BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 18 2006, pages 2254–2261
doi:10.1093/bioinformatics/btl384

Gene expression

CART variance stabilization and regularization for high-throughput genomic data

Ariadni Papana[1] and Hemant Ishwaran[2,*]

[1]Department of Statistics, Case University, 10900 Euclid Avenue, Cleveland OH 44106, USA and [2]Department of Quantitative Health Sciences, Cleveland Clinic, 9500 Euclid Avenue, Cleveland OH 44195, USA

Received on March 22, 2006; revised on June 29, 2006; accepted on July 6, 2006
Advance Access publication July 14, 2006
Associate Editor: Joaquin Dopazo

ABSTRACT

Motivation: mRNA expression data obtained from high-throughput DNA microarrays exhibit strong departures from homogeneity of variances. Often a complex relationship between mean expression value and variance is seen. Variance stabilization of such data is crucial for many types of statistical analyses, while regularization of variances (pooling of information) can greatly improve overall accuracy of test statistics.

Results: A Classification and Regression Tree (CART) procedure is introduced for variance stabilization as well as regularization. The CART procedure adaptively clusters genes by variances. Using both local and cluster wide information leads to improved estimation of population variances which improves test statistics. Whereas making use of cluster wide information allows for variance stabilization of data.

Availability: Sufficient details for our CART procedure are given so that the interested reader can program the method for themselves. The algorithm is also accessible within the Java software package BAMarray^{TM}, which is freely available to non-commercial users at www.bamarray.com.

Contact: hemant.ishwaran@gmail.com

1 INTRODUCTION

It is unlikely that Gosset could ever have envisioned his ‘Student’ *t*-test (see Student, 1908) being used in scientific applications calling for tens of thousands, sometimes even hundreds of thousands, of simultaneous applications of the test. But this in fact is a common scenario seen today when searching for differentially expressing genes from DNA microarray data. The popularity of the *t*-test for microarray analysis can be explained in part by its simplicity and the speed at which it can be computed. Computationally simple and efficient procedures are obviously highly desirable when working with high-throughput data. The wide spread use of the *t*-test, however, also stems from methodological considerations. An important and well-recognized aspect of the *t*-test is its ability to account for the underlying variability in the data. This property makes it preferable to simple-minded methods that focus only on fold-changes.

In the context of a microarray experiment, the *t*-test can notationally be described as follows. Let $Y_{i,j}$ be the expression value for a gene-transcript j from microarray chip i, where $i = 1, \ldots, n$ and $j = 1, \ldots, P$. Each of the n chips are assumed to have a specific group label, which for example could be the phenotype corresponding to the underlying target sample, or if the samples are collected from tissues of different stages of a disease process, the group label could indicate stage of progression of disease. We indicate group labels using a group membership variable $\mathcal{G}_i \in \{1, \ldots, g\}$. For example, in a study with microarray data collected from two types of tissues (for concreteness, say control and diseased), $g = 2$ and either $\mathcal{G}_i = 1$ or $\mathcal{G}_i = 2$ with the label indicating the type of tissue sample (control or diseased) being interrogated by chip i.

In a two group problem, the *t*-test for testing for a differential effect for a specific gene j is

$$t_j = \frac{\overline{Y}_{1,j} - \overline{Y}_{2,j}}{\sqrt{\widehat{\sigma}^2_{1,j}/n_1 + \widehat{\sigma}^2_{2,j}/n_2}}, \qquad (1)$$

where n_k is the sample size for group k, $\overline{Y}_{k,j} = \sum_{\{i:\mathcal{G}_i=k\}} Y_{i,j}/n_k$ is the mean for group $k = 1, 2$ and

$$\widehat{\sigma}^2_{k,j} = \frac{1}{(n_k - 1)} \sum_{\{i:\mathcal{G}_i=k\}} (Y_{i,j} - \overline{Y}_{k,j})^2$$

is the sample variance for group k. Most often (1) is applied using Welch’s approximate degrees of freedom.

Equation (1) is a two-sample *t*-test with unequal variances. However, if population variances are anticipated to be equal ($\sigma^2_{1,j} = \sigma^2_{2,j}$), precision can be improved using an equal variance test statistic:

$$t_j = \frac{\overline{Y}_{1,j} - \overline{Y}_{2,j}}{\widehat{\sigma}_j \sqrt{1/n_1 + 1/n_2}}, \qquad (2)$$

having $n-2$ degrees of freedom, where $n = n_1 + n_2$ and

$$\widehat{\sigma}^2_j = \frac{1}{(n-2)} \sum_{k=1}^{2} (n_k - 1)\widehat{\sigma}^2_{k,j}.$$

is a pooled estimate of the population variance.

On the other hand, if population variances are equal and these values are known, then further power gains can be obtained. If $\sigma^2_{1,j} = \sigma^2_{2,j} = \sigma^2_j$, and σ^2_j is known, a more accurate testing approach is obtained using

$$t_j = \frac{\overline{Y}_{1,j} - \overline{Y}_{2,j}}{\sigma_j \sqrt{1/n_1 + 1/n_2}}. \qquad (3)$$

[*]To whom correspondence should be addressed.

Notice that (3) has infinite degrees of freedom since it uses the true population variance σ_j^2. Of course, computing (3) is hypothetical since population variances are unknown in practice. However, if genes share population variances, and genes can be clustered by these shared values, then given the tremendous amount of data available within a microarray experiment, it becomes possible to estimate shared variance parameters with high accuracy and to compute t-statistics along the lines of (3). A key goal of this paper will be to show how to cluster genes and to combine information across the data to achieve these kinds of accurate estimates. Our approach is based on a Classification and Regression Tree (CART) splitting algorithm. At the same time while we seek regularization, we show that our CART procedure also has the property that it variance stabilizes the data. In itself variance stabilization is important because microarray data often exhibit severe departures from homogeneity of variances. Often one sees a complex relationship between the mean expression of a gene and its sample variance. Since many statistical procedures, other than t-tests, rely on equality of variances for improved inference, variance stabilization is crucial.

2 METHODS

The simplest way to estimate the population variance for a gene is by using the sample variance of its expression values. More precise estimates, however, are possible by carefully synthesizing and combining information from expression values of other genes. We refer to this method for improved estimation of population variances as variance regularization.

Variance regularization is a well-studied and well-appreciated concept in the microarray literature. Perhaps the earliest such discussion appeared in Baldi and Long (2001). There, variance regularization was considered in the context of a two-sample t-test with unequal variances similar to (1). An empirical Bayes approach was employed where population variances were estimated by a weighted mixture of the sample variance $\widehat{\sigma}_{k,j}^2$ and an overall inflation factor selected using expression values from all the data. A similar approach was used in Tusher *et al*. (2001) and Efron *et al*. (2001). There, permutation methods were applied to t-tests involving pooled sample variances inflated by using an overall 'fudge factor' (the term 'fudge factor' was actually coined in the SAM software package and not the previous papers). In a similar vein, shrinkage estimation was used in Wright and Simon (2003), Smyth (2004), Cui *et al*. (2005) and Ji and Wong (2005) as a means for regularization. For example in Cui *et al*. (2005), F-tests for identifying genes with differential effects were used. Different estimates for the variance were considered, including a Stein-based shrinkage estimator. In Ji and Wong (2005), empirical Bayes shrinkage estimation was used. Interestingly, here the application was not to microarrays but to the class of tiling arrays, thus showing the idea of variance regularization is growing in its usage.

2.1 Stabilization and regularization via clustering

A common ingredient to each of the previous methods involves shrinkage of the sample variance to a global value. Here we discuss a different approach following the method of Ishwaran and Rao (2003, 2005). In this approach genes are clustered into groups with similar variances, and gene population variances are estimated using information from the cluster. This is a different type of shrinkage of the sample variance, being more akin to adaptive local shrinkage.

2.2 Clustering

The clustering method of Ishwaran and Rao (2003, 2005) applies to any number of groups g assuming equality of variances across experimental groups. For each gene j, it is assumed

$$\sigma_{1,j}^2 = \cdots = \sigma_{g,j}^2 = \sigma_j^2. \tag{4}$$

Assumption (4) will be the starting point for our algorithm. Later, in Section 3.3, we extend the approach to address unequal variances across groups.

The method of Ishwaran and Rao (2003, 2005) works as follows. Define the pooled sample variance, $\widehat{\sigma}_j^2$, by

$$\widehat{\sigma}_j^2 = \frac{1}{(n-g)} \sum_{k=1}^{g} (n_k - 1)\widehat{\sigma}_{k,j}^2.$$

Cluster genes by their pooled sample standard deviation, $\widehat{\sigma}_j$. Create C clusters, where C is some number greater than or equal to 1. In Ishwaran and Rao (2005), an ad hoc deterministic rule was used for clustering $\widehat{\sigma}_j$. In Section 2.6 we discuss a more systematic approach using CART.

Let $\mathscr{C}$ denote the resulting cluster configuration. Rescale the data within each cluster l of $\mathscr{C}$ by dividing all expression values by the square root of the cluster mean pooled sample variance. That is, all expression values $Y_{i,j}$ in a given cluster are transformed to $Y_{i,j}^* = Y_{i,j}/\widehat{\sigma}(l_j)$, where l_j denotes the cluster j belongs to, and

$$\widehat{\sigma}^2(l) = \frac{1}{\#\{j : l_j = l\}} \sum_{l_j = l} \widehat{\sigma}_j^2.$$

Because all expression values within a cluster are multiplied by the same value, the signal to noise ratio for any given gene (mean value to standard deviation ratio) remains unchanged.

Typically, the number of genes within any given cluster will be large. Because of this, $\widehat{\sigma}^2(l)$ should precisely estimate the shared population variance of the cluster, $\sigma^2(l_j)$. Under the assumption (4), and ignoring variability in $\widehat{\sigma}^2(l_j)$, we have $V(Y_{i,j}^*) = \sigma^2(l_j)/\widehat{\sigma}^2(l_j)$, which should be approximately equal to one. Consequently, the data are transformed in such a way that homogeneity of variances is satisfied.

Increased precision in estimating $\sigma^2(l_j)$ has another important benefit: it can be used to derive a regularized t-test. For ease of notation, consider the case when $g = 2$. The test for detecting a differential effect for j is defined as

$$t_j = \frac{\overline{Y}_{1,j}^* - \overline{Y}_{2,j}^*}{\tau_j \sqrt{1/n_1 + 1/n_2}}, \tag{5}$$

where $\tau_j^2 = \sigma^2(l_j)/\widehat{\sigma}^2(l_j)$. Setting $\tau_j = 1$ yields the hypothetical t-test (3) when $\sigma_j^2 = 1$ with an infinite degrees of freedom. This is a form of regularization that should lead to increased accuracy in identifying differentially expressing genes. Other forms of regularization are also possible. These simply involve substituting different values for τ_j, obtained using information from the cluster l_j. We illustrate one such approach in Section 4.

2.3 Stopping rules for C

These types of benefits are highly dependent on the number of clusters C used in the clustering procedure. Care must be taken to ensure that C is selected appropriately. If C is too large, overfitting is likely and poor regularization and stabilization will result. For example, if the number of clusters equals the number of genes ($C = P$), then the transformed pooled standard deviations, denoted by $\widehat{\sigma}_j^*$, will satisfy $\widehat{\sigma}_j^* = 1$ for each j. But this is likely to be a poor transformation. Even if an equal variance model is true, we still expect variability in $\widehat{\sigma}_j^*$ around 1. Therefore, rather than choosing a large value of C, and potentially losing power, the prefered method is to start with $C = 1$, in which all genes are assumed to have the same variance, and then gradually increase C until an equal variance model is justified.

To determine an appropriate value for C, a distance measure approach is used. This measure is calculated as follows. After transforming the data compute $\widehat{\sigma}_j^*$. Calculate the empirical distribution function for $\widehat{\sigma}_j^*$ and compare this to the theoretical null distribution for $\widehat{\sigma}_j$ under assumption (4) assuming $\sigma_j = 1$ for all j:

$$\sigma_{1,j} = \cdots = \sigma_{g,j} = \sigma_j = 1, \quad j = 1, \ldots, P. \tag{6}$$

If $\hat{F}_{\mathscr{C}}$ is the empirical distribution function for $\widehat{\sigma}_j^*$ under $\mathscr{C}$ and F is the theoretical null distribution for $\widehat{\sigma}_j$, the distance between the distributions is defined to be

$$\mathscr{D}(\mathscr{C}) = \frac{1}{99} \sum_{s=1}^{99} \left| \hat{F}_{\mathscr{C}}\left(\frac{s}{100}\right) - F\left(\frac{s}{100}\right) \right|.$$

The smallest such $\mathscr{D}(\mathscr{C})$ indicates the best configuration $\mathscr{C}$.

2.4 Variance stabilization and regularization algorithm

In summary, the algorithm is described as follows:

```
1: for $C = 1$ to 100 do
2:    Select an appropriate cluster configuration $\mathscr{C}$ with $C$ clusters.
3:    Rescale expression values within each cluster $l$ by dividing by $\widehat{\sigma}(l)$.
4:    Calculate $\mathscr{D}(\mathscr{C})$.
5: end for
6: The optimal $\mathscr{C}$ is the value with the smallest $\mathscr{D}(\mathscr{C})$.
7: Rescale observations using the optimal $\mathscr{C}$. All population variances
   are assumed to equal one for the transformed data.
```

2.5 Comments

- Observe that $\mathscr{D}(\mathscr{C})$ uses only the 1–99th percentiles of $\widehat{\sigma}_j^2$. This is for computational reasons since without some type of restriction the algorithm becomes too slow. Our experience has shown the space of trees under this restriction to be more than rich enough.
- In Ishwaran and Rao (2005), F was defined to be the distribution function for the square root of a χ^2-random variable with $n-g$ degrees of freedom. This is the distribution for $\widehat{\sigma}_j$ assuming normality for the data under the null (6). We adopt this approach here. This greatly simplifies computations and also compares favorably to non-parametric choices, such as permutation methods that we have experimented with.
- It is crucial to select a good cluster configuration for a given C. Ishwaran and Rao (2005) used an ad hoc deterministic rule based on pre-selected percentile values of $\widehat{\sigma}_j$. However, relying on pre-selected percentile splits may not always work well. Another concern is that the method uses a top-down approach starting from larger percentiles and working downwards. If in the distribution of $\widehat{\sigma}_j^2$ there is significant heterogeneity in smaller values, then a large number of splits is needed before the algorithm reaches these values. This will force a large number of clusters and will over-regularize the data.

2.6 CART gene clustering

The previous point makes it clear a more systematic approach for clustering genes is needed. For this reason we introduce a CART-like mechanism for creating clusters [for more background on CART see Breiman *et al.* (1984)]. This method also uses the percentiles for $\widehat{\sigma}_j$ to define clusters, but percentile splitting values are data adaptively chosen by maximizing node purity of the tree in a classical CART-like approach.

Node purity used for splitting a tree is defined by $\Delta(\mathscr{C}) = 1/(1 + \mathscr{D}(\mathscr{C}))$, where $\mathscr{C}$ represents the tree-cluster. In this way, maximizing node purity is equivalent to minimizing the distance measure $\mathscr{D}(\mathscr{C})$. The CART procedure can be briefly summarized as follows (note: as mentioned earlier, we restrict permissible splits to the 1st through 99th percentiles of $\widehat{\sigma}_j$, which we denote by $Q_1, \ldots, Q_{99}$):

(1) Begin at the root node. This corresponds to $C = 1$ and cluster configuration $\mathscr{C}_1 = \{\widehat{\sigma}_1, \ldots, \widehat{\sigma}_P\}$. Define the purity of the root node as $\Delta_1(*) = (1 + \mathscr{D}(\mathscr{C}_1))^{-1}$.

(2) To form a cluster of size $C = 2$ split the data at Q_s. All $\widehat{\sigma}_j \le Q_s$ become cluster 2, all other values become cluster 1. Call the resulting configuration $\mathscr{C}_2(s)$ and let $\Delta_2(s) = (1 + \mathscr{D}(\mathscr{C}_2(s)))^{-1}$ denote its node purity. The best cluster configuration of size $C = 2$ is denoted by $\mathscr{C}_2 = \mathscr{C}_2(s^*)$. This configuration satisfies $\Delta_2(s^*) \ge \Delta_2(s)$ for all s.

(3) Form clusters of size $C = 3$ by splitting $\mathscr{C}_2$ using the remaining splits in $\{Q_1, \ldots, Q_{99}\}$. Let $\mathscr{C}_3(s)$ be the cluster configuration of size $C = 3$ obtained by using the split Q_s. The best configuration of size $C = 3$ is denoted by $\mathscr{C}_3 = \mathscr{C}_3(s^*)$ and satisfies $\Delta_3(s^*) \ge \Delta_3(s)$ for all s.

(4) Repeat. Select the cluster configuration $\mathscr{C}^* = \mathscr{C}_C(s^*)$ having the highest node purity $\Delta_C(s^*)$, where $1 \le C \le 100$.

(5) To variance stabilize the data, cluster $\widehat{\sigma}_j$ using the optimal configuration $\mathscr{C}^*$. Rescale expression values within each cluster $l \in \mathscr{C}^*$ by dividing by $\widehat{\sigma}(l)$.

Note: The algorithm used throughout the paper followed the procedure outlined above but with one small change made to reduce computations. Rather than determining cluster configurations for each value $C = 1, \ldots, 100$, the algorithm included a check which halted at the first sign of a decrease in node purity. While this does not guarantee a global maximum, our experience shows the node purity function is concave in C, and when node purity begins to decrease, it continues to decrease.

3 ILLUSTRATION: VARIANCE STABILIZATION

3.1 Cognitive impairment

For our first illustration we consider a microarray experiment based on rat brain tissue [Blalock *et al.* (2003)]. The goal of this study was to identify gene expressions involved in age-dependent cognitive decline. Male Fischer rats aged 4 months (Young, $n_1 = 10$), 14 months (Middle aged, $n_2 = 9$) and 24 months (Aged, $n_3 = 10$) were trained sequentially over 7 days after which hippocampal tissue was collected and then interrogated using Affymetrix microarrays. The data are available at the National Center for Biotechnology Information (NCBI) Gene Expression Omnibus (GEO) data repository under series record accession number GSE854. See www.ncbi.nlm.nih.gov/geo.

The data were already normalized and we applied our CART procedure directly to the normalized data without any further pre-processing. This is the typical scenario for how our method is applied in practice. Microarray data are first normalized prior to inference, and because such data often exhibit departures from equality of variances, it is advisable to variance stabilize it before inference. Our procedure works well with popular normalization methods such as MAS 5.0 (Affymetrix, 2001) and RMA (Irizarry *et al.* 2003).

The fact that normalization procedures do not necessarily stabilize the variance is made amply clear in Figure 1. There we have plotted the mean expression value versus the standard deviation for each age group. Note the increase in standard deviation as mean expression increases. Clearly there is a severe violation of homogeneity of variance.

Figure 2 depicts the percentile splitting values found by the CART procedure. The y-axis on the left-hand side plots splits as a function of C (where $C = 1, \ldots, 100$), while the y-axis on the right-hand side are pooled standard deviations for the percentile. Each line on the plot is the splitting value for a cluster l traced out as C increases starting from when it is formed. For example, when $C = 2$ the best split is Q_{71}, which is roughly equal to a pooled

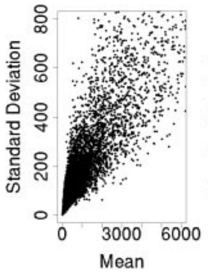

Fig. 1. Mean expression value versus standard deviation for each gene ($P = 8740$) from brain tissue data (Blalock *et al.*, 2003). Plots from left to right are Young aged, Middle aged and Aged rats.

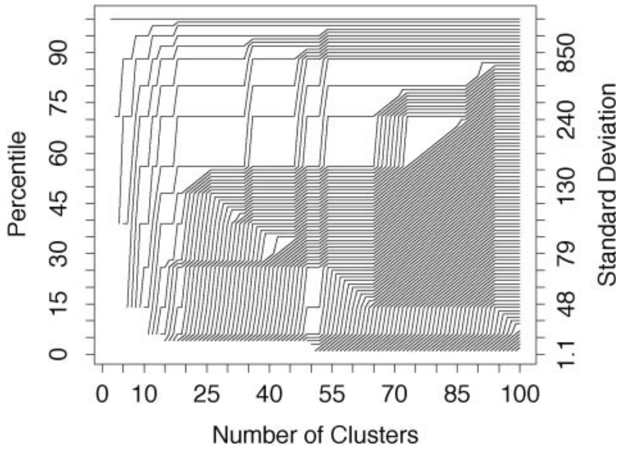

Fig. 2. Cluster configurations defined by percentiles (y-axis on left) and pooled standard deviations (y-axis on right) using the CART Stabilization-Regularization algorithm applied to brain tissue data.

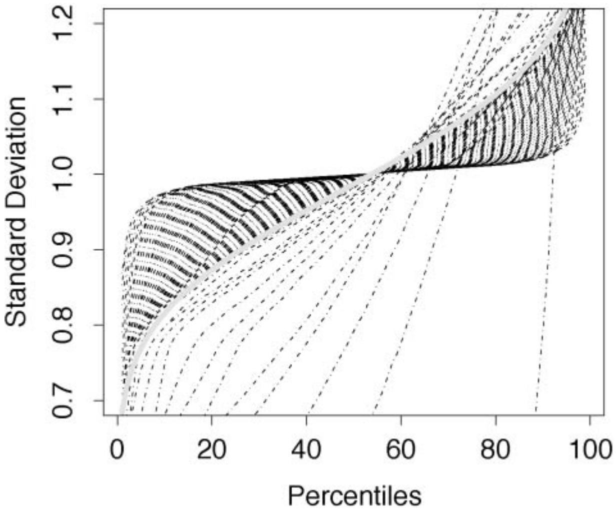

Fig. 3. Percentile values of pooled standard deviations after transforming brain tissue data using C clusters ($C = 1$, corresponding to the original non-transformed data, is the most vertical curve, whereas $C = 100$ is nearly horizontal). Thick gray line is target null.

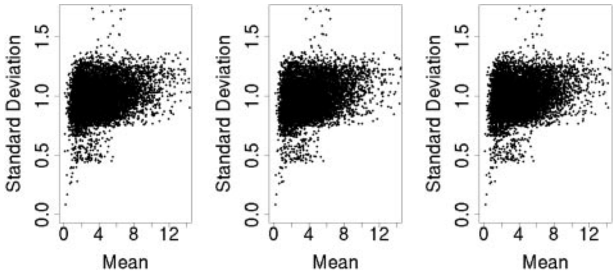

Fig. 4. Mean expression values versus standard deviations from CART transformed data (brain tissue data). Plots from left to right are Young aged, Middle aged and Aged rats.

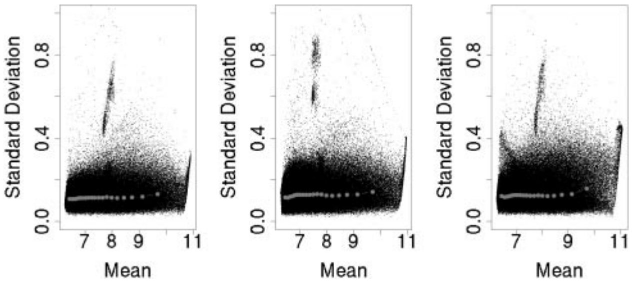

Fig. 5. Mean expression versus standard deviation using the Bioconductor procedure vsn(). Here each point on the plot corresponds to a probe from background corrected raw CEL data. As before, plots from left to right are Young, Middle and Aged rats.

standard deviation of 240. Cluster 1 is $\widehat{\sigma}_j > Q_{71}$ and cluster 2 corresponds to $\widehat{\sigma}_j \leq Q_{71}$.

Figure 2 shows that clusters split rapidly early on. It is not until $C = 20$ and higher that clusters start to stabilize. Clearly the algorithm is efficiently carving up the data with very few splits. Figure 3 shows how well $\hat{F}_{\mathscr{C}}$ approximates the null target distribution function F as a function of C. On the plot we have superimposed the 99 percentiles of the transformed pooled standard deviations for each cluster configuration $\mathscr{C}_C$ for $C = 1$ to $C = 100$. The thick gray line is the theoretical values under F. As can be seen there is a rapid convergence to the null distribution, after which overfitting starts to occur. As C gets very large all standard deviations begin to converge to the value of one and overfitting will be very bad. Our algorithm stops very close to the theoretical null (recall that our procedure terminates once node purity starts to decrease; to produce Figures 2 and 3 we forced our algorithm to investigate all C values). The optimal value was $C = 15$ with a node purity value of 0.997 (the maximum value possible being 1.0).

The effectiveness of the transformation can be seen in Figure 4. There we have plotted the mean expression versus the standard deviation for the CART transformed data. Except for a small clump of genes with standard deviations near 0.5 the transformation is seen to be highly effective. As a comparison we have also plotted the means and standard deviations obtained using the variance stabilizing procedure of Huber *et al.* (2002). This procedure can be used for simultaneous normalization and variance stabilization of the data. We implemented the method using the 'vsn()' command in Bioconductor [see Chapter 2 of Gentleman *et al.* (2005) for illustration]. Figure 5 are the plots derived using the original

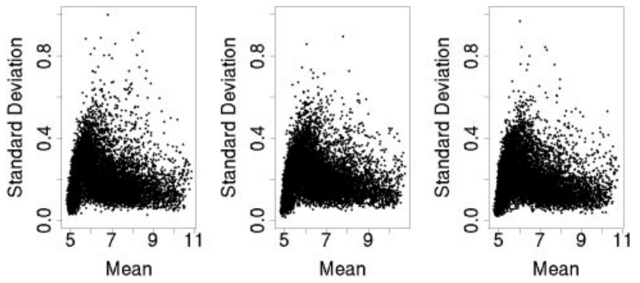

Fig. 6. vsn() applied to normalized data of Figure 4.

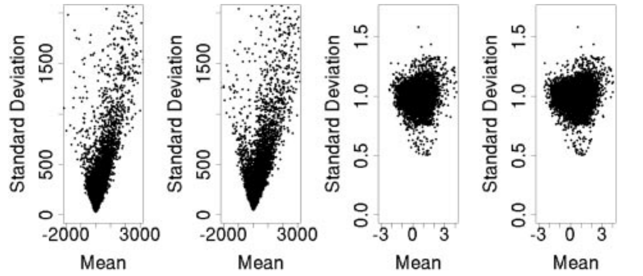

Fig. 7. Mean expression values versus standard deviations from embryonal brain tumor data [dataset C from Pomeroy *et al*. (2001)]. First two plots are deaths and survivors from untransformed data, while last two plots are deaths and survivors from CART transformed data ($C = 17$ with node purity 0.995).

CEL files. The vsn() procedure can also be applied to any data matrix. Figure 6 was obtained using vsn() on the normalized data used in Figure 4.

3.2 Unequal variances across groups

So far our algorithms have assumed sample variability may differ across genes but not across experimental groups. The next example shows this assumption may not always hold. We illustrate a graphical tool for assessing equality of variances across groups and discuss how to modify the CART algorithm in this case.

For our example we consider dataset C from Pomeroy *et al*. (2001). Here tissue samples were collected from 60 children with medulloblastomas. Of these data, 21 samples came from individuals who died (group 1) and 39 from survivors (group 2). Samples were interrogated using HuGeneFL DNA microarrays. The resulting data were then normalized using software from Affymetrix [see Pomeroy *et al*. (2001)]. In total there were $P = 7129$ genes on each array. The data can be found at www.broad.mit.edu/mpr/CNS.

Figure 7 plots the mean expression value versus the standard deviation for each of the two groups. Once again we see that the CART procedure has effectively stabilized the variance.

However, we had some concerns regarding the assumption of equal variances across the two groups. To graphically test this we used the following procedure. Using the optimal cluster configuration found by CART we separately transformed the data for each of our two groups. This was done by dividing the expression values in group $k \in \{1, 2\}$ in cluster l by the square root of the mean cluster sample variance for the group:

$$\widehat{\sigma}_k^2(l) = \frac{1}{\#\{j : l_j = l\}} \sum_{l_j = l} \widehat{\sigma}_{k,j}^2.$$

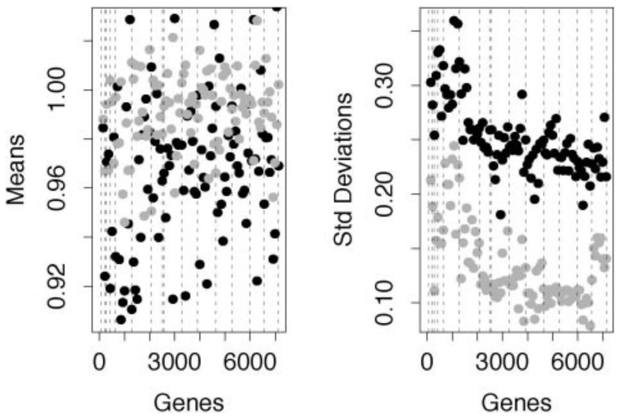

Fig. 8. CART clustering configuration was used to separately transform survivors (gray points) and deaths (black points) from embryonal tumor data. Means and standard deviations of the transformed standard deviations are given in left and right plots respectively. Dashed vertical lines are percentile splitting points of the cluster configuration (genes have been sorted so that percentile splitting values decrease going from left to right).

Thus, if chip i belongs to group k, the transformed expression value for a gene j is $Y_{i,j}^* = Y_{i,j}/\widehat{\sigma}_k(l_j)$ and the transformed sample variance is $\widehat{\sigma}_{k,j}^2/\widehat{\sigma}_k^2(l_j)$.

We then combined the transformed standard deviations for each group k into blocks of sample size 100 within a CART specified cluster and computed the resulting mean and standard deviation of each block. These values are presented in Figure 8.

If population variances are equal between survivors and deaths, the mean value of the transformed standard deviations for each group should be concentrated near one. However, Figure 8 shows transformed standard deviations for deaths are systematically lower (left plot) with systematically higher variability (right plot). The higher variability is owing to a sample size effect and is not unexpected. Recall that the sample size for survivors is $n_2 = 39$, while for deaths $n_1 = 21$. The lower mean values, however, cannot be explained away, and suggests variability differences across the groups.

3.3 Clustering strategy for unequal variances

To deal with the issue of differing variances across groups we modify our CART clustering procedure. Recall $\widehat{\sigma}_k^2(l_j)$ is the mean cluster sample variance for gene j for group k, where cluster formation is based on the pooled standard deviations over all groups. Since population variances are unequal, it stands to reason that a better cluster configuration, and hence better regularized estimate of variance, can be obtained by working with each group separately.

The idea is straightforward and applies to more than two groups. One simply runs the CART procedure separately on each group k, obtaining an optimal cluster configuration $\mathscr{C}_k^*$. For example, for two groups k and k', merge the two cluster configurations into a more refined cluster configuration $\mathscr{C}^*$ and use this new configuration for regularized estimates of population variances. For example, if 'x' represents a gene, '|' a cluster boundardy, and $\mathscr{C}_1^*$ and $\mathscr{C}_2^*$ are indicated by the top and bottom rows of the left-side of the figure below, then the refined cluster $\mathscr{C}^*$ formed by merging $\mathscr{C}_1^*$ and $\mathscr{C}_2^*$ is given on the right of the figure:

$$\begin{array}{cccccccccccccccc} \mathrm{x} & \mathrm{x} & | & \mathrm{x} & & \mathrm{x} & | & \mathrm{x} & \Rightarrow & \mathrm{x} & \mathrm{x} & | & \mathrm{x} & | & \mathrm{x} & | & \mathrm{x} \\ \mathrm{x} & \mathrm{x} & & \mathrm{x} & | & \mathrm{x} & & \mathrm{x} & & \mathrm{x} & \mathrm{x} & | & \mathrm{x} & | & \mathrm{x} & | & \mathrm{x} \end{array}$$

In general, the new cluster configuration will contain more clusters than either of the original configurations, but in our experience will not be so large that over-regularization occurs. For example, for the embryonal brain tumor data, $\mathscr{C}_1^*$ and $\mathscr{C}_2^*$ had $C = 10$ and $C = 14$ clusters, respectively, while the merged cluster configuration, $\mathscr{C}^*$, contained $C = 20$ major clusters (clusters with more than 100 genes). Let $\widehat{\sigma}_1^*(l_{1,j})$ and $\widehat{\sigma}_2^*(l_{2,j})$ denote the square root of the mean cluster sample variance for gene j from groups 1 and 2 based on $\mathscr{C}^*$. The modified regularized unequal variance t-test is

$$t_j^* = \frac{\overline{Y}_{1,j} - \overline{Y}_{2,j}}{\sqrt{\widehat{\sigma}_1^{*2}(l_{1,j})/n_1 + \widehat{\sigma}_2^{*2}(l_{2,j})/n_2}}.$$

In our experience this modification works very well when equality of variances across groups is suspect. We note that the variance stabilization procedure discussed above is incorporated in the 3.0 release of BAMarray™ and an interactive diagnostic plot is available for assessing accuracy of stabilization. Interested readers should visit www.bamarray.com for more details.

4 ILLUSTRATION: REGULARIZATION

Performance of the CART regularized t-test was studied using two sets of data. Our first set of data was synthetically produced by simulation from the well-known additive and multiplicative error model of Durbin *et al*. (2002). Our second example used microarray data from the rich spike-in experiment of Choe *et al*. (2005). Performance in both cases was measured by how well the CART t-test performed relative to the following tests: (1) conventional t-test, (2) t-test computed by logarithmically transforming the data, (3) Cyber-T test described in Baldi and Long (2001), available as R-code at http://visitor.ics.uci.edu/genex/cybert and (4) the Δh difference statistic of Huber *et al*. (2002) calculated using the vsn() procedure discussed earlier.

In addition to the CART t-test, we also studied the performance of a hybrid CART t-test. In place of the mean pooled variance of a cluster, the hybrid test used a weighted estimate of the variance. The 'local' variance for a gene (for a specific group) within a cluster l was estimated by averaging out the closest 100 neighbors to it in terms of mean signal (this is similar to the strategy used by Cyber-T). For the hybrid test, the sample variance for a gene for a given group was calculated by taking this value and multiplying it by W, the fraction of observations within the cluster relative to the number of genes, and then adding to this $(1 - W)$ times the mean pooled variance of the cluster. In this way, clusters with small numbers of genes will have small values for W and will tend towards the overall mean pooled variance of the cluster, thus sharing information more globally. For bigger clusters, we have the luxury of more local sharing of information. In such cases this is achieved because W will be large. The hybrid test, similar to Cyber-T, applies in unequal variance settings only.

4.1 Simulated data

Our synthetic data were simulated using the additive and multiplicative error model of Durbin *et al*. (2002). Specifically, for our simulation we took $P = 10\,000$, $n = m = 5$ where expression values for group 1 were sampled using

$$Y_{i,j} = \mu_{1,j} + (\mu_{1,j} + \beta_1)\exp(\rho\eta_{i,j}) + \nu_1\varepsilon_{i,j}, \qquad (7)$$

for $i = 1, \ldots, n_1$, whereas for group 2,

$$Y_{i,j} = \mu_{2,j} + (\mu_{2,j} + \beta_2)\exp(\rho\eta_{i,j}) + \nu_2\varepsilon_{i,j}, \qquad (8)$$

for $i = n_1 + 1, \ldots, n_1 + n_2$. We took $\{\eta_{i,j}, \varepsilon_{i,j}\}$ to be i.i.d. N (0,1) variables. This type of model ensures the sample variance for a gene will be related to its mean signal. In particular, when $\eta_{i,j}$ is small, the means in (7) and (8) are dominated by $\mu_{1,j}$ and $\mu_{2,j}$ respectively. However, when $\eta_{i,j}$ is large, the means are dominated by $(\mu_{1,j} + \beta_1)M$ and $(\mu_{2,j} + \beta_2)M$, respectively, where $M = E\{\exp(\rho\eta_{i,j})\}$. At the same time there is a corresponding increase in the variance by $(\mu_{1,j} + \beta_1)^2V$ and $(\mu_{2,j} + \beta_2)^2V$, respectively, where $V = \mathrm{Var}\{\exp(\rho\eta_{i,j})\}$.

With 80% probability, we selected $\mu_{1,j} = \mu_{2,j} = 0$. The other 20% of the time, corresponding to differentially expressing genes, $\mu_{1,j}$ and $\mu_{2,j}$ were independently sampled from an exponential density with mean k, where k was randomly sampled from $\{1, \ldots, 10\}$. We ran two distinct simulations. For our first simulation, referred to as Synthetic (i), we set $\beta_1 = \beta_2 = 1$, $\rho = 0.2$ and $\nu_1 = \nu_2 = 1$. This represents a setting where variances are equal over groups. Observe that ρ is chosen significantly smaller than ν. In particular, the value $\rho = 0.2$ ensures the standard deviation $V^{1/2}$ is ~21% of the standard deviation of $\varepsilon_{i,j}$, which is not uncommon in practice. Our second simulation, Synthetic (ii), used $\beta_1 = \beta_2 = 1$ and $\nu_1 = 1$ and $\nu_2 = 3$. Because ν_2 is larger than ν_1, this represents a setting where variances are unequal across groups.

4.2 Spike-in data

Our second example uses the spike-in array data of Choe *et al*. (2005). Here data were collected using Affymetrix GeneChips involving target samples comprising 3860 individual cRNAs of known sequence spiked-in at various concentrations. A total of six chips were collected. The first three representing control data (C), and the last three, spiked-in data (S). In total there was 14 010 probesets, of which 2535 represent genes spiked-in at equal 1× concentrations in both groups (the non-differentially expressing control genes), while 1331 represent genes spiked-in at higher levels in the S group (the differentially expressing genes). The remaining 11 475 probesets are considered 'empty' and represent non-differentially expressing genes. The complex nature of the data requires elaborate normalization. However, rather than attempting our own normalization of the data, we used the top 10 normalized datasets discussed in Choe *et al*. (2005). These are available at http://www.elwood9.net/spike. Careful analysis revealed that the first five and last five of these datasets have similar mean and standard deviations within their respective groups, but are dissimilar across groups. This is because a median polish summarization method was used for the first five datasets, whereas MAS 5.0 summarization was used for the last five datasets [see Choe *et al*. (2005) for details]. Therefore, when analyzing performance we considered the two groups separately. We refer to these two groups as Spike-in (i) and Spike-in (ii), respectively.

Table 1. Performance of different tests

Simulation	FP	FN	Miss
Synthetic (i)			
t-CART	293.3	296.6	589.9
t-CARThybrid[a]	—	—	—
Δh	502.4	505.8	1008.1
t-test	341.1	344.4	685.5
t-log	339.1	342.5	681.6
Cyber-T[a]	—	—	—
Synthetic (ii)			
t-CART	472.3	470.5	942.9
t-CARThybrid	481.5	479.7	961.2
Δh	536.9	535.1	1072.0
t-test	540.9	539.1	1080.0
t-log	542.4	540.5	1082.9
Cyber-T	485.9	484.1	970.0
Spike-in (i)			
t-CART	493.0	423.0	916.0
t-CARThybrid	400.4	330.4	730.8
Δh	1105.2	1035.2	2140.4
t-test	454.0	384.0	838.0
t-log	453.4	383.4	836.8
Cyber-T	380.8	310.8	691.6
Spike-in (ii)			
t-CART	498.0	428.0	926.0
t-CARThybrid	453.8	383.8	837.6
Δh	881.0	811.0	1692.0
t-test	521.4	451.4	972.8
t-log	502.0	432.0	934.0
Cyber-T	421.0	351.0	772.0

[a]Not defined under equal variance assumption.

4.3 Results

Table 1 tabulates the results from our experiments. The table records the total number of false positives (FP), the total number of false negatives (FN) and the total number of misclassifications (Miss) for each procedure. Since each of our procedures have different properties, and hence have different cutoff values for identifying differentially expressing genes, the values reported are based on the top 20% and top 10% of genes ranked by absolute value of the test statistic for the synthetic and spike-in datasets respectively. The values reported in Table 1 for Synthetic (i) and (ii) are averages (rounded off) over 100 independent replications.

Except for Synthetic (i), all results were based on tests assuming unequal variances across groups. In particular, CART unequal variance tests used the method of Section 3.3 (the hybrid test being defined suitably), whereas standard *t*-tests used (1). For Synthetic (i), all tests assumed an equal variance model (Cyber-T and the hybrid CART test do not apply here). In this case, regularized CART *t*-tests used (5) with $\tau_{\mathrm{j}} = 1$, whereas standard *t*-tests were based on (2).

The conclusions from Table 1 are very interesting. These are listed as follows:

(1) For the synthetic datasets, the Δh method, corresponding to vsn(), is nearly the worst performer. This was surprising to us, since this is a parametric procedure specifically designed to estimate mean–variance relationships of the form described by (7) and (8). We found that as we increased ρ towards the value of 1 (the standard deviation of $\varepsilon_{i,j}$), the performance of $\Delta\ h$ improved, but as one of our referees pointed out, such large values of ρ are unrealistic in practice. Moreover, when applied to the spike-in datasets, performance of Δh is also bad. Generally, our results do not favor Δh.

(2) The CART procedures are best for the synthetic datasets. The fact that they are better than a parametric method specifically designed for this example is highly promising. Over the spike-in data, the hybrid CART procedure does very well. In fact, of all procedures it strikes the best balance in all examples. Using both local and cluster wide information appears to be a robust form of regularization.

(3) Cyber-T has the best performance over the spike-in data. This fact was also noted in Choe *et al.* (2005). The large number of genes with low expression in this experiment seems to favor Cyber-T's method for pooling information. However, for the synthetic data, the method does not perform quite as well.

(4) The *t*-tests have average performance in all examples. It is clear that careful use of regularization can lead to significant improvement in their performance.

(5) It is interesting to note how all procedures are worse in the unequal variance Synthetic (ii) simulation compared with the equal variance Synthetic (i) simulation. This shows the tremendous loss in power in small sample sizes which occurs if we cannot tap into an equal variance assumption.

ACKNOWLEDGEMENTS

The authors thank the Executive Editor, Alfonso Valencia, the Associate Editor, and two anonymous referees for their work in reviewing the paper. Hemant Ishwaran was supported by NSF grant DMS-0405675. Funding to pay the Open Access publication charges for this article was provided by the Cleveland Clinic.

Conflict of Interest: none declared.

REFERENCES

Affymetrix Microarray Suite User Guide (2001) *Version 5.0, Affymetrix*, Santa Clara, 2001.

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

Blalock,E.M. *et al.* (2003) Gene microarrays in hippocampal aging: statistical profiling identifies novel process correlated with cognitive impairment. *J. Neuroscience*, **23**, 3807–3819.

Breiman,L. *et al.* (1984) *Classification and Regression Trees*. Wadsworth, Belmont, California.

Choe,S.E. *et al.* (2005) Preferred analysis methods for Affymetrix GeneChips revealed by a wholly defined control dataset. *Genome Biology*, **6**, R16.

Cui,X. *et al.* (2005) Improved statistical tests for differential gene expression by shrinking variance components estimates. *Biostatistics*, **6**, 59–75.

Durbin,B. *et al.* (2002) A variance-stabilizing transformation for gene-expression microarray data. *Bioinformatics*, **18**, S105–S110.

Efron,B. *et al.* (2001) Empirical Bayes analysis of a microarray experiment. *J. Am. Stat. Assoc.*, **96**, 1151–1160.

Gentleman,R., Carey,V., Huber,W., Irizarry,R. and Dudoit,S. (2005) *Bioinformatics and Computational Biology Solutions using R and Bioconductor*. Springer, New York.

Huber,W. *et al*. (2002) Variance stabilization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics*, **18**, S96–S104.

Irizarry,R.A. *et al*. (2003) Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, **4**, 249–264.

Ishwaran,H. and Rao,J.S. (2003) Detecting differentially expressed genes in microarrays using Bayesian model selection. *J*. *Am*. *Stat*. *Assoc*., **98**, 438–455.

Ishwaran,H. and Rao,J.S. (2005) Spike and slab gene selection for multigroup microarray data. *J*. *Amer*. *Stat*. *Assoc*., **100**, 764–780.

Ji,H. and Wong,W.H. (2005) TileMAp: create chromosomal map tiling array hybridizations. *Bioinformatics*, **21**, 3629–3636.

Pomeroy,S.L. *et al*. (2001) Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, **415**, 436–442.

Smyth,G.K. (2004) Linear models and empirical Bayes methods for assessing differential expression in microarray experiments. *Stat*. *Appl*. *Genet*. *Mol*. *Biol*., **3**, Article 3.

Student (1908), The probable error of a mean. *Biometrika*, **6**, 1–25.

Tusher,V.G. *et al*. (2001) Significance analysis of microarrays applied to the ionizing radiation response. *Proc*. *Natl Acad*. *Sci*. *USA*, **98**, 5116–5121.

Wright,G.W. and Simon,R.M. (2003) A random variance model for detection of differential gene expression in small microarray experiments. *Bioinformatics*, **19**, 2448–2455.