale batch effects we use a second regression on the squared residuals (for the use of a second regression on the residuals in a different context see [7]). The squared residuals are given by $\hat{E}^2 = \hat{E} \circ \hat{E}$ with $\circ$ denoting the Hadamard product. Similarly to what we did in the first stage, the expected value of $\hat{E}^2(i, j)$ can be model through the expression

$$\delta_{ij}^2 = \sum_{l=1}^{m_b} X_b(i, l) D_b(l, j)$$

where now D_b is an $m_b \times g$ matrix of parameters used to describe scale batch effects. The model can be written in matrix form as

$$\hat{E}^2 = X_b D_b + F \tag{3}$$

F being a matrix of errors with zero expectation. A least squares solution is

$$\hat{D}_b = X_b^+ E^2$$

and we can estimate δ_{ij}^2 by

$$\hat{\delta}_{ij}^2 = \sum_{l=1}^{m_b} X_b(i, l) \hat{D}_b(l, j).$$

Note that $\hat{D}_b(l, j)$ is the mean of the elements $E^2(i, j)$ in the batch l. This ensures that $\hat{D}_b(l, j)$ is positive and therefore that the elements of $X_b \hat{D}_b$ are positives. This cannot be ensured if the extended model

$$\hat{E}^2 = XD + F \quad \text{with } D' = [D_b' \ D_c']$$

is employed and unconstrained least squares are used.

In summary our two-stage procedure estimates B and D_b with two regressions in succession; first it estimates B and then with a regression on the squared residuals it estimates D_b. For gene j, $j = 1, \ldots, g$, let us denote with $\hat{\sigma}_j^2 = (1/n) \sum_{i=1}^n \hat{\delta}_{ij}^2$ the mean of its estimated scale batch effects. Once these estimates have been obtained the data are transformed to eliminate the batch effects through the following steps:

1. $Y_1 = Y - X\hat{B}$;
2. $Y_2 = Y_1 \circ \hat{\Delta}^{-1}$ where $\hat{\Delta}^{-1}$ is an $n \times g$ matrix with elements $\hat{\Delta}^{-1}(i, j) = 1/\hat{\delta}_{ij}$;
3. $Y_3 = Y_2 \circ \hat{\Delta}_2$ where $\hat{\Delta}_2(i, j) = \sqrt{\hat{\sigma}_j^2}$;
4. $Y_4 = Y_3 + \frac{1}{n}\mathbf{1} X_b \hat{B}_b + X_c \hat{B}_c$ where $\mathbf{1}$ is an $n \times n$ matrix of ones.

Similarly to [16] we propose a slightly different procedure for unsupervised analyses. In such exploratory analyses the effects of the biological covariates are not of direct interest and we can consider only the information about the batches. Using $\hat{B} = \hat{B}_b = X_b^+ Y$ the procedure is again given by the three steps above with step 4 replaced by step

4b. $Y_4 = Y_3 + \frac{1}{n}\mathbf{1}X_b\hat{B}_b$.

2.1 Bagging

The use of bagging was initially proposed by [3]. The idea is to obtain an aggregated predictor through the average of bootstrap versions of a proposed predictor. Breiman showed that the aggregated predictor has better accuracy. Schäfer and Strimmer [17] used these techniques to build estimators of the partial correlation matrix for inferring gene association networks. In particular they proposed three bagging estimators which are based on the use of the pseudo inverse. This led us to employ such tool in our method. We propose two procedures for bagging our method. Bootstrap samples of size n were created sampling with replacement the subjects from the original sample.

The first procedure is implemented as follows. For each bootstrap sample v, $v = 1, \ldots, V$ we get the estimates $\hat{B}_v$ and $\hat{D}_v$ of the matrices B and D as described previously. The bagging estimators are then

$$\hat{B}_{\text{bag}} = \sum_{v=1}^{V} \frac{\hat{B}_v}{V}, \qquad \hat{D}_{\text{bag}} = \sum_{v=1}^{V} \frac{\hat{D}_v}{V}.$$

The steps for the removal of the batch effects are then applied using such matrices instead of $\hat{B}$ and $\hat{D}$, respectively. We call such procedure *full bagging*.

In the second procedure the estimates $\hat{B}_v$ for $v = 1, \ldots, V$ are obtained as before but the normalization steps are applied on

$$\hat{B}_{\text{bag}} = \sum_{v=1}^{V} \frac{\hat{B}_v}{V}, \qquad \hat{D}_{\text{bag}} = X^+ \hat{E}_{\text{bag}}^2$$

where $\hat{E}_{\text{bag}} = Y - X\hat{B}_{\text{bag}}$. We call such procedure *partial bagging*.

3 Simulation Study

Real data sets do not allow the knowledge of true differentially expressed genes and the behavior of false positives and false negatives can be better understood using simulated data. For this purpose we designed a simulation study where the generated data had characteristics similar to those of real microarray applications. We considered two settings with different assumptions. In the first setting the data were generated for 54675 independent variables (as for GeneChip Human Genome U133 Plus 2.0 Arrays). In the second setting a smaller study with 1000 correlated variables was developed. We considered the case of a two groups comparison with samples belonging

to two different batches. Since the methods proposed in this paper will be applied on normalized data, we compared them with three similar commonly used methods: *mean centering*, *standardization* and *combat*. These were the methods with optimum performances in the comparison analyses performed by [4] or [13]. *Partial bagging* and *full bagging* used 100, 150 or 200 bootstrap samples.

To detect differentially expressed genes we used the shrinkage approach proposed by [18] and to avoid false positives we chose to control the local false discovery rate (see [6, 22]). Genes with an estimated local false discovery rate below 0.05 were considered differentially expressed. The number of true differentially expressed genes was in line with the model proposed by Efron for which a small proportion of differentially expressed genes is assumed. Each design was simulated 150 times.

For the first setting the means of the genes in the real study described in Sect. 4 were used as means of the simulated variables. We supposed 500 differentially expressed genes, with differences in expression (ξ_j) generated from a Cauchy distribution with location parameter 2 and scale parameter 1. With such distribution many parameters are close to zero and the detection of the respective differentially expressed genes more difficult. The aim was to see which method provided the greatest power in such situation. Each group had a sample size equal to 18. In the balanced case nine subjects belonged to the first batch and nine to the second one. In the unbalanced cases the subjects were distributed in the batches as follows: three and 15 in the first group, 15 and three in the second group, or six and 12 in the first group, 12 and six in the second group. Location batch effects were sampled from a Cauchy distribution with location parameter 0 and scale parameter 1. Similarly to [16] errors were generated from a Normal distribution with mean 0 and standard deviation equal to $\sqrt{0.2}$. Scale batch effects were simulated from a Uniform distribution between 0.5 and 2. In this way the variance of the errors in a specific batch can be dilated or shrunk.

In the second setting the means of the genes were generated from a Normal distribution with mean 6 and standard deviation $\sqrt{0.4}$. We supposed 100 differentially expressed genes, with differences in expression (ξ_j) generated from a Cauchy distribution with location parameter 2 and scale parameter 1. Sample sizes and allocation of the subjects into batches were set as above. Location batch effects were sampled from an Inverse Gamma distribution with shape parameter 2.5 and scale parameter 1.5 (this distribution is often used with empirical Bayes methods). Errors were generated from a Multivariate Normal distribution with a null mean vector and positive definite covariance matrix with variances between 0.5 and 1.5. Such matrix was generated with the function genPositiveDefMat in the R package clusterGeneration (see [10]). Using these errors the simulated genes are correlated. Scale batch effects were simulated from a Uniform distribution between 0.5 and 2.

3.1 Simulation Results

For all methods the false positive rate (FPR) was very close to zero and substantial differences can be noted only with regard to the true positive rate (TPR). Therefore we show only the results about the TPRs. In Tables 1 and 2 we report the means of the TPRs over the 150 simulations and the respective standard errors. The *standardization* method had always the worst performances. This was expected since a

Table 1 This table shows the TPRs for the simulations on the 54675 independent variables generated as described in the first setting

		$n_1 = n_2 = 18$ $(3-15; 15-3)$	$n_1 = n_2 = 18$ $(6-12; 12-6)$	$n_1 = n_2 = 18$ $(9-9; 9-9)$
ber full bagging 100	mean	0.8364	0.8846	0.8931
	se	0.0014	0.0010	0.0010
ber full bagging 150	mean	0.8376	0.8845	0.8932
	se	0.0014	0.0011	0.0010
ber full bagging 200	mean	0.8376	0.8845	0.8934
	se	0.0013	0.0010	0.0010
ber partial bagging 100	mean	0.8384	0.8855	0.8942
	se	0.0014	0.0010	0.0010
ber partial bagging 150	mean	0.8386	0.8855	0.8942
	se	0.0013	0.0011	0.0010
ber partial bagging 200	mean	0.8383	0.8853	0.8943
	se	0.0014	0.0011	0.0010
ber	mean	**0.8396**	**0.8857**	**0.8944**
	se	0.0014	0.0011	0.0010
combat	mean	0.8239	0.8742	0.8836
	se	0.0014	0.0011	0.0010
mean centering	mean	0.7838	0.8527	0.8747
	se	0.0015	0.0013	0.0010
standardization	mean	0.6939	0.7513	0.7561
	se	0.0038	0.0040	0.0034

simple standardization has often the effect of flattening the fold changes. Using *mean centering* the performances improved, but such method uses only information about the means in the batches and therefore was outperformed by *combat* and *ber* in both settings. In the first setting *ber* and its bagging version gave TPRs which are slightly higher than those of *combat*. As the correlation between the variables increased (second setting) also the gap between *combat* and such methods increased. The performances of all the methods improved with the balancing of the batches in each group.

4 A Multi-Center Leukemia Study

In this section we consider the data set analyzed by [12] which is freely available at Gene Expression Omnibus (GEO) repository (http://www.ncbi.nlm.nih.gov/geo/) with accession number GSE15434. To get the data we used GEOquery [5]. The data were normalized with RMA. In the data set there were 251 pediatric leukemia samples from three centers: Dresden, $n = 78$; Munich, $n = 96$; Ulm, $n = 77$. Affymetrix HG-U133 Plus 2.0 chips containing 54675 probe sets were used in these experiments. One of the main purposes of the authors was to study the nucleophasmin gene mutations, NPM1. There were 138 positive subjects (NPM1-mutated) and 113 negative

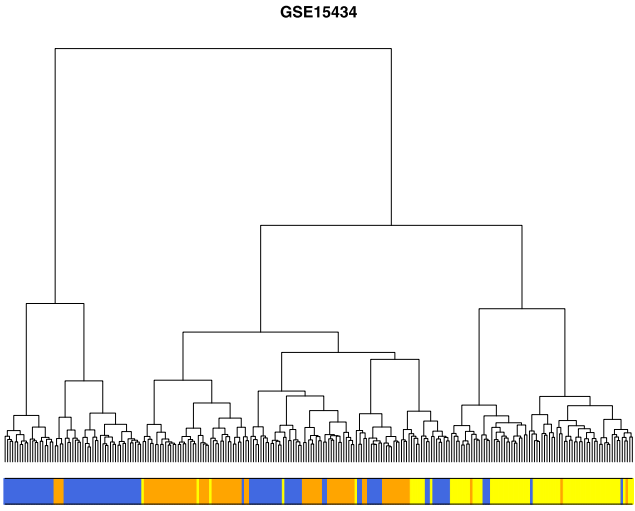
Table 2 This table shows the TPRs for the simulations on the 1000 correlated variables generated as described in the second setting

		$n_1 = n_2 = 18$ $(3-15; 15-3)$	$n_1 = n_2 = 18$ $(6-12; 12-6)$	$n_1 = n_2 = 18$ $(9-9; 9-9)$
ber full bagging 100	mean	0.4864	0.5488	0.5768
	se	0.0086	0.0095	0.0098
ber full bagging 150	mean	**0.4976**	0.5464	0.5802
	se	0.0083	0.0099	0.0094
ber full bagging 200	mean	0.4918	0.5437	0.5775
	se	0.0083	0.0104	0.0095
ber partial bagging 100	mean	0.4973	**0.5561**	0.5859
	se	0.0087	0.0096	0.0092
ber partial bagging 150	mean	0.4944	0.5492	0.5828
	se	0.0085	0.0102	0.0098
ber partial bagging 200	mean	0.4934	0.5534	**0.5862**
	se	0.0085	0.0099	0.0096
ber	mean	0.4973	0.5507	0.5827
	se	0.0087	0.0098	0.0092
combat	mean	0.4519	0.5070	0.5355
	se	0.0087	0.0110	0.0102
mean centering	mean	0.4058	0.4647	0.5054
	se	0.0085	0.0102	0.0103
standardization	mean	0.3612	0.4176	0.4416
	se	0.0091	0.0109	0.0103

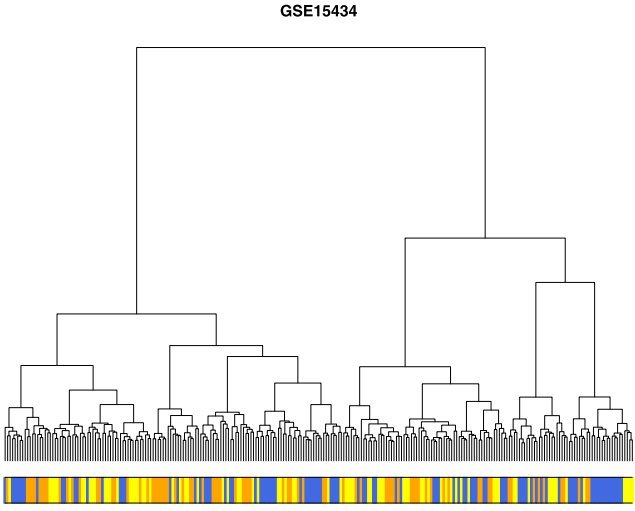
subjects (NPM1-unmutated). The authors used a linear model and restricted maximum likelihood for comparing the two groups while taking into account that samples belonged to three different centers. With such approach the batch effects were estimated but not removed.

The batch effects can be seen in Fig. 1(a) where the samples in the three largest groups identified through the dendrogram are prone to be clustered accordingly to their respective sources. We used *ber* for the removal of the batch effects. Figure 1(b) shows that the samples from the three centers are now well intermixed in the largest groups highlighted by the dendrogram.

We searched for differentially expressed probe sets as described in Sect. 3. If the batch effects were removed using *combat*, then 398 probe sets were identified as differentially expressed. Using *mean centering* the differentially expressed probe sets were 392. Instead applying *ber* to the normalized data the number of differentially expressed probe sets increased to 418. Similarly using bagging (*full bagging*, 200 bootstrap samples) we got 423 differentially expressed probe sets. These results highlight that our methods can lead to an appreciable increase in power. Let us remark that [12] reported a consensus signature for robustness across the three laboratories of 301 probe sets. Using our methods we obtained a great power in detecting differentially



(a) GSE15434's data set dendrogram before batch effects removal; the colored bar represents the batches: DRE; MUC; ULM.



(b) GSE15434's data set dendrogram after batch effect removal; the colored bar represents the batches: DRE; MUC; ULM.

Fig. 1 GSE15434's data set (Color figure online)

expressed probe sets, being sure at the same time that batch effects were not only estimated but also removed.

Among our 423 differentially expressed probe sets there are all the important genes discussed by [12]. Moreover we can enlarge the HOX family mentioned by the authors adding the genes HOXA11 and HOXB8 to their list. Therefore our findings lead to interpretable biological results and can strengthen their conclusions.

Remark 1 The results from the simulation studies in the previous section show that *ber* and its bagging versions lead to a number of false positives close to zero. However, for the GSE15434's data set we do not know which are the genes really differentially expressed and therefore the observed increase in power could be due to an increase in false positives. In the next section we avoid such possible bias comparing the methods on the basis of their prediction performances.

5 A Breast Cancer Study

We now analyze another data set freely available from GEO repository. The data set with accession number GSE2990 concerns a breast cancer study on 189 patients. The histologic grade is an important prognostic factor and [21] studied how it is associated with gene expression profiles of breast cancers. Affymetrix HG-U133A chips containing 22283 probe sets were used in this study. In their paper the CEL files were normalized separately in four groups, according to the institutions (Oxford = OXF or Uppsala = KI) and the batches of measurements (untreated or tamoxifen-treated series). We repeated the study normalizing all together the 167 samples for which the grade information is available: KIT $n = 21$, KIU $n = 64$, OXFT $n = 38$, OXFU $n = 44$. Specifically our interest was in the prediction of tumor grade (histological grade status 1, 2 or 3) from gene expression profiles.

Predictions were done using "shrunken centroids discriminant analysis" also known as PAM (Prediction Analysis for Microarrays) algorithm. We used the implementation proposed by [20] in the CMA bioconductor package. The error rate was evaluated on 50 training/learning sets. These were obtained through 5-fold cross-validation repeated 10 times. The error rate used to measure the performances was the misclassification rate. Hyperparameter tuning was performed by inner cross-validation (3-folds) on the learning sets, using the default grid of values in the CMA package.

The misclassification error rates were: *standardization* 0.4085, *mean centering* 0.3970, *combat* 0.3576, *ber* 0.3143, *full bagging ber (200 samples) 0.3130*. The two versions of *ber* have therefore the best predictive performances, followed by *combat* and finally by *mean centering* and *standardization*.

6 Discussion

We have proposed a two-step procedure for the removal of batch effects. In the first step we can consider only the effects of the batches or also the effect of other biological variables. In the second step we model only the scale effects of the batches. Let us note that similarly to [16] a further generalization of our procedure is possible: the effects of other non-biological study variables can be easily included in the proposed model through the design matrix X_b and the matrix of parameters B. This, in principle, can lead to a full normalization method. However, the purpose of this paper was the development of a method for the removal of the batch effects from normalized data. This allows the removal of the batch effects using again many well established normalization methods such RMA or others.

Our estimation procedure is not based on a specific parametric distribution of the expression levels. Avoiding strong assumptions on the data our strategy can be applied to a wide range of microarray studies. Moreover, since we do not use any specific parametric distribution we do not need to estimate the related parameters and therefore our procedure has a low computational cost.

A fundamental task when analysing microarray data is the detection of differentially expressed genes. We investigated the impact of our approach with reference to such goal, comparing it with other methods in terms of TPR (and FPR). Real and simulations studies show that *ber* can outperform the other methods in many settings. Such optimality was noted also in the analysis of a class prediction problem. In particular *ber* and its bagging versions had the best performances for correlated variables, which is the case for microarray data. In our simulations the use of bagging gave only a slight advantage over the simple *ber*, but bagging is, however, recommended in real applications to get robust results.

Acknowledgements The author is grateful to two referees for the helpful comments and valuable suggestions. The author wants to thank Mahmoodi Pezhman, Andrea Zangrando and Pietro Franceschi for the careful reading of the manuscript. This work was supported by Fondazione Cittá della Speranza.

References

1. Barnett S (1990) Matrices: methods and applications. Oxford University Press, Oxford
2. Benito M, Parker J, Du Q, Xiang D, Perou CM, Marron JS (2004) Adjustment of systematic microarray data biases. Bioinformatics 20(1):105–114. doi:10.1093/bioinformatics/btg385
3. Breiman L (1996) Bagging predictors. Mach Learn 24(2):123–140. doi:10.1023/A:1018054314350
4. Chen C, Grennan K, Badner J, Zhang D, Gershon E, Jin L, Liu C (2011) Removing batch effects in analysis of expression microarray data: an evaluation of six batch adjustment methods. PLoS ONE 6(2):e17238. 10.1371/journal.pone.0017238
5. Davis S, Meltzer PS (2007) GEOquery: a bridge between the Gene Expression Omnibus (GEO) and bioconductor. Bioinformatics 23(14):1846–1847. doi:10.1093/bioinformatics/btm254
6. Efron B (2004) Large-scale simultaneous hypothesis testing: the choice of a null hypothesis. J Am Stat Assoc 99(465):96–104. doi:10.1198/016214504000000089
7. Glejser H (1969) A new test for heteroskedasticity. J Am Stat Assoc 64(325):316–323
8. Huber WE, von Heydebreck A, Sultmann H, Poustka A, Vingron M (2002) Variance stabilization applied to microarray data calibration and to the quantification of differential expression. Bioinformatics 18(1):S96–S104
9. Irizarry RA, Hobbs B, Collin F, Beazer-Barclay YD, Antonellis KJ, Scherf U, Speed TP (2003) Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics 4(2):249–264
10. Joe H (2006) Generating random correlation matrices based on partial correlations. J Multivar Anal 97:2177–2189
11. Johnson WE, Li C, Rabinovic A (2007) Adjusting batch effects in microarray expression data using empirical Bayes methods. Biostatistics 8(1):118–127. doi:10.1093/biostatistics/kxj037
12. Kohlmann A, Bullinger L, Thiede C, Schaich M, Schnittger S, Döhner K, Dugas M, Klein HU, Döhner H, Ehninger G, Haferlach T (2010) Gene expression profiling in AML with normal karyotype can predict mutations for molecular markers and allows novel insights into perturbed biological pathways. Leukemia 24:1216
13. Luo J, Schumacher M, Scherer A, Sanoudou D, Megherbi D, Davison T, Shi T, Tong W, Shi L, Hong H, Zhao C, Elloumi F, Shi W, Thomas R, Lin S, Tillinghast G, Liu G, Zhou Y, Herman D, Li Y, Deng Y, Fang H, Bushel P, Woods M, Zhang J (2010) A comparison of batch effect removal methods for enhancement of prediction performance using MAQC-II microarray gene expression data. Pharmacogenomics J. 10:278–291

14. McCall MN, Bolstad BM, Irizarry MA (2010) Frozen robust multiarray analysis (fRMA). Biostatistics 11(2):242–253. doi:10.1093/biostatistics/kxp059
15. McCall MN, Uppal K, Jaffee HA, Zilliox MJ, Irizarry RA (2011) The gene expression barcode: leveraging public data repositories to begin cataloging the human and murine transcriptomes. Nucleic Acids Res 39:D1011–D1015. doi:10.1093/nar/gkq1259
16. Mecham BH, Nelson PS, Storey JD (2010) Supervised normalization of microarrays. Bioinformatics 26(10):1308–1315. doi:10.1093/bioinformatics/btq118
17. Schäfer J, Strimmer K (2005) An empirical Bayes approach to inferring large-scale gene association networks. Bioinformatics 21(6):754–764. doi:10.1093/bioinformatics/bti062
18. Schäfer J, Strimmer K (2005) A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. Stat Appl Genet Mol Biol 4(1):32. doi:10.2202/1544-6115.1175
19. Shabalin AA, Tjelmeland H, Fan C, Perou CM, Nobel AB (2008) Merging two gene-expression studies via cross-platform normalization. Bioinformatics 24(9):1154–1160. doi:10.1093/bioinformatics/btn083
20. Slawski M, Daumer M, Boulesteix AL (2008) CMA—a comprehensive bioconductor package for supervised classification with high dimensional data. BMC Bioinform 9:439. doi:10.1186/1471-2105-9-439
21. Sotiriou C, Wirapati P, Loi S, Harris A, Fox S, Smeds J, Nordgren H, Farmer P, Praz V, Haibe-Kains B, Desmedt C, Larsimont D, Cardoso F, Peterse H, Nuyten D, Buyse M, de Vijver MJV, Bergh J, Piccart M, Delorenzi M G (2006) Expression profiling in breast cancer: understanding the molecular basis of histologic grade to improve prognosis. J Natl Cancer Inst 98(4):262–272. doi:10.1093/jnci/djj052
22. Strimmer K (2008) A unified approach to false discovery rate estimation. BMC Bioinform 9:303. doi:10.1186/1471-2105-9-303
23. Zilliox MJ, Irizarry RA (2007) A gene expression barcode for microarray data. Nat Methods 4:911–913. doi:10.1038/nmeth1102