Biostatistics (2007), **8**, 2, *pp*. 212–227
doi:10.1093/biostatistics/kxl002
Advance Access publication on May 11, 2006

Averaged gene expressions for regression

MEE YOUNG PARK*

Google Inc., 1600 Amphitheatre Parkway, Mountain View, CA 94043, USA
meeyoung@google.com

TREVOR HASTIE

Department of Statistics and Department of Health Research & Policy,
Stanford University, CA 94305, USA

ROBERT TIBSHIRANI

Department of Health Research & Policy and Department of Statistics,
Stanford University, CA 94305, USA

Summary

Although averaging is a simple technique, it plays an important role in reducing variance. We use this essential property of averaging in regression of the DNA microarray data, which poses the challenge of having far more features than samples. In this paper, we introduce a two-step procedure that combines (1) hierarchical clustering and (2) Lasso. By averaging the genes within the clusters obtained from hierarchical clustering, we define supergenes and use them to fit regression models, thereby attaining concise interpretation and accuracy. Our methods are supported with theoretical justifications and demonstrated on simulated and real data sets.

Keywords: Averaging; Hierarchical clustering; Lasso; Variance reduction.

1. Introduction

In this paper, we present a way to improve the regression of gene expression measurements through coupling with the hierarchical clustering method. The DNA microarray data consist of several thousands of genes (predictors) and only a few hundreds of or less than a hundred experiments (observations). Since we focused on supervised methods, we assumed that there is also a response variable; a regression was performed using the continuous response, such as the survival time. Analyzing such data requires special treatment, in particular, overcoming the collinearity among the predictors, which results in large variance of the estimates and inaccurate prediction. We propose a simple yet efficient method of averaging to solve this problem. By averaging, we could also extract a subset of genes with essential predictive power and partition the subset into groups, within which the genes are coherent.

An overview of our method is as follows: We first applied hierarchical clustering (Eisen *and others*, 1998 or Chapter 14 of Hastie *and others*, 2001b) to the genes to obtain a dendrogram that reveals their

*To whom correspondence should be addressed.

nested correlation structure. At each level of the hierarchy, we created a unique set of genes and supergenes by computing the average expression of the current clusters. We then used the different sets of genes and supergenes as inputs for regression, in particular, Lasso (Tibshirani, 1996).

Hierarchical clustering is an especially attractive clustering method in our approach because it provides multiple levels at which the supergenes can be formed. Due to the simple fact that the Euclidean distance measure among the genes is a monotone function of their correlation (when the genes are properly standardized), hierarchical clustering provides flexibility in model selection in such a way that the genes are merged into supergenes in order of their correlation. Lasso yields a sparse fit; the useful property of automatic gene selection motivated us to present Lasso as an ideal procedure for regression. We achieved clearer interpretation and accuracy by combining an unsupervised method with a supervised method. As an alternative to hierarchical clustering, we may also use known facts about the gene association. We present an example of grouping the genes based on the Gene Ontology (GO) (available at http://www.geneontology.org and introduced in various publications, such as Ashburner *and others*, 2000; GO-Consortium, 2004) and using the principal components as supergenes. Although we did not significantly expand on this method, the GO database provides useful information that may be helpful in various applications.

Defining the average expression from a certain cluster to be a new feature, we in effect forced every component of the cluster to play the same role in prediction; in other words, all their coefficients were constrained to be the same. This restriction, depending on the correlation structure of the predictors, reduces the variance in prediction, and this idea has been explored in various settings. For instance, ridge regression (Hoerl and Kennard, 1970), by penalizing the size of the L_2 norm of the coefficients, allows the predictors with strong correlation to bear similar coefficients. In Hastie *and others* (2001a), the authors proposed the "tree harvesting" method that used $2p - 1$ (where p is the number of genes) average expression profiles from the hierarchical clustering dendrogram as potential features. In the forward selection stage, the model was sequentially augmented by adding a new univariate feature or an interaction term to the preexisting ones; in the following backward deletion stage, the feature causing the least improvement was removed, thereby generating an order for the final model selection. We used the same strategy of obtaining the averaged features from hierarchical clustering, but applied different methods to handle them in a model. Zou and Hastie (2005) proposed "elastic net," an automatic way to let the correlated, important variables have comparable coefficients and to leave out unimportant variables. The elastic net regression is a regularization scheme with a penalty that combined that of ridge regression and Lasso. Bair *and others* (2004) introduced the supervised principal component (SuperPC) method by which the principal component directions were found using only the predictors related to the outcome variable; through the principal component analysis, the correlated variables were automatically collected and their coefficients were constrained to be similar. Yu (2005) also suggested different approaches to forming overlapping/nonoverlapping gene clusters; the resulting information was used for providing groups of genes as predictors in regression.

In the following sections, we illustrate and support our approach in more detail with examples and justifications. We explore the use of averaged gene expressions for regression in Section 2 and illustrate the method with a microarray data example in Section 3. In Section 4, we present several experiments that used the GO. We conclude with a summary and possible extensions of our studies in Section 5. The appendix contains the proof.

2. Hierarchical clustering and averaging for regression

In this section, we investigate the hierarchical clustering and averaging method in regression settings. In particular, we focus on Lasso (Tibshirani, 1996; Efron *and others*, 2004 for algorithm), a regression method with L_1 penalization of the coefficients. Since we aim to fit a model for gene expression data, we assume that the number of predictors is large, although many are unimportant. Fitting a Lasso would shrink down the coefficients of noisy predictors, assigning nonzero coefficients to only the significant variables. We present a detailed description of the algorithm, which is followed by a justification of when and why our method works.

2.1 *Algorithm*

Let (x_i, y_i) for $i = 1, \ldots, n$ denote pairs of a gene expression profile ($x_i \in \mathbb{R}^p$) and the corresponding response variable ($y_i \in \mathbb{R}$). First we apply hierarchical clustering of the genes to yield their nested correlation structure. With p different levels of hierarchy, we create a unique set of genes and supergenes at each level by averaging the gene expressions within the current clusters. We regress y on every set of the predictors (genes and supergenes) using Lasso. For each fit of Lasso, we obtain a set of solution paths of the coefficients to which we usually apply cross-validation and select the optimal degree of shrinkage. Algorithm 1 summarizes the steps.

Algorithm 1: Hierarchical Clustering and Averaging for Regression

1. Apply hierarchical clustering of the genes to yield the nested correlation structure.
2. At each level of hierarchy, create supergenes by averaging the gene expressions at each cluster. This gives p different sets of genes and supergenes that represent each level.
3. For every set of the predictors (genes and supergenes), fit Lasso, using y as the response variable.
4. Using cross-validation, find the optimal degree of shrinkage and level of hierarchy.

2.2 *Two-way factor for goodness of the fit*

We have a two-way factor that affects the goodness of the fit; a bias-variance trade-off occurs as the granularity of clustering or the amount of shrinkage changes. While this bias-variance trade-off occurs, we observe a monotonicity in the minimizing criterion with respect to the two factors. Lasso minimizes the following loss + penalty measure:

$$\|y - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1. \tag{2.1}$$

Let us compare the following objective function values:

$$M_1 = \min_{\beta\in\mathbb{R}} \sum_{i=1}^{n} (y_i - \beta x_{1i} - \beta x_{2i})^2 + \lambda_1(2|\beta|), \tag{2.2}$$

$$M_2 = \min_{\beta_1,\beta_2\in\mathbb{R}} \sum_{i=1}^{n} (y_i - \beta_1 x_{1i} - \beta_2 x_{2i})^2 + \lambda_1(|\beta_1| + |\beta_2|), \tag{2.3}$$

$$M_3 = \min_{\beta_1,\beta_2\in\mathbb{R}} \sum_{i=1}^{n} (y_i - \beta_1 x_{1i} - \beta_2 x_{2i})^2 + \lambda_2(|\beta_1| + |\beta_2|). \tag{2.4}$$

It is easily seen that $M_1 \geqslant M_2$, while $M_2 \geqslant M_3$ for $\lambda_1 \geqslant \lambda_2$. This monotonicity will be demonstrated with data sets in Sections 2.5 and 3.

To find the most desired fit, we optimize the two parameters simultaneously by drawing a one-dimensional path on the two-dimensional space. If we start from the topmost level of hierarchy with λ large enough to shrink the single coefficient close to zero, move in a descending direction of the objective function, and end at the bottommost level with $\lambda = 0$, infinitely many paths can be drawn. Once the path is specified, we cross-validate along the curve, hoping that a near-optimal point will be found on the way. Finding a path so that either the correlation factor representing the hierarchical level increases (i.e. the height decreases) or the shrinkage factor (λ) decreases along the curve ensures that the path is searching in a descending direction. We define an appropriate path by adjusting the relative rate between the increment in the correlation factor and the decrement in the shrinkage factor.

2.3 *Improving in accuracy*

Our method is advantageous when there exist multiple variables with strong positive correlations. In fact, in the case of ordinary least-squares (OLS) method, if the sample correlations of the predictors are high enough, an averaged predictor yields the coefficient estimates with lower expected squared error than the raw predictors. The following theorem illustrates this fact.

THEOREM 2.1 Let $X_1, X_2, \ldots, X_m$, columns of $\mathbf{X}$, be predictors with the sample correlation structure, $\mathrm{corr}(X_j, X_k) = \rho > 0$ for $j \neq k$. Without loss of generality, assume that the predictors are standardized so that $\mathbf{X}^T\mathbf{X}$ is a symmetric matrix with the diagonal elements being 1 and all the off-diagonal elements being ρ. Let y be the response variable such that $y_i = \sum_{j=1}^{m} \beta_j X_{ji} + \epsilon_i$ where ϵ_i's are iid with mean 0 and variance σ^2. Let $\hat{\beta}$ be the OLS estimates of the coefficients

$$\hat{\beta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}y.$$

Let $\hat{\beta}^A$ be the OLS estimate of the coefficient when y is regressed on the sum of the m predictors. $\tilde{\beta}$ denotes the corresponding vector of estimates for the original predictors

$$\tilde{\beta} = (\hat{\beta}^A, \ldots, \hat{\beta}^A)^T,$$

$$\text{where } \hat{\beta}^A = \frac{\sum_{i=1}^{n} X_{.i}\, y_i}{\sum_{i=1}^{n} {X_{.i}}^2} \quad \text{and} \quad X_{.i} = \sum_{j=1}^{m} X_{ji}.$$

Then $E_{y|X}[(\tilde{\beta} - \beta)^T(\tilde{\beta} - \beta)] < E_{y|X}[(\hat{\beta} - \beta)^T(\hat{\beta} - \beta)]$ if and only if

$$\rho > 1 - \frac{\sigma^2}{\sum_{j=1}^{m}(\beta_j - \bar{\beta})^2/(m-1)}, \quad \text{where } \bar{\beta} = \sum_{j=1}^{m} \beta_j/m. \tag{2.5}$$

This theorem claims that if the true coefficients of the predictors are similar, thereby making the ratio $\sum_{j=1}^{m}(\beta_j - \bar{\beta})^2/\sigma^2$ small, then the range of ρ to improve the fit by averaging is large. Figure 1 illustrates the range. The curves represent different numbers of variables (m), and for each curve, we expect improvement in the upper left-hand region. Although $\tilde{\beta}$ yields a larger bias than $\hat{\beta}$, the former is a more accurate estimate due to a smaller variance.

The theorem can easily be generalized to a block-diagonal correlation structure. The average features within each block may yield a more accurate fit than the individual predictors.

2.4 *An example of underlying model*

We present a data structure for which the averaged predictors are the optimal features to regress on. Assume the following scenario: U_1 and U_2 are independent random variables; X_j are (fixed) linear functions of U_1 for m_1 different $j's$ and linear functions of U_2 for m_2 other $j's$, and y, the response, is a linear combination of U_1 and U_2. Both y and $\mathbf{X}$ are functions of $U's$, which are unknown. If σ_X^2 (the error variance

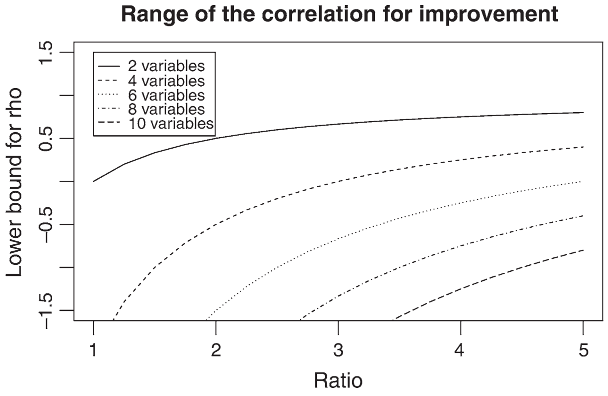

Fig. 1. If the true coefficients of the predictors are similar, thereby making the ratio $\sum_{j=1}^{m}(\beta_j - \bar{\beta})^2/\sigma^2$ small, the range of ρ to improve the fit by averaging is large. We expect improvement in the upper left-hand region.

Table 1. *Correlation matrix of the simulated data: the average sample correlations for all the pairs of the blocks*

Variable label	1–30	31–60	61–90	91–200
1–30	0.678	−0.060	0.047	0.000
31–60	−0.060	0.336	−0.022	0.001
61–90	0.047	−0.022	0.118	−0.003
91–200	0.000	0.001	−0.003	0.000

of **X**) is small enough, any linear combination $\sum_{j=1}^{m_1} w_j X_j$ (without loss of generality, assume that $w_j \geqslant 0$ and $\sum w_j = 1$) may be a reasonable predictor replacing U_1, and the same applies to $\sum_{j=m_1+1}^{m_1+m_2} w_j X_j$ for U_2. However, we must choose $w_j = 1/m_1$ ($w_j = 1/m_2$ for the latter) to minimize the variance of the weighted average. We present a realization of this scenario and its result through a simulation.

2.5 *Simulation*

Setting. A data set with $n = 100$ and $p = 200$ was generated according to the scheme described above. Among 200 predictors, 90 were significant; they were divided into three blocks (30 each), predictors in the three blocks being functions of the latent variables U_1, U_2, and U_3, respectively. The response variable y was a function of all three latent variables: $y = 2U_1 - 2U_2 + 3U_3 + \epsilon$.

This simulation yields the correlation structure shown in Table 1. Each entry of the matrix is the average sample correlation between two blocks. The average correlation within the first block was high enough that we expected all 30 components to be present in a tight cluster; however, the components in the third block would naturally be split into subgroups.

Result. Figure 2 is the dendrogram from the hierarchical clustering of the predictors based on the average linkage. The nodes belonging to the groups 1, 2, and 3 are labeled with 1, 2, and 3, respectively; the noisy variables are unlabeled. The horizontal dotted line (where 98 clusters are formed) indicates the optimal level selected based on the cross-validation. We next describe the cross-validation procedure.

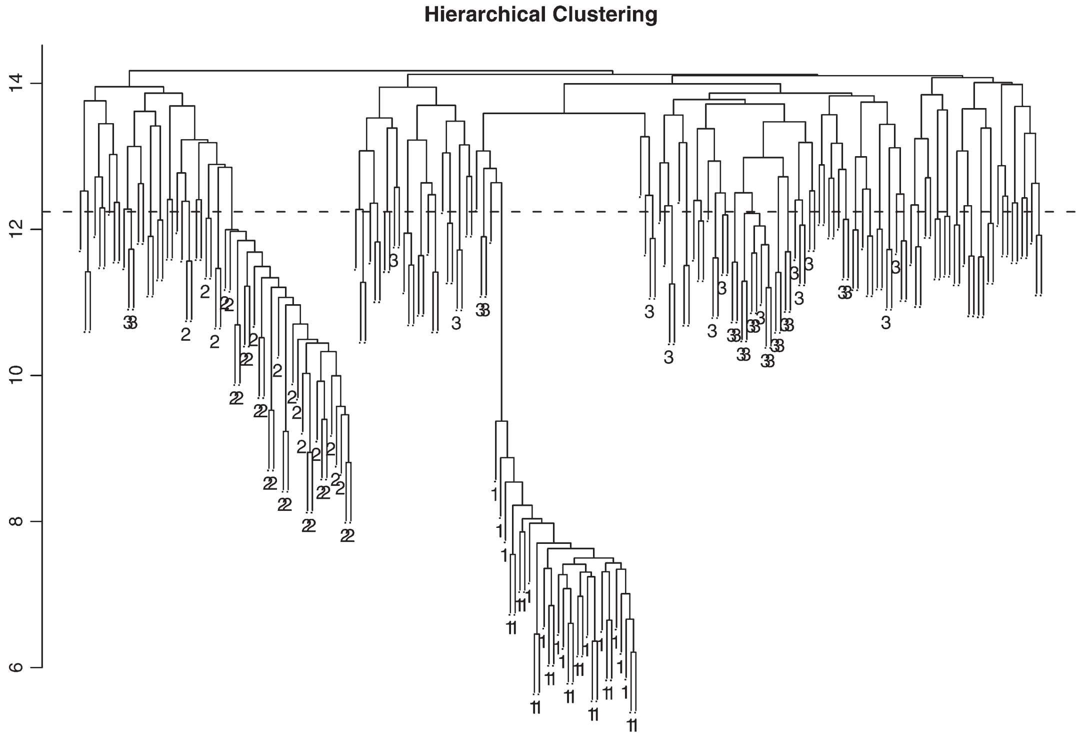

Fig. 2. The dendrogram from the hierarchical clustering of the predictors based on the average linkage. The labels represent the corresponding blocks. The horizontal dotted line (where 98 clusters are formed) indicates the optimal level selected based on the cross-validation.

The left-hand side of Figure 3 is the contour plot of the loss + penalty (2.1) values in terms of λ and the height. We used height^2, since it is linear in the corresponding correlation score. Six different paths were drawn on the plot so that $\lambda \propto (\text{height}^2)^d$ with $d = 0.5, 1, 1.5, 3, 5, 7$ from the left to the right, respectively. We noticed that the paths with larger d favored instances with higher levels of hierarchy. The contours of the cross-validation errors were illustrated on the same two-dimensional plane, on the right-hand side of Figure 3. From the contours, we anticipated that the optimal model would be selected somewhere along the paths with $d = 5, 7$. In practice, we would avoid assessing the cross-validation errors all over the plane, but restrict our search to the given paths. We have shown the level sets all over the plane for a clearer demonstration of our strategy.

The six solid curves in Figure 4 connect the cross-validation errors through the respective paths shown in Figure 3. The dotted curves connect the mean squared errors along the same set of paths. We found the minimum from all the values of the cross-validation errors shown on the plot and selected the optimal model (optimal λ and height) according to the one standard error rule. The chosen λ and the height are marked by a solid dot on both Figures 3 and 4.

Figure 5 presents the coefficient paths for the optimal models selected: the optimal Lasso fits on the averaged predictors and on the original predictors. The (dotted) vertical lines denote the chosen shrinkage levels. As for the fit on the averaged predictors, the three coefficients with relatively large absolute values were assigned to the three blocks with positive correlations; all 30 predictors from the first block, 27 from the second block, and only 7 from the third block were averaged to form the three major features. The other nonzero coefficients were for single or double predictors belonging to the third group. Few noisy

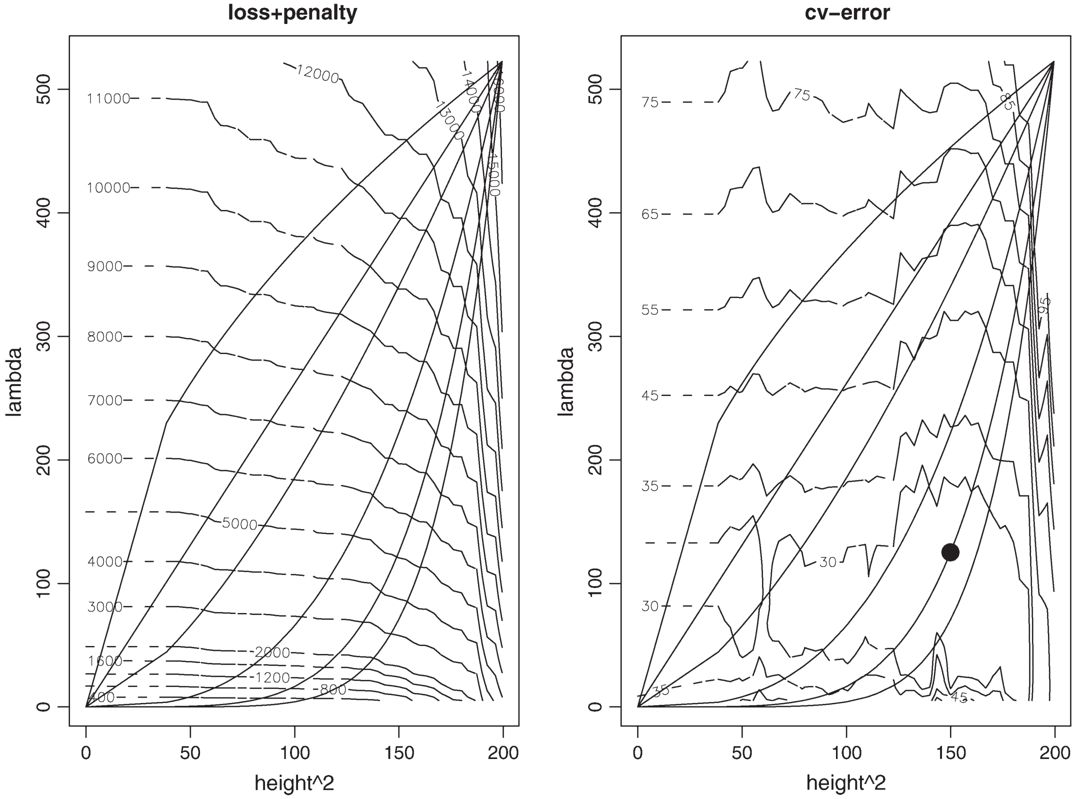

Fig. 3. (Left) The contours of the loss + penalty values in terms of λ and the height: Six different paths are drawn on the plot so that $\lambda \propto (\text{height}^2)^d$ with $d = 0.5, 1, 1.5, 3, 5, 7$ from the left to the right, respectively. (Right) The contours of the cross-validation errors illustrated on the same two-dimensional plane.

variables were in the active set. On the other hand, for the fit on the original variables, 45 coefficients were nonzero, 8 of which were assigned to noisy variables.

The results are summarized in Table 2. The optimal Lasso fit on the averaged predictors correctly identified 78 (30/27/21 from the three groups) out of 90 true predictors, and the 78 were grouped into 18 clusters; however, regular Lasso only recovered 37 (8/14/15 from the three groups) significant predictors. The comparison of R^2 and the P-values on the test data of size 200 shows that the fit on the averaged predictors not only yields a model that is more interpretable but also explains a much higher percentage of the variation in response.

3. MICROARRAY DATA EXAMPLE

In this section, we discuss the implementation of our method in a microarray data set. We used the data set of van't Veer *and others* (2002), which consists of 295 samples; we divided it into 147 for training and 148 for testing. van't Veer *and others* (2002) measured the gene expressions for 24 481 genes to analyze their relationship to the survival time/time to metastases of the breast cancer patients. We used the survival time recorded in 2004 as the response variable.

To reduce the number of genes to a reasonable size and to remove insignificant genes, we first fit 24 481 univariate Cox proportional hazards models and selected the 3017 most significant genes based on

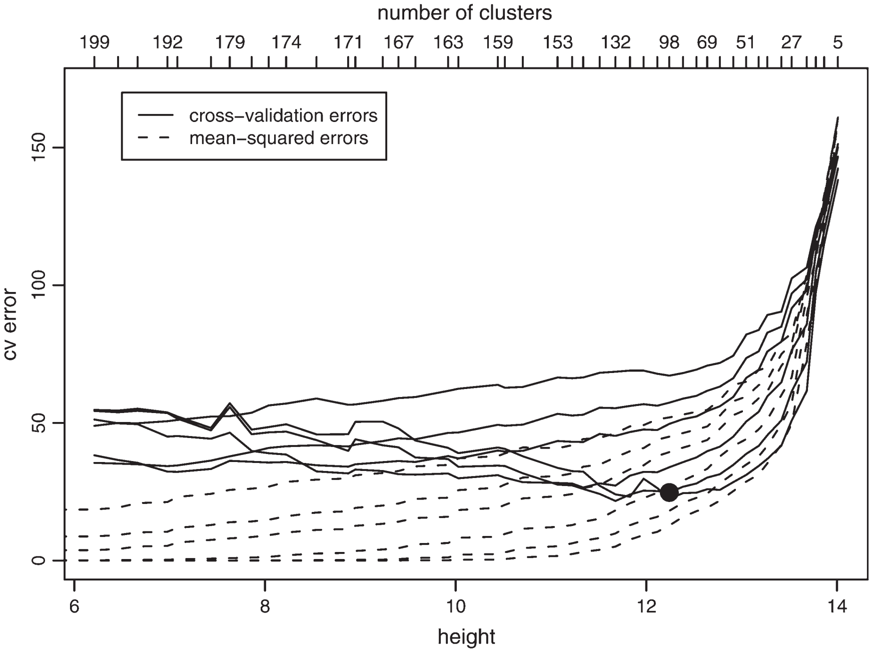

Fig. 4. The six solid curves connect the cross-validation errors through the paths shown in Figure 3. The dotted curves connect the mean squared errors along the same set of paths. We found the minimum from all the values of the cross-validation errors shown on the plot and selected the optimal model (optimal λ and height at the solid dot) according to the one standard error rule.

the rankings of their P-values. The choice of this cutoff P-value could be another parameter for the whole procedure, but here we did not tune it separately. Instead, we used a large cutoff value (0.15) so that we do not eliminate any features that are potentially significant.

After standardizing the genes, we applied hierarchical clustering and fit Lasso. As can be seen in Figure 6, the loss + penalty (with squared error loss) quantity has a monotone pattern in terms of both λ and the correlation score. We investigated five different paths, as indicated on the same contour plot: for these curves, $\lambda \propto (\text{height}^2)^d$ with $d = 2, 6, 7, 9, 13$.

In Figure 7, cross-validation errors are connected along each path. The dotted curves are the mean squared errors along the same five paths. Based on this display of the cross-validation errors, we found the minimum cross-validation error and applied one standard error rule; we selected the model that fits the clusters defined at the height of 1.30 with $\lambda = 6.57$. The optimal height and λ are marked by a solid dot on Figures 6 and 7.

We compared this model with the regular Lasso fit using 3017 individual genes and, in addition, tested the model by fitting the Cox proportional hazards model with test data. Table 3 summarizes the results. When tested by fitting the Cox proportional hazards model on 148 test samples, R^2 was larger, and the P-value was smaller for the fit with averaged genes than the original predictors.

In addition to applying the averaging strategy to Lasso, we did so to the Cox proportional hazards model with L_1 regularization as well. As proposed in Tibshirani (1997), we incorporated variable selection into the Cox model by maximizing the log partial likelihood subject to an L_1 norm constraint on the coefficients. We used the "coxpath" algorithm proposed by Park and Hastie (2006) and the cross-validation method proposed by Verweij and Van Houwelingen (1993) to choose an appropriate level of shrinkage. The third and the fourth rows of Table 3 refer to Cox Lasso with averaged genes and that with single

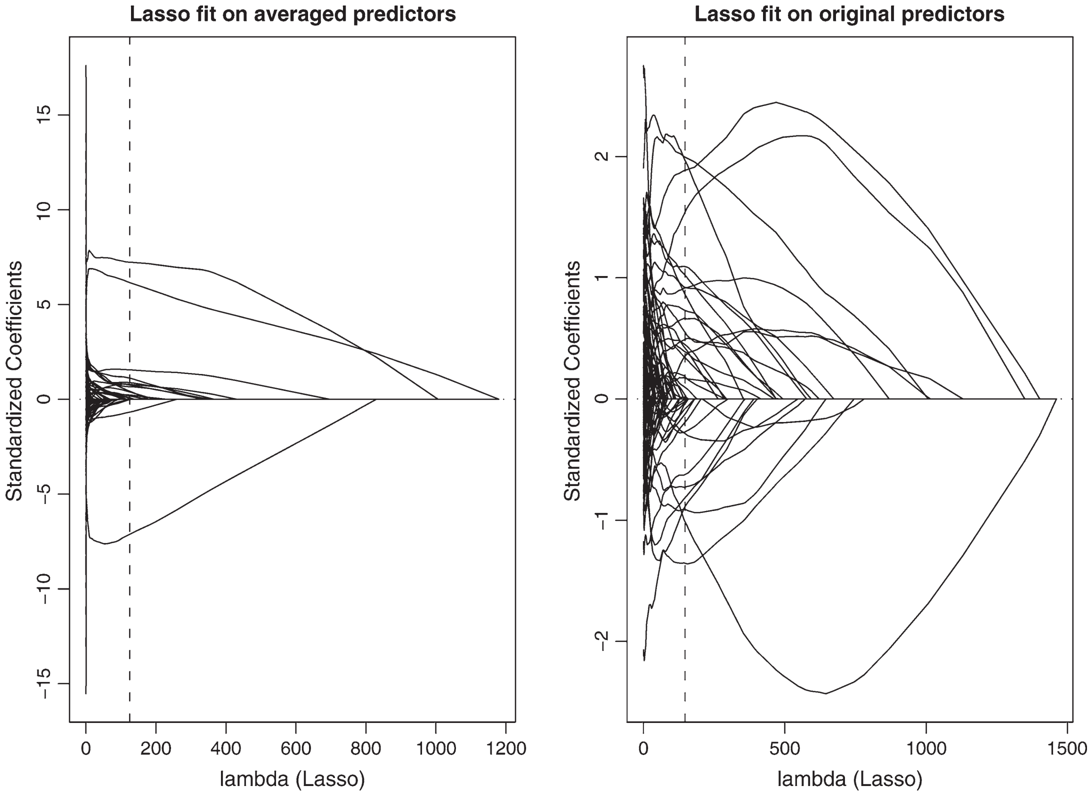

Fig. 5. The coefficient paths for the optimal models selected: (left) the optimal Lasso fit on the averaged predictors, and (right) the optimal fit on the original predictors. The (dotted) vertical lines denote the chosen shrinkage levels.

Table 2. *Comparison of optimal Lasso fits on the averaged and the original predictors*

Optimal Lasso fit on	Averaged predictors	Original predictors
Number of nonzero coefficients	18/98	45/200
Number of significant variables in the model	78/90	37/90
R^2 on the test data	0.866	0.171
P-value on the test data	7.1×10^{-08}	0.857

genes. As we incorporated the censor status and fit Cox models, Cox Lasso with averaged genes and that with individual genes performed slightly better than Lasso with averaged genes and that with original genes, respectively.

We also compared the performance to those of the tree harvesting (Hastie *and others*, 2001a), elastic net (Zou and Hastie, 2005), and SuperPC (Bair *and others*, 2004) methods. Although tree harvesting is similar to our method, its forward stepwise selection procedure searches over a large model space in a greedy manner compared to Lasso. It only selected one cluster of size 32, and augmenting the model further increased the cross-validated deviance score. Although tree harvesting performed comparable to our method, the number of gene clusters it might discover was limited because of its selection method which is less smooth. Elastic net, by adding the L_2 penalty term as a grouping device, showed an improvement

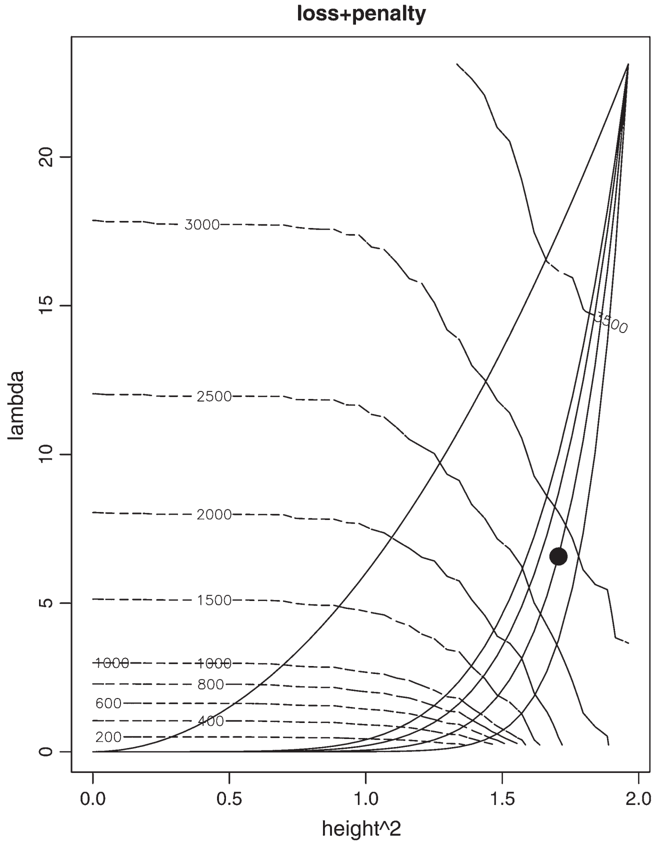

Fig. 6. The contour plot of loss + penalty (with squared error loss) values in terms of λ and the height: Five different paths are drawn on the plot so that $\lambda \propto (\text{height}^2)^d$ with $d = 2, 6, 7, 9, 13$ from the left to the right, respectively.

from Lasso with original predictors. The SuperPC method performed slightly better than our method, using two leading principal components. However, to discover gene clusters through the SuperPC analysis, one must make a strong assumption that all the groups form orthogonal directions with one another. If this assumption is not true, then the SuperPC model is likely to identify only a few of the groups.

Table 4 contains more information on the 17 supergenes in the active set of the optimal model. As a way to validate coherence among the genes forming each cluster, especially the large ones, we used the GO. The GO database (available at http://www.geneontology.org and introduced in various publications, such as Ashburner *and others*, 2000; GO-Consortium, 2004) contains information on genes or the interconnected gene products that form a directed acyclic graph (DAG) structure. There are three nonoverlapping ontologies: molecular function, biological process, and cellular component. Using a related device, Go-TermFinder (Boyle *and others*, 2004), we investigated whether each of our clusters is significantly associated with any node in GO, which in turn would imply that there is a common factor within the cluster. For a given cluster, Go-TermFinder gathers all the nodes in which the genes of our cluster are relatively abundant compared to the background of all the other genes. It computes (corrected) P-values based on the hypergeometric distribution for all the possible nodes. All the clusters in Table 4 that are larger than 13 were significantly abundant (Bonferroni corrected P-value < 0.05) in at least one of the GO annotations, although it was hard to expect a statistical significance for groups of size 2 or 3.

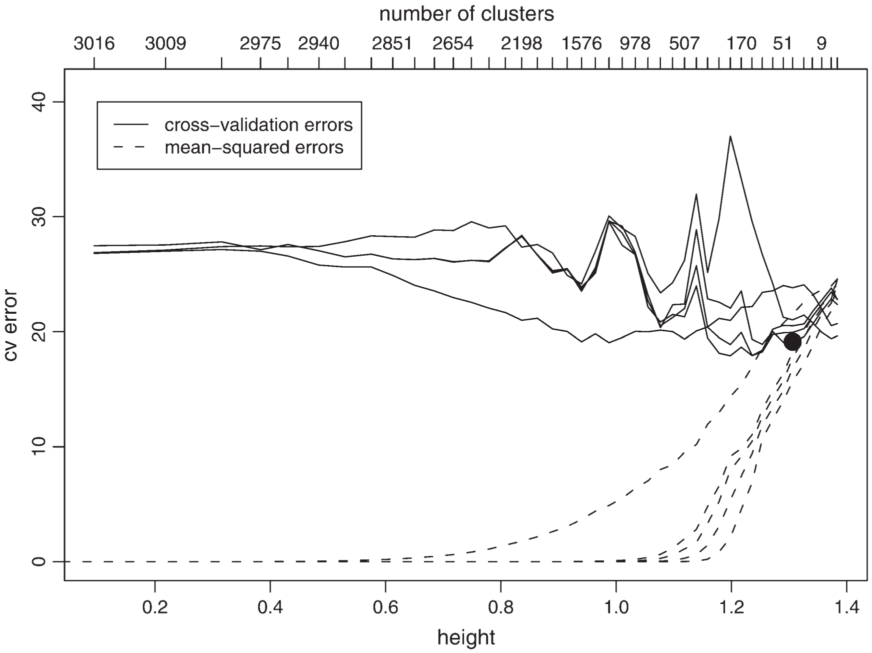

Fig. 7. The six solid curves connect the cross-validation errors through the respective paths shown in Figure 6. The dotted curves connect the mean squared errors along the same set of paths. We found the minimum from all the values of the cross-validation errors shown on the plot and selected the optimal model according to the one standard error rule (optimal $\lambda = 6.57$ and height = 1.30 at the solid dot).

Table 3. *Comparison of different methods*

	Nonzero coefficients	R^2	P-value
Lasso with averaged predictors	17/35	0.105	3.61×10^{-05}
Lasso with original predictors	62/3017	0.037	1.66×10^{-02}
Cox Lasso with averaged predictors	15/28	0.116	1.92×10^{-05}
Cox Lasso with original predictors	9/3017	0.086	2.71×10^{-04}
Tree harvesting	1/6033	0.116	1.96×10^{-05}
Elastic net ($\lambda_2 = 1$)	12/3017	0.056	2.35×10^{-04}
SuperPC	Two leading components	0.136	1.33×10^{-05}

Table 4. *The* 17 *features in the active set of the optimal Lasso fit with averaged genes. All the clusters that are larger than* 13 *were significantly abundant in at least one of the GO annotations*

Size	Coefficient	Size	Coefficient	Size	Coefficient
2	0.307	12	0.377	54	14.656
2	5.343	68	−8.099	12	−8.289
13	0.847	71	−3.154	53	6.943
42	−8.689	11	1.335	96	−0.894
10	−9.203	24	15.065	7	10.302
3	−3.761	9	5.039		

4. Using the GO database

We previously used hierarchical clustering under the assumption that the genes with highly correlated expression levels are likely to be involved in the same functions or processes. As an alternative to the hierarchical clustering method, we propose a more direct way of grouping, using the GO introduced in Section 3. We used the GO annotations to replace the clustering step in our original algorithm.

Using the GO database, we found all the clusters or the gene products representing a function, process or a cellular component, in which more than one of the genes in our data set were included. In an ideal situation, every gene in our list would have been annotated in GO, but it was not true. Thus, we illustrated our strategy using all the genes that belonged to at least one of the GO clusters. Since the genes in the predefined GO clusters were not necessarily highly (positively) correlated in their expression levels, we used the first principal component as a supergene instead of the average expression.

In this application, we fit regression models using the principal components from all the clusters simultaneously. Because the gene products in GO form a DAG structure instead of a nested structure, as in hierarchical clustering, we did not choose any one level of granularity. However, we filtered the clusters based on the following:

- Size of the clusters,
- Variation along the first principal component direction.

Once the principal components were achieved, we fit either a Lasso or a Cox proportional hazards model with variable selection.

Table 5 summarizes the number of genes in our data set that were found in three different ontologies, as well as the number of clusters to which they were assigned. The other rows of Table 5 show how many genes were left after the two-stage filtering and how many of the coefficients were nonzero in Lasso fit/Cox Lasso fit with penalization.

The performance of Lasso (the first row) and penalized Cox model (the second row) on the test data is summarized in Table 6 to provide a comparison. The third row contains the result from fitting Lasso on the union of the genes that belonged to the clusters that we used. The figures in Table 6 can be compared to the results in Table 3, Section 3. The first two fits using the clustering information from the GO (the first two rows) yield *P*-values that are comparable to the previous method of using hierarchical clustering. Lasso fits on several thousand individual genes (the third row) yield larger *P*-values than all other methods with grouping procedures.

Using the GO not only provided an automatic grouping of the genes but also revealed to which functions/processes/components the selected groups were associated with. Table 7 lists the eight GO terms from the biological process ontology whose corresponding principal components had nonzero coefficients

Table 5. *The number of genes in our data set that were found in different ontologies. Starting with row* 1*) the number of genes in our data set that were found in the GO database;* 2*) the number of unique clusters to which the genes were assigned;* 3*) the number of clusters left after the two-stage filtering;* 4*) the number of nonzero coefficients in Lasso fit;* 5*) the number of nonzero coefficients in Cox Lasso fit*

Ontologies	Function	Process	Component
Total number of genes included	14 802/24 481	13 874/24 481	10 241/24 481
Total number of clusters	2491	2916	555
Number of clusters after filtering	534/2491	625/2916	99/555
Nonzero Lasso coefficients	11/534	8/625	7/99
Nonzero Cox Lasso coefficients	7/534	11/625	7/99

Table 6. *R^2 and the P-values for the test set performance are listed.* 1) *Lasso fit,* 2) *L_1 penalized Cox model fit, and* 3) *Lasso fit on individual genes included in any clusters from the GO database*

Methods (number of genes used/24 481)		R^2	P-value
GO clusters—Lasso	Function	0.074	3.51×10^{-04}
	Process	0.101	6.49×10^{-05}
	Component	0.111	1.95×10^{-05}
GO clusters—Cox Lasso	Function	0.077	5.88×10^{-04}
	Process	0.082	3.87×10^{-04}
	Component	0.106	4.53×10^{-05}
Individual genes from the GO clusters	Function (4828)	0.064	0.00126
	Process (3769)	0.021	0.0722
	Component (1136)	0.041	0.0116

Table 7. *GO terms from the biological process ontology whose corresponding principal components had nonzero coefficients in the Lasso fit. The pairs of numbers in the size column indicate the number of components that our data contained and the total numbers of genes in the nodes of the GO*

GO term	Size	Function description
GO:0009267	3/116	Cellular response to starvation
GO:0016579	3/87	Protein deubiquitination
GO:0000076	5/57	DNA replication checkpoint
GO:0007064	2/48	Mitotic sister chromatid cohesion
GO:0051293	2/60	Establishment of spindle localization
GO:0015788	2/3	UDP-N-acetylglucosamine transport
GO:0006529	2/30	Asparagine biosynthesis
GO:0030330	4/19	DNA damage response, signal transduction by p53 class mediator

in the Lasso fit. The column labeled "size" contains pairs of numbers, indicating the number of components that our data contained and the total numbers of genes in the nodes of the GO. Lasso selected the clusters of rather small sizes, which means that the clusters of larger sizes contained many noisy genes, possibly assigning spurious loadings to the first principal component. Although the regression fits using the GO showed reasonable performances, it would have been more impressive if larger clusters had been chosen. The relationship among the genes contained in a node of the GO is reflected in the gene expression, for instance through high correlations among the elements; however, the correlations may be insignificant depending on the granularity of the node.

5. DISCUSSION

In this study, we introduced a simple, but effective, method of combining multiple variables into a representative feature and using the new feature in regression. We first used hierarchical clustering to obtain the sets of correlated variables; we averaged the variables within each cluster and input the averages as regressors to Lasso. When the variables were measured in comparable units and were positively correlated, their average was a strong feature, yielding a fit with lower variance than the individual variables.

The gene expression data often satisfy the conditions for improving the fit through averaging. Microarray data are composed of a large number of genes (predictors) that are often divided into blocks; within each block, the genes are highly correlated. These data are the main target of our technique.

In the following paragraphs, we describe several directions in which our method may be modified for wider applications.

Although we applied hierarchical clustering with the average linkage for all the analyses, two other popular choices are the complete linkage and the single linkage. As illustrated in Theorem 2.1, averaging reduces the variance but increases the bias of the coefficient estimates; the amount of change depends on the group sizes and the correlations. The complete linkage tends to yield smaller clusters with higher correlations among the elements compared to the single linkage; therefore, these different dissimilarity measures will cause a different amount of changes in variance and bias in resulting Lasso models. Complete linkage is favorable because it generates clusters with stronger correlations; but because of the small group sizes, the variance reduction might not be as large as in the case of the single linkage. We used the average linkage as a compromise.

All three dissimilarity measures of the hierarchical clustering method mentioned above only detect positively correlated elements or groups of elements. The dissimilarity may be characterized more generally so that the elements with highly negative correlations are merged simultaneously. However, before averaging two negatively correlated variables, one of them must be multiplied by -1; an analogous adjustment is necessary when averaging more than two variables with negative correlations.

Another way to modify the clustering scheme is to incorporate the relationship of the predictors with y in the dissimilarity measure. By adding a flavor of supervised learning, one can form the clusters considering the correlations of the predictors with y in addition to the correlations among the predictors.

We proposed using the averaged features as inputs for Lasso a regression method. The idea can be extended to other situations, such as classification and clustering. By averaging the initial variables, we can provide a smaller number of features with lower dimensions for classification and clustering.

In Section 4, we used the clustering result available in the GO rather than discovering the structure from the data. Similarly, other qualitative facts about the genes may replace or accompany the clustering step in our procedure.

Acknowledgments

Trevor Hastie was partially supported by grant DMS-0505676 from the National Science Foundation, and grant 2R01 CA 72028-07 from the National Institutes of Health. Robert Tibshirani was partially supported by National Science Foundation Grant DMS-9971405 and National Institutes of Health Contract N01-HV-28183. *Conflict of Interest:* None declared.

APPENDIX

A.1 *Proof of theorem*

It can be easily shown that $\tilde{\beta}$ is equivalent to the least-squares estimate of β when all m elements are constrained to be the same. Thus, defining an $m \times (m-1)$ matrix

$$J = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ -1 & 1 & \ddots & \vdots \\ 0 & -1 & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 \\ 0 & \cdots & 0 & -1 \end{pmatrix}$$

and $\tilde{\beta}(\lambda) = \text{argmin}_{\beta}(y - \mathbf{X}\beta)^T(y - \mathbf{X}\beta) + \lambda(\beta^T J)(\beta^T J)^T$,

$$\tilde{\beta} = \lim_{\lambda \to \infty} \tilde{\beta}(\lambda).$$

For any positive λ,

$$\begin{aligned} \beta(\lambda) &= (\mathbf{X}^T\mathbf{X} + \lambda J J^T)^{-1}\mathbf{X}^T y \\ &= (\mathbf{X}^T\mathbf{X} + \lambda J J^T)^{-1}(\mathbf{X}^T\mathbf{X})\hat{\beta} \\ &= (I - \lambda(\mathbf{X}^T\mathbf{X})^{-1}J(I + \lambda J^T(\mathbf{X}^T\mathbf{X})^{-1}J)^{-1}J^T)\hat{\beta}. \end{aligned}$$

Letting $\lambda \to \infty$,

$$\tilde{\beta} = (I - (\mathbf{X}^T\mathbf{X})^{-1}J(J^T(\mathbf{X}^T\mathbf{X})^{-1}J)^{-1}J^T)\hat{\beta}.$$

Letting $Z = (\mathbf{X}^T\mathbf{X})^{-1}J(J^T(\mathbf{X}^T\mathbf{X})^{-1}J)^{-1}J^T$,

$$\begin{aligned} E_{y|X}[(\tilde{\beta} - \beta)^T(\tilde{\beta} - \beta)] - E_{y|X}[(\hat{\beta} - \beta)^T(\hat{\beta} - \beta)] &= \sigma^2\text{trace}[(\mathbf{X}^T\mathbf{X})^{-1}(-2Z + Z^2)] + \beta^T Z^T Z\beta \\ &= -\sigma^2\frac{m-1}{1-\rho} + \sum_{j=1}^{m}(\beta_j - \bar{\beta})^2. \end{aligned}$$

The above equations imply that $E_{y|X}[(\tilde{\beta} - \beta)^T(\tilde{\beta} - \beta)] < E_{y|X}[(\hat{\beta} - \beta)^T(\hat{\beta} - \beta)]$ if and only if

$$\rho > 1 - \frac{\sigma^2}{\sum_{j=1}^{m}(\beta_j - \bar{\beta})^2/(m-1)}, \quad \text{where } \bar{\beta} = \sum_{j=1}^{m}\beta_j/m.$$

REFERENCES

ASHBURNER, M., BALL, C., BLAKE, J., BOTSTEIN, D., BUTLER, H., CHERRY, J., DAVIS, A., DOLINSKI, K., DWIGHT, S., EPPIG, J. *and others* (2000). Gene ontology: tool for the unification of biology. The gene ontology consortium. *Nature Genetics* **25**, 25–9.

BAIR, E., HASTIE, T., PAUL, D. AND TIBSHIRANI, R. (2004). Prediction by supervised principal components. *Journal of the American Statistical Association* **101**, 119–137.

BOYLE, E., WENG, S., GOLLUB, J., JIN, H., BOTSTEIN, D., CHERRY, J. AND SHERLOCK, G. (2004). Go::termfinder-open source software for accessing gene ontology information and finding significantly enriched gene ontology terms associated with a list of genes. *Bioinformatics* **20**, 3710–5.

EFRON, B., HASTIE, T., JOHNSTONE, I. AND TIBSHIRANI, R. (2004). Least angle regression. *Annals of Statistics* **32**, 407–99.

EISEN, M., SPELLMAN, P., BROWN, P. AND BOTSTEIN, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences of the United States of America* **95**, 14863–8.

GO-CONSORTIUM (2004). The gene ontology (go) database and informatics resource. *Nucleic Acids Research* **32**, 235–61.

HASTIE, T., TIBSHIRANI, R., BOTSTEIN, D. AND BROWN, P. (2001a). Supervised harvesting of expression trees. *Genome Biology* **2**, 1–12.

HASTIE, T., TIBSHIRANI, R. AND FRIEDMAN, J. (2001b). *Elements of Statistical Learning; Data Mining, Inference, and Prediction*. New York: Springer.

HOERL, A. E. AND KENNARD, R. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics* **12**, 55–67.

PARK, M. AND HASTIE, T. (2006). l_1 Regularization path algorithm for generalized linear models. *Technical Report*. Stanford University, Stanford.

TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B* **58**, 267–88.

TIBSHIRANI, R. (1997). The lasso method for variable selection in the Cox model. *Statistics in Medicine* **16**, 385–95.

VAN'T VEER, L., DAI, H., VAN DE VIJVER, M., HE, Y., HART, A., MAO, M., PETERSE, H., VAN DER KOOY, K., MARTON, M., WITTEVEEN, A. *and others* (2002). Gene expression profiling predicts clinical outcomes of breast cancer. *Nature* **415**, 530–6.

VERWEIJ, P. AND VAN HOUWELINGEN, H. (1993). Cross-validation in survival analysis. *Statistics in Medicine* **12**, 2305–14.

YU, X. (2005). Regression methods for microarray data, [PhD. Thesis]. Stanford University, Stanford. pp 23–39.

ZOU, H. AND HASTIE, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B* **67**, 301–20.

[*Received January 3, 2006; revised April 27, 2006; accepted for publication May 8, 2006*]