Prediction by Supervised Principal Components

Eric BAIR, Trevor HASTIE, Debashis PAUL, and Robert TIBSHIRANI

In regression problems where the number of predictors greatly exceeds the number of observations, conventional regression techniques may produce unsatisfactory results. We describe a technique called supervised principal components that can be applied to this type of problem. Supervised principal components is similar to conventional principal components analysis except that it uses a subset of the predictors selected based on their association with the outcome. Supervised principal components can be applied to regression and generalized regression problems, such as survival analysis. It compares favorably to other techniques for this type of problem, and can also account for the effects of other covariates and help identify which predictor variables are most important. We also provide asymptotic consistency results to help support our empirical findings. These methods could become important tools for DNA microarray data, where they may be used to more accurately diagnose and treat cancer.

KEY WORDS: Gene expression; Microarray; Regression; Survival analysis.

1. INTRODUCTION

In this article we study a method for predicting an outcome variable Y from a set of predictor variables $X_1, X_2, \ldots, X_p$, measured on each of N individuals. In the typical scenario that we have in mind, the number of measurements p is much larger than N. In the example that motivated our work, $X_1, X_2, \ldots, X_p$ are gene expression measurements from DNA microarrays. The outcome Y might be a quantitative variable that we might assume to be normally distributed. More commonly in microarray studies, Y is a survival time, subject to censoring.

One approach to this kind of problem would be a supervised prediction method. For example, we could use a form of regression applicable when $p > N$; partial least squares (Wold 1975) would be one reasonable choice, as would ridge regression (Hoerl and Kennard 1970). However, Figure 1 illustrates why a semisupervised approach may be more effective.

We imagine that there are two cell types, and that patients with the good cell (2) type live longer on the average. However, there is considerable overlap in the two sets of survival times. We might think of survival time as a "noisy surrogate" for cell type. A fully supervised approach would give the most weight to those genes having the strongest relationship with survival. These genes are partially, but not perfectly, related to cell type. If we could instead discover the underlying cell types of the patients (often reflected by a sizeable signature of genes acting together in pathways), then we would do a better job of predicting patient survival.

Now we can extract information about important cell types from both the relationship between Y and $X_1, X_2, \ldots, X_p$ and the correlation among the predictors themselves. Principal components analysis (PCA) is a standard method for modeling correlation. Regression on the first few principal components would seem like a natural approach, but this might not always work well. The fictitious data given in Figure 2 illustrate the problem (if we were to use only the largest principal component). This is a heatmap display with each gene represented by a row and each column containing data from one patient on one microarray. Gene expression is coded from blue (low) to yellow (high). In this example, the largest variation is seen in the genes marked A, with the second set of 10 patients having higher expression in these genes than the first 10. The set of genes marked B show different variation, with the second and fourth blocks of patients having higher expression in these genes. The remainder of the genes show no systematic variation. At the bottom of the display, the red points are the first two singular vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ (principal components) of the matrix of expression values. In microarray studies these are sometimes called "eigengenes" (Alter, Brown, and Botstein 2000). (The broken lines represent the "true" grouping mechanism that generated the data in the two groups.) Now if the genes in A are strongly related to the outcome Y, then Y will be highly correlated with the first principal component. In this instance we would expect a model that uses $\mathbf{u}_1$ to predict Y to be very effective. However, the variation in genes A might reflect some biological process that is unrelated to the outcome Y. In that case, Y might be more highly correlated with $\mathbf{u}_2$ or some higher-order principal component.

The supervised principal components technique that we describe in this article is designed to uncover such structure automatically. This technique was described in a biological setting by Bair and Tibshirani (2004) in the context of a related method known as "supervised clustering." The supervised principal component idea is simple: Rather than performing principal component analysis using all of the genes in a dataset, we use only those genes with the strongest estimated correlation with Y. In the scenario of Figure 2, if Y were highly correlated with the second principal component $\mathbf{u}_2$, then the genes in block B would have the highest correlation with Y. Hence we would compute the first principal component using just these genes, and this would yield $\mathbf{u}_2$.

As this example shows, using principal components helps uncover groups of genes that express together. Biologically, one or more cellular processes, accompanied by their cadre of expressing genes, determine the survival outcome. This same model underlies other approaches to supervised learning in microarray studies, including supervised gene shaving (Hastie et al. 2000)

Eric Bair is Post-Doctoral Fellow, Department of Statistics, Stanford University, Stanford, CA 94305, and Department of Neurology, UCSF (E-mail: *ebair@stat.stanford.edu*). Trevor Hastie is Professor, Departments of Statistics and Health, Research & Policy (E-mail: *hastie@stat.stanford.edu*), Debashis Paul is a Graduate Student, Department of Statistics (E-mail: *debashis@stat.stanford.edu*), and Robert Tibshirani is Professor, Department of Statistics and Health, Research & Policy (E-mail: *tibs@stat.stanford.edu*), Stanford University, Stanford, CA 94305. The authors thank the editor, two associate editors, and referees for suggestions that substantially improved this article. Bair was supported in part by a National Science Foundation graduate research fellowship. Tibshirani was supported in part by National Science Foundation grant DMS-99-71405 and National Institutes of Health contract N01-HV-28183. Hastie was supported in part by National Science Foundation grant DMS-02-04612 and National Institutes of Health grant 2R01 CA 72028-07.

Journal of the American Statistical Association
March 2006, Vol. 101, No. 473, Theory and Methods
DOI 10.1198/016214505000000628

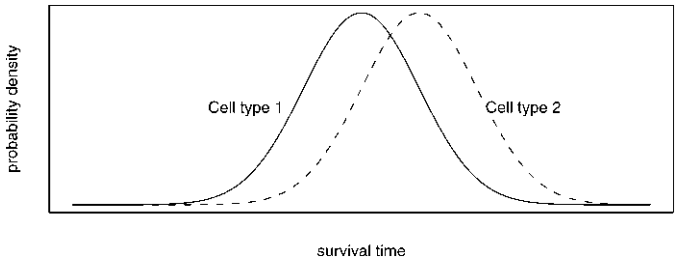

Figure 1. Underlying Conceptual Model. There are two cell types; patients with the good cell type live longer on average. However, there is considerable overlap in the two sets of survival times. Hence it could be advantageous to try to uncover the cell types and use these to predict survival time, rather than to predict survival time directly.

and tree harvesting (Hastie, Tibshirani, Botstein, and Brown 2001). The supervised principal components procedure can be viewed as a simple way to identify the clusters of relevant predictors by selection based on scores to remove the irrelevant sources of variation and application of principal components to identify the groups of coexpressing genes.

As far as we know, Bair and Tibshirani (2004) were the first to discuss the idea of supervised principal components in detail. But other authors have presented related ideas. Ghosh (2002) prescreened genes before extracting principal components, but seemed to do so for computational reasons. Jiang et al. (2004) used a similar idea in the context of merging the results from two different datasets. Nguyen and Rocke (2002) and Hi and Gui (2004) discussed partial least squares (PLS) approaches to survival prediction from microarray data. As we discuss in this article, this is a related but different method, and PLS did not perform as well as supervised principal components in our tests. PLS does not do an initial thresholding of features, and this is the key aspect of our procedure that underlies its good performance.

In the next section we define the supervised principal components procedure. Section 3 gives a brief summary of our consistency results, and Section 4 discusses an importance measure for individual features and a reduced model. Section 5 gives an example from a lymphoma study, Section 6 discusses alternative approaches to semisupervised prediction, including "gene shaving," and Section 7 presents a simulation study comparing the various methods. Section 8 summarizes the results of supervised principal components on some survival studies. Section 9 gives details of the theoretical results. The article concludes with some generalizations, including covariate adjustment and the use of unlabeled data in Section 10 and a discussion of limitations and future work in Section 11. The Appendix contains details of some proofs for Section 9.

2. SUPERVISED PRINCIPAL COMPONENTS

2.1 Description

We assume that there are p features measured on N observations (e.g., patients). Let $\mathbf{X}$ be an $N \times p$ matrix of feature measurements (e.g., genes), and let $\mathbf{y}$ be the N-vector of outcome measurements. We assume that the outcome is a quantitative variable; we discuss other types of outcomes, such as censored survival times. Here in a nutshell is the supervised principal component proposal:

1. Compute (univariate) standard regression coefficients for each feature.
2. Form a reduced data matrix consisting of only those features whose univariate coefficient exceeds a threshold θ in absolute value (θ is estimated by cross-validation).
3. Compute the first (or first few) principal components of the reduced data matrix.
4. Use these principal component(s) in a regression model to predict the outcome.

We now give details of the method. Assume that the columns of $\mathbf{X}$ (variables) have been centered to have mean 0. Write the singular value decomposition (SVD) of $\mathbf{X}$ as

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T, \tag{1}$$

where $\mathbf{U}, \mathbf{D}$, and $\mathbf{V}$ are $N \times m$, $m \times m$, and $m \times p$, and $m = \min(N-1, p)$ is the rank of $\mathbf{X}$. Here $\mathbf{D}$ is a diagonal matrix containing the singular values d_j; and the columns of $\mathbf{U}$ are the principal components $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m$; these are assumed to be ordered, so that $d_1 \geq d_2 \geq \cdots \geq d_m \geq 0$.

Let $\mathbf{s}$ be the p-vector of standardized regression coefficients for measuring the univariate effect of each gene separately on $\mathbf{y}$,

$$s_j = \frac{\mathbf{x}_j^T \mathbf{y}}{\|\mathbf{x}_j\|}, \tag{2}$$

with $\|\mathbf{x}_j\| = \sqrt{\mathbf{x}_j^T \mathbf{x}_j}$. Actually, a scale estimate $\hat{\sigma}$ is missing in each of the s_j's, but because it is common to all, we can omit it. Let C_θ be the collection of indices such that $|s_j| > \theta$. We denote by $\mathbf{X}_\theta$ the matrix consisting of the columns of $\mathbf{X}$ corresponding to C_θ. The SVD of $\mathbf{X}_\theta$ is

$$\mathbf{X}_\theta = \mathbf{U}_\theta \mathbf{D}_\theta \mathbf{V}_\theta^T. \tag{3}$$

Letting $\mathbf{U}_\theta = (\mathbf{u}_{\theta,1}, \mathbf{u}_{\theta,2}, \ldots, \mathbf{u}_{\theta,m})$, we call $\mathbf{u}_{\theta,1}$ the first supervised principal component of $\mathbf{X}$, and so on. We now fit a univariate linear regression model with response $\mathbf{y}$ and predictor $\mathbf{u}_{\theta,1}$,

$$\hat{\mathbf{y}}^{\mathrm{spc},\theta} = \bar{\mathbf{y}} + \hat{\gamma} \cdot \mathbf{u}_{\theta,1}. \tag{4}$$

Note that because $\mathbf{u}_{\theta,1}$ is a left singular vector of $\mathbf{X}_\theta$, it has mean 0 and unit norm. Hence $\hat{\gamma} = \mathbf{u}_{\theta,1}^T \mathbf{y}$, and the intercept is $\bar{\mathbf{y}}$, the mean of $\mathbf{y}$ (expanded here as a vector of such means).

We use cross-validation of the log-likelihood (or log partial-likelihood) ratio statistic to estimate the best value of θ. In most examples in this article we consider only the first supervised principal component; in the examples of Section 8, we allow the possibility of using more than one component.

Note that, from (3),

$$\begin{aligned} \mathbf{U}_\theta &= \mathbf{X}_\theta \mathbf{V}_\theta \mathbf{D}_\theta^{-1} \\ &= \mathbf{X}_\theta \mathbf{W}_\theta. \end{aligned} \tag{5}$$

So, for example, $\mathbf{u}_{\theta,1}$ is a linear combination of the columns of $\mathbf{X}_\theta$: $\mathbf{u}_{\theta,1} = \mathbf{X}_\theta \mathbf{w}_{\theta,1}$. Hence our linear regression model estimate can be viewed as a restricted linear model estimate using *all* of the predictors in $\mathbf{X}_\theta$,

$$\hat{\mathbf{y}}^{\mathrm{spc},\theta} = \bar{\mathbf{y}} + \hat{\gamma} \cdot \mathbf{X}_\theta \mathbf{w}_{\theta,1} \tag{6}$$

$$= \bar{\mathbf{y}} + \mathbf{X}_\theta \hat{\boldsymbol{\beta}}_\theta, \tag{7}$$

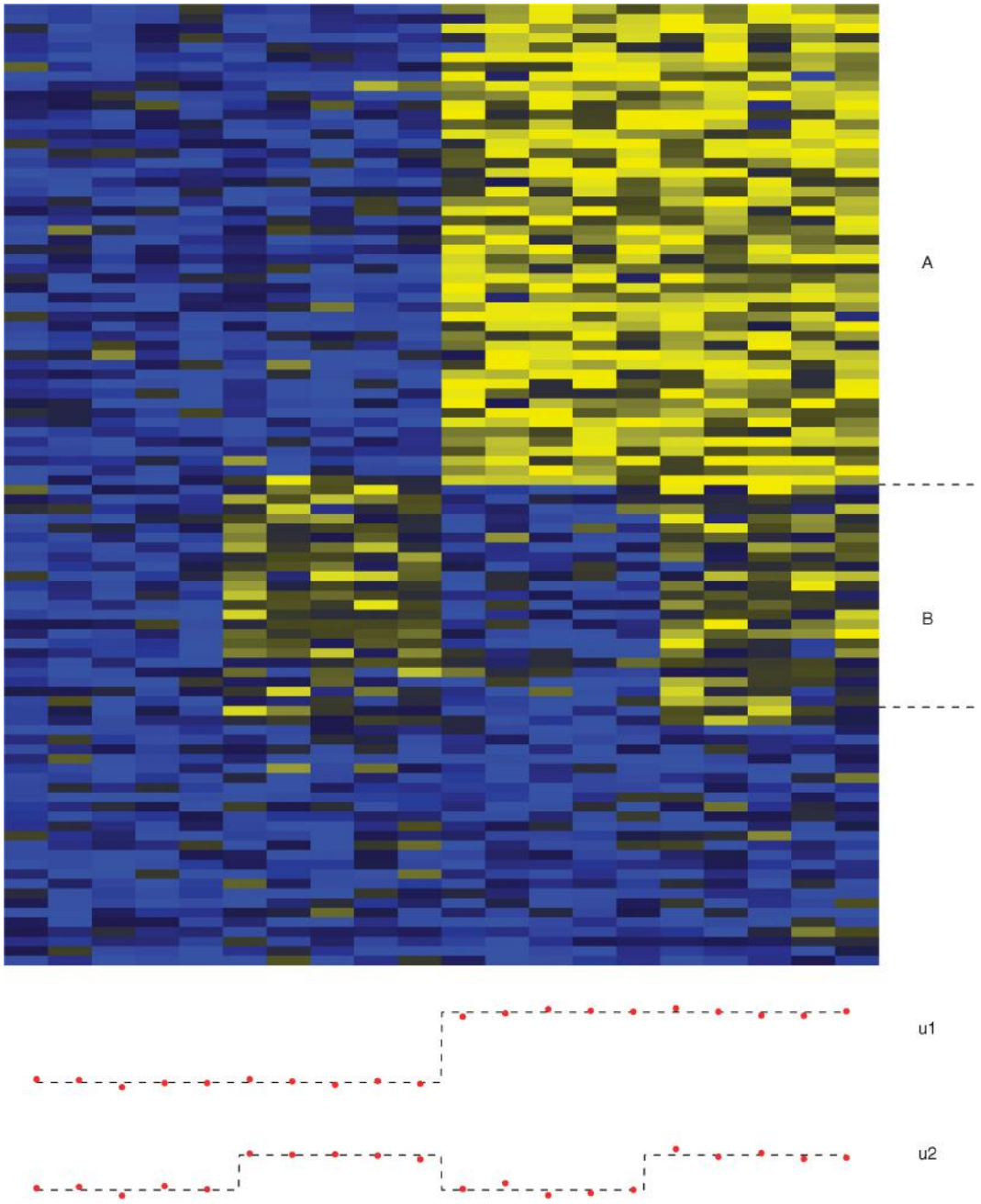

Figure 2. Fictitious Microarray Data for Illustration. A heatmap display with each gene represented by a row, and each column giving the data from one patient on one microarray. Gene expression is coded from blue (low) to yellow (high). The largest variation is seen in the genes marked A, with the second set of 10 patients having higher expression in these genes. The set of genes marked B show different variation, with the second and fourth blocks of patients having higher expression in these genes. At the bottom of the display are shown the first two singular vectors (principal components) of the matrix of expression values (red points), and the actual grouping generators for the data (dashed lines). If the outcome is highly correlated with either principal component, then the supervised principal component technique will discover this.

where $\hat{\boldsymbol{\beta}}_\theta = \hat{\gamma}\mathbf{w}_{\theta,1}$. In fact, by padding $\mathbf{w}_{\theta,1}$ with 0's (corresponding to the genes excluded by C_θ), our estimate is linear in all p genes.

Given a test feature vector $\mathbf{x}^*$, we can make predictions from our regression model as follows:

1. Center each component of $\mathbf{x}^*$ using the means we derived on the training data, $\mathbf{x}_j^* \leftarrow \mathbf{x}_j^* - \bar{\mathbf{x}}_j$.
2. $\hat{\mathbf{y}}^* = \bar{\mathbf{y}} + \hat{\gamma} \cdot \mathbf{x}_\theta^{*T}\mathbf{w}_{\theta,1} = \bar{\mathbf{y}} + \mathbf{x}_\theta^{*T}\hat{\boldsymbol{\beta}}_\theta$,

where $\mathbf{x}_\theta^*$ is the appropriate subvector of $\mathbf{x}^*$.

In the case of uncorrelated predictors, it is easy to verify that the supervised principal components procedure has the desired behavior. It yields all predictors whose standardized univariate coefficients exceed θ in absolute value.

Our proposal is also applicable to generalized regression settings, for example, survival data, classification problems, or data typically analyzed by a generalized linear model. In these cases we use a score statistic in place of the standardized regression coefficients in (2) and use a proportional hazards or appropriate generalized regression in (4). Let $\ell_j(\beta)$ be the log-

likelihood or partial likelihood relating the data for a single predictor X_j and the outcome y, and let $U_j(\beta_0) = d\ell/d\beta|_{\beta=\beta_0}$ and $I_j(\beta_0) = -d^2\ell_j/d\beta^2|_{\beta=\beta_0}$. Then the score statistic for predictor j has the form

$$s_j = \frac{U_j(0)^2}{I_j(0)}. \tag{8}$$

Of course, for the Gaussian log-likelihood, this quantity is equivalent to the standardized regression coefficient (2).

One could consider iterating the supervised principal components procedure. Thus we would find features whose inner product with the current supervised principal components was largest, use those features to compute the new principal components, and so on. But this procedure will tend to converge to the usual (unsupervised) principal components, because there is nothing to keep it close to the outcome after the first step. An iterative procedure would make sense only if it were based on a criterion involving both the variance of the features and the goodness of fit to the outcome. We consider such a criterion in the next section, although we ultimately do not pursue it (for reasons given there).

2.2 An Underlying Model

We now consider a model to support the supervised principal components method. Suppose that we have a response variable Y that is related to an underlying latent variable U by a linear model,

$$Y = \beta_0 + \beta_1 U + \varepsilon. \tag{9}$$

In addition, we have expression measurements on a set of genes X_j indexed by $j \in \mathcal{P}$, for which

$$X_j = \alpha_{0j} + \alpha_{1j} U + \epsilon_j, \qquad j \in \mathcal{P}. \tag{10}$$

The errors ε and ϵ_j are assumed to have mean 0 and are independent of all other random variables in their respective models.

We also have many additional genes $\mathbf{X}_k$, $k \notin \mathcal{P}$, which are independent of U. We can think of U as a discrete or continuous aspect of a cell type, which we do not measure directly. $\mathcal{P}$ represents a set of genes comprising a pathway or process associated with this cell type, and the $\mathbf{X}_j$'s are noisy measurements of their gene expression. We would like to identify $\mathcal{P}$, estimate U, and hence fit the prediction model (9). This is a special case of a latent structure model or single-component factor analysis model (Mardia, Kent, and Bibby 1979).

The supervised principal components algorithm (SPCA) can be seen as a method for fitting this model:

1. The screening step estimates the set $\mathcal{P}$ by $\hat{\mathcal{P}} = C_\theta$.
2. Given $\hat{\mathcal{P}}$, the SVD of $\mathbf{X}_\theta$ estimates U in (10) by the largest principal component $\mathbf{u}_{\theta,1}$.
3. Finally, the regression fit (4) estimates (9).

Step 1 is natural, because on average the regression coefficient $S_j = \mathbf{X}_j^T\mathbf{Y}/\|\mathbf{X}_j\|$ is non-0 only if α_{1j} is non-0 (assuming that the genes have been centered). Hence this step should select the genes $j \in \mathcal{P}$. Step 2 is natural if we assume that the errors ϵ_j have a Gaussian distribution, with the same variance. In this case the SVD provides the maximum likelihood estimates for the single-factor model (Mardia et al. 1979). The regression in step 3 is an obvious final step.

In fact, given $\mathcal{P}$, the model defined by (9) and (10) is a special structured case of an *errors-in-variables* model (Miller 1986; Huffel and Lemmerling 2002). One could set up a joint optimization criterion,

$$\min_{\beta_0,\beta_1,\{\alpha_{0,j},\alpha_{1,j}\},u_1,\ldots,u_N} \frac{\sum_{i=1}^N (y_i - \beta_0 - \beta_1 u_i)^2}{\sigma_Y^2} + \sum_{j\in\mathcal{P}} \frac{\sum_{i=1}^N (x_{ij} - \alpha_{0j} - \alpha_{1j}u_i)^2}{\sigma_X^2}. \tag{11}$$

Then it is easy to show that (11) can be solved by an augmented and weighted SVD problem. In detail, we form the augmented data matrix

$$\mathbf{X}_a = (\mathbf{y} : \mathbf{X}), \tag{12}$$

and assign weight $\omega_1 = \sigma_X^2/\sigma_Y^2$ to the first column and weight $\omega_j = 1$ to the remaining columns. Then, with

$$\mathbf{v}_0 = \begin{pmatrix} \beta_0 \\ \alpha_{0j_1} \\ \vdots \\ \alpha_{0j_q} \end{pmatrix}, \qquad \mathbf{v}_1 = \begin{pmatrix} \beta_1 \\ \alpha_{1j_1} \\ \vdots \\ \alpha_{1j_q} \end{pmatrix}, \tag{13}$$

(with $q = |\mathcal{P}|$) the rank-1 *weighted* SVD $\mathbf{X}_a \approx \mathbf{1}\mathbf{v}_0^T + \mathbf{u}\mathbf{v}_1^T$ solves the optimization problem in (11). Although this approach might seem more principled than our two-step procedure, SPCA has a distinct advantage. Here $\hat{\mathbf{u}}_{\theta,1} = \mathbf{X}_\theta \mathbf{w}_{\theta,1}$, and hence it can be defined for future $\mathbf{x}^*$ data and used for predictions. In the errors-in-variables approach, $\hat{\mathbf{u}}_{\mathrm{EV}} = \mathbf{X}_A \mathbf{w}_{\mathrm{EV}}$, which involves $\mathbf{y}$ as well and leaves no obvious estimate for future data. We return to this model in Section 6.

This latent-variable model can be easily extended to accommodate multiple components $U_1, \ldots, U_m$. One way of doing this is to assume that

$$Y = \beta_0 + \sum_{m=1}^M \beta_m U_m + \varepsilon \tag{14}$$

and

$$X_j = \alpha_{0j} + \sum_{m=1}^M \alpha_{1jm} U_m + \epsilon_j, \qquad j \in \mathcal{P}. \tag{15}$$

Fitting this model proceeds as before, except now we extract M rather than one principal component from $\mathbf{X}_\theta$. We study this model more deeply in Section 9.

2.3 An Example

The SPCA model anticipates other sources of variation in the data, unrelated to the response. In fact these sources can be even stronger than those driving the response, to the extent that principal components would identify them first. By guiding the principal components, SPCA extracts the desired components.

We simulated data from a scenario like that of Figure 2. We used 1,000 genes and 40 samples, all with base error model being Gaussian with unit variance. We then defined the mean

vectors $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ as follows. We divide the samples into consecutive blocks of 10, denoted by the sets (a, b, c, d). Then

$$\mu_{1i} = \begin{cases} -2 & \text{if } i \in a \cup b \\ +2 & \text{otherwise} \end{cases} \tag{16}$$

and

$$\mu_{2i} = \begin{cases} -1 & \text{if } i \in a \cup c \\ +1 & \text{otherwise.} \end{cases} \tag{17}$$

The first 200 genes have mean structure $\boldsymbol{\mu}_1$,

$$x_{ij} = \mu_{1i} + \epsilon_{ij}, \qquad j = 1, \ldots, 200, i = 1, \ldots, 40. \tag{18}$$

The next 50 genes have mean structure $\boldsymbol{\mu}_2$,

$$x_{ij} = \mu_{2i} + \epsilon_{ij}, \qquad j = 201, \ldots, 250, i = 1, \ldots, 40. \tag{19}$$

In all cases $\epsilon_{ij} \sim \mathrm{N}(0, 1)$, which is also how the remaining 750 genes are defined. Finally, the outcome is generated as $y_i = \alpha \cdot \mu_{1_i} + (1 - \alpha) \cdot \mu_{2i} + \varepsilon_i$, where ε_i is $\mathrm{N}(0, 1)$. The first two principal components of $\mathbf{X}$ are approximately $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ (see Fig. 2).

We tried various values of $\alpha \in [0, 1]$, as shown in Figure 3. Plotted is the correlation of the supervised principal components predictor with an independent (test set) realization of $\mathbf{y}$ as θ in the screening process $|s_j| > \theta$ is varied. The number of genes surviving the screening is shown on the horizontal axis. The extreme right end of each plot represents standard principal components regression. When $\alpha = 0$, so that the outcome is correlated with the second principal component, supervised PC easily improves on principal components regression. When α reaches .5, the advantage disappears, but supervised PC does no worse than principal components regression.

3. CONSISTENCY OF SUPERVISED PRINCIPAL COMPONENTS

In Section 9 we show that the standard principal components regression is not consistent as the sample size and number of features grow, whereas supervised principal components is consistent under appropriate assumptions. Because the details are lengthy, we give a summary first and defer the full discussion until Section 9.

We consider a latent variable model of the form (9) and (10) for data with N samples and p features. We denote the full $N \times p$ feature matrix by $\mathbf{X}$, and the $N \times p_1$ block of $\mathbf{X}$ by $\mathbf{X}_1$, corresponding to the features $j \in \mathcal{P}$. We assume that as $N \to \infty$, $p/N \to \gamma \in (0, \infty)$ and $p_1/N \to 0$ "fast." Note that p and p_1 may be fixed or may approach ∞. Given this setup, we prove the following:

- Let $\tilde{U}$ be the leading principal component of $\mathbf{X}$ and let $\tilde{\beta}$ be the regression coefficient of Y on $\tilde{U}$. Then $\tilde{U}$ is not generally consistent for U, and likewise $\tilde{\beta}$ is not generally

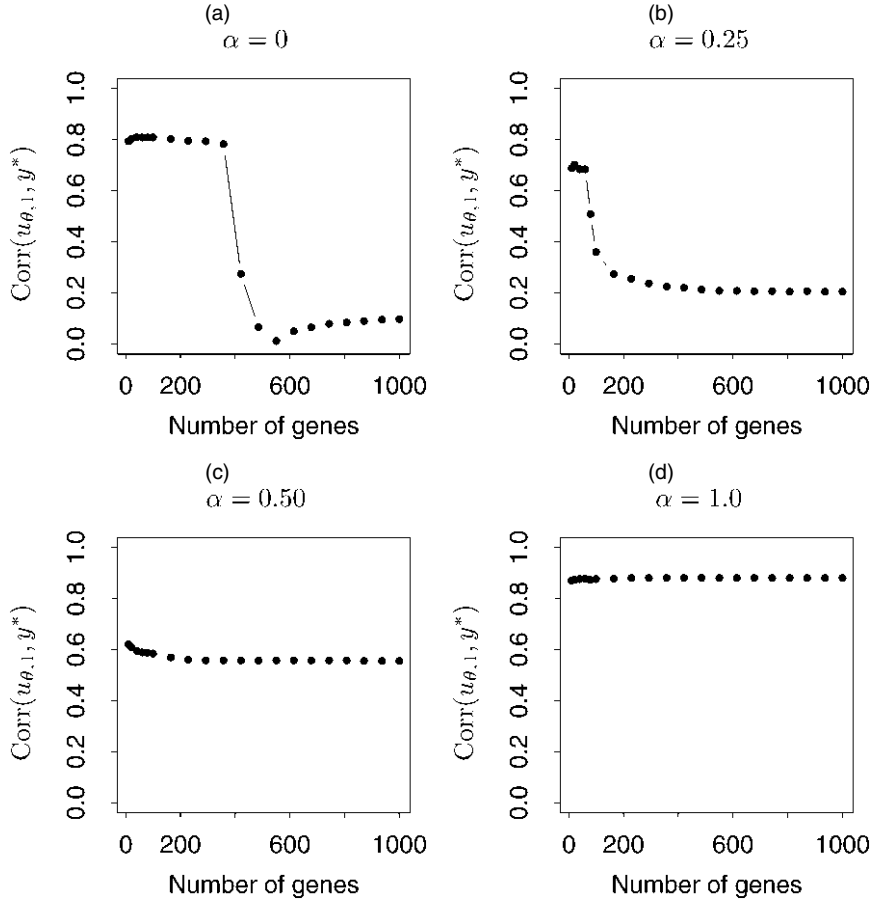

Figure 3. Correlation Between the First Supervised Principal Component $\mathbf{u}_{\theta,1}$ and a Test Outcome $\mathbf{y}$, as the Weight α Given to the First Principal Component in the Data Generation Is Varied. The number of genes used by the procedure is shown on the horizontal axis in each panel. The sharp switch (a) and (b) corresponds to the point at which the order of the principal components is reversed.

consistent for β. Because U is a random variable, in Section 9 we define what we mean by consistency.

- Assume that we are given $\mathbf{X}_1$. Then if $\hat{U}$ is the leading principal component of $\mathbf{X}_1$ and $\hat{\beta}$ is the regression coefficient of Y on $\hat{U}$, these are both consistent.
- If $\mathbf{X}_1$ is not given but is estimated by thresholding univariate features scores (as in the supervised principal components procedure), then the corresponding $\hat{U}$ and $\hat{\beta}$ are consistent.

We have also derived analogous results for Cox's proportional hazards model. Details are given in a technical paper available at *http://www-stat.stanford.edu/~tibs/spc/cox.ps* (or *pdf*).

4. IMPORTANCE SCORES AND A REDUCED PREDICTOR

Having derived the predictor $\mathbf{u}_{\theta,1}$, how do we assess the contributions of the p individual features? We define the *importance score* as the inner product between each feature and $\mathbf{u}_{\theta,1}$,

$$\text{imp}_j = \langle \mathbf{x}_j, \mathbf{u}_{\theta,1} \rangle. \tag{20}$$

Features j with large values of $|\text{imp}_j|$ contribute most to the prediction of $\mathbf{y}$. If the features are standardized, then this is just the correlation between each gene and the supervised principal component.

In some applications we would like to have a model that uses only a small number of features. For example, a predictor that requires expression measurements for a few thousand genes is not likely to be useful in a everyday clinical settings; microarrays are too expensive and complex for everyday use, and simpler assays like reversetranscription–polymerase chain reaction can measure only 50 or 100 genes at a time. In addition, isolation of a smaller gene set could aid biological understanding of the disease.

There are a number of ways to obtain a series of reduced models. One way would be to apply the lasso (Tibshirani 1996) to the data $(\mathbf{X}, \hat{\mathbf{y}}^{\text{spc}})$. The LAR algorithm (Efron, Hastie, Johnstone, and Tibshirani 2004) provides a convenient method for computing the lasso solutions. One drawback of this approach is that the series of models typically will involve different sets of features, which can be difficult for a scientist to assimilate.

Here we take a simpler approach. We define

$$\hat{\mathbf{u}}_{\text{red}} = \sum_{|\text{imp}_j| > \gamma} \ell_j \cdot \mathbf{x}_j, \tag{21}$$

where $\ell_j = \mathbf{u}_{\theta,1}^T \mathbf{x}_j / d_1$ is the loading for the jth feature and d_1 is the first singular value from the SVD (3). This predictor keeps only features with importance scores γ or larger, and weights these features by their loadings.

One could compute importance scores, and the corresponding reduced predictor, for all features (not just the ones used in computation of the supervised principal components). For example, there could be a feature not in the first set that has a higher inner product with the supervised principal component than a feature that is in the first set. However, we restrict attention to the features in the first set, for a couple of reasons. With this approach, a value of $\gamma = 0$ yields the original supervised principal components predictor, facilitating a comparison between the full and reduced models.

Second, allowing the reduced model to use features that are outside the first set leads naturally to an iterated version of the procedure in which we recompute the supervised principal component using genes with highest importance score, compute new scores and repeat. However, this procedure will typically converge to a usual first principal component (i.e., it is unsupervised). Hence we do not consider this iterated version, and restrict attention to genes that pass the initial threshold.

We illustrate this idea in the next section.

5. EXAMPLE: SURVIVAL OF LYMPHOMA PATIENTS

This dataset, from Rosenwald et al. (2002), consists of 240 samples from patients with diffuse large B-cell lymphoma (DLBCL), with gene expression measurements for 7,399 genes. The outcome was survival time, either observed or right-censored. We randomly divided the samples into a training set of size 160 and a test set of size 80. The results of various procedures are given in Table 1. We used the genes with top 25 Cox scores (cutoff of 3.53) in computing the first supervised principal component. Although PLS (described in Sec. 6) provides a strong predictor of survival, supervised principal components is even stronger.

Figure 4 shows the cross-validation curve for estimating the best threshold. Each model is trained, and then the log-likelihood ratio (LR) test statistic is computed on the left-out data. To have sufficient data in the left-out data to compute a meaningful LR statistic, we use two-fold cross-validation (rather than the more typical five- or ten-fold). This process is repeated five times and the results are averaged. In our experiments, this method yields a reasonable estimate of the best threshold but often underestimates the test set LR statistic (because the training and validation sets are half of the actual sizes). This is the case here, where the cross-validated LR statistic is just significant but the test set LR statistic is strongly significant.

This example also illustrates that the procedure can be sensitive to the threshold value. If we instead choose a threshold of 2 (a reasonable choice according to Fig. 4), then 865 genes are selected. The correlation of the resulting supervised principal component with the one found with 25 genes is only about .5. The supervised principal component predictor gives a p value of .02 in the test set; this significant but not as strong as that from the 25-gene predictor.

Figure 5 shows the test set log-likelihood ratio statistic obtained by fitting regression models of various sizes to the output of supervised principal component regression. We see that if the top few genes are used, then there is no loss in predictive power.

Figure 6 shows the top 25 genes and their loadings. Details are given in the figure caption.

Table 1. Lymphoma Data: Test Set Results for the Various Methods

Method	*Z-score*	*P value*
First principal component	−1.04	.2940
Partial least squares	2.51	.0112
First supervised principal component (25 genes)	−2.93	.0045

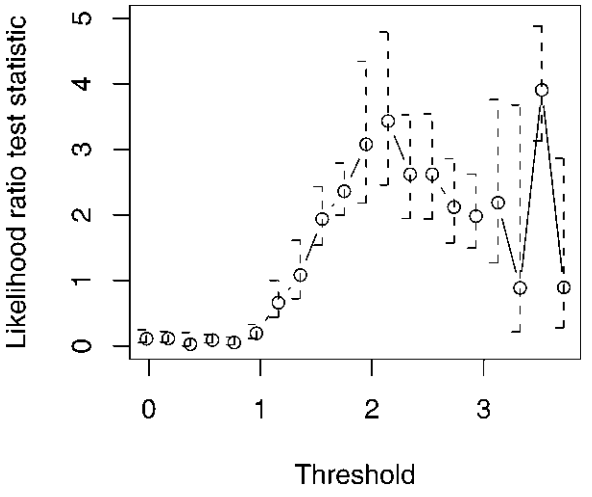

Figure 4. Lymphoma Data: Cross-Validation Curve for Estimating the Best Threshold.

6. SOME ALTERNATIVE APPROACHES

In this section we discuss some alternative approaches to this problem, some classical and some reflecting other approaches that we have explored.

6.1 Ridge Regression

Ridge regression (Hoerl and Kennard 1970) is a classical regression procedure when there are many correlated predictors, and one that could reasonably be applied in the present setting. Ridge regression fits the full linear regression model but manages the large number of predictors in these genomic settings by regularization (Hastie and Tibshirani 2003). Ridge regression solves

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \boldsymbol{\beta}_0 - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|^2, \tag{22}$$

where the second term shrinks the coefficients toward 0. The regularization parameter λ controls the amount of shrinkage, and for even the smallest $\lambda > 0$, the solution is defined and is unique. It can also be shown that this form of regularization shrinks the coefficients of strongly correlated predictors toward each other, an attractive property in this setting.

Using the singular value representation (1), the fitted values from a ridge regression have the form

$$\begin{aligned}\hat{\mathbf{y}}^{\mathrm{RR}} &= \bar{\mathbf{y}} + \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}\mathbf{y} \\ &= \bar{\mathbf{y}} + \sum_{j=1}^{m} \mathbf{u}_j \frac{d_j^2}{d_j^2 + \lambda}\mathbf{u}_j^T\mathbf{y}. \end{aligned} \tag{23}$$

Ridge regression is like a smooth version of principal components regression; rather than retaining the first k principal components and discarding the rest, it weights the successive components by a factor that decreases with decreasing eigenvalue d_j^2. Note that ridge regression is a linear method; that is, $\hat{\mathbf{y}}^{\mathrm{RR}}$ is a linear function of $\mathbf{y}$. In contrast, SPCA is nonlinear, because of the initial gene-selection step.

6.2 The Lasso

The lasso Tibshirani (1996) is a variation on ridge regression that solves

$$\min_{\boldsymbol{\beta}} \|\mathbf{y} - \boldsymbol{\beta}_0 - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda \sum_{j=1}^{p} |\beta_j|, \tag{24}$$

where the second term shrinks the coefficients toward 0. The absolute value form of the penalty has the attractive property that it can shrink some coefficients exactly to 0. The "basis pursuit" proposal of Chen, Donoho, and Saunders (1998) uses the same idea in a signal processing context. Computation of the lasso is more challenging that computation of ridge regression. Problem (24) is a convex optimization, which can be very difficult if the number of features p is large. The least-angle regression (LARS) algorithm (Efron et al. 2004) provides an efficient method for computation of the lasso, exploiting the fact that as λ changes, the profiles of the estimates are piecewise linear. For other likelihood-based models, such as the Cox model, the Euclidean distance in (22) is replaced by the (negative) log-likelihood or log-partial-likelihood. The coefficient profiles are not piecewise linear, so the LARS approach cannot be applied. Some details of the lasso for nonlinear models have been given by Tibshirani (1996, 1997).

When p is larger than the sample size N, the number of non-0 coefficients in a lasso solution is at most N (for any λ). Although sparse solutions are generally attractive, these solutions may be too sparse, because, for example, for microarray data they would allow only N genes to appear in a given model.

We now consider several approaches to supervised principal components that modify the optimization criterion behind principal components analysis in a supervisory fashion.

6.3 Partial Least Squares

PLS is one such approach, with a long history (Wold 1975; Frank and Friedman 1993; Hastie, Tibshirani, and Friedman 2001). PLS works as follows:

1. Standardize each of the variables to have mean 0 and unit norm, and compute the univariate regression coefficients $\mathbf{w} = \mathbf{X}^T\mathbf{y}$.
2. Define $\mathbf{u}_{\mathrm{PLS}} = \mathbf{X}\mathbf{w}$, and use it in a linear regression model with $\mathbf{y}$.

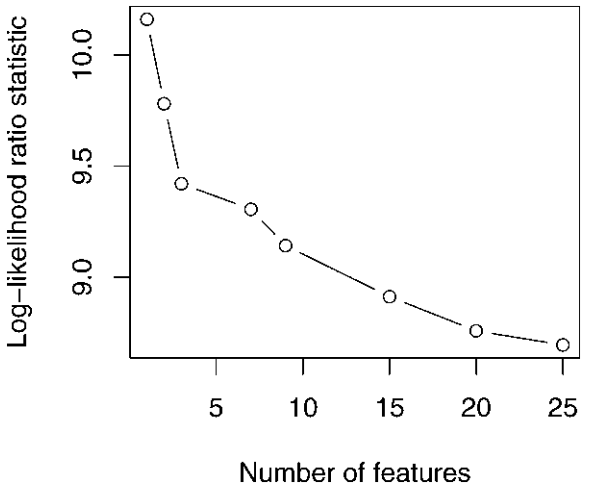

Figure 5. Lymphoma Data: Test Set Log-Likelihood Ratio Statistic Obtained From the Reduced Predictor Approximation.

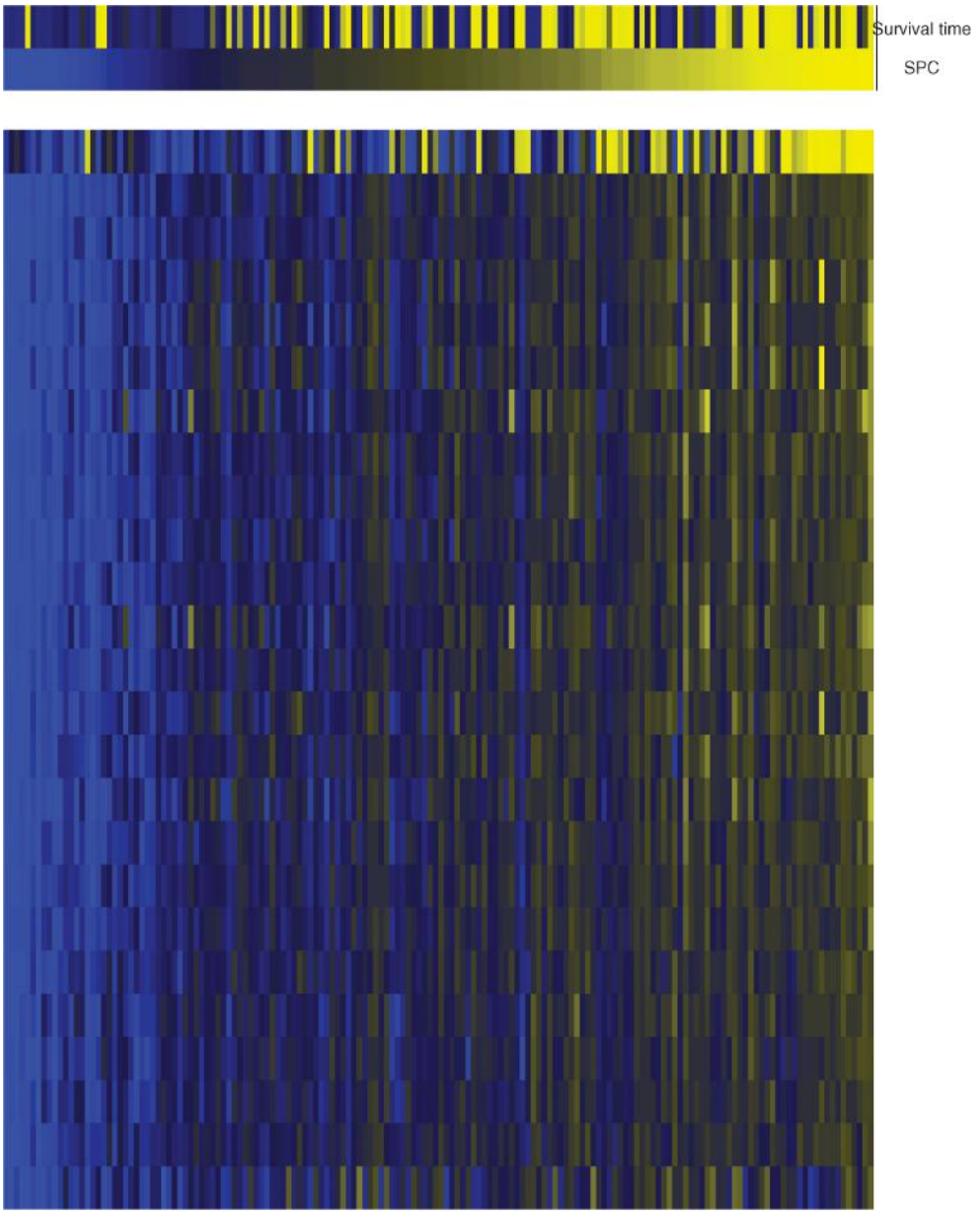

Figure 6. Lymphoma Data: Heatmap Display of the Top 25 Genes. The top two rows of the figure show the observed survival times and first supervised principal component (SPC) $\mathbf{u}_{\theta,1}$; for survival times T censored at time c, we show $\hat{E}(T \mid T \geq c)$ based on the Kaplan–Meier estimator. All columns have been sorted by increasing value of $\mathbf{u}_{\theta,1}$. On the right of the heatmap the "loadings" $w_{\theta,1}$ are shown [see (6)]; the genes (rows) are sorted by decreasing value of their loading. All genes but the last one have positive loadings.

Although PLS goes on to find subsequent orthogonal components, one component is sufficient for our purposes here. PLS explicitly uses $\mathbf{y}$ in estimating its latent variable. Interestingly, it can be shown that the (normalized) $\mathbf{w}$ in PLS solves (Frank and Friedman 1993)

$$\max_{\|\mathbf{w}\|=1} \operatorname{corr}^2(\mathbf{y}, \mathbf{X}\mathbf{w}) \operatorname{var}(\mathbf{X}\mathbf{w}), \tag{25}$$

a compromise between regression and PCA.

Frank and Friedman (1993) concluded that the variance term dominates, and hence that PLS would in general be similar to principal components regression. We can see this in the context of consider model (9)–(10). The expected values of the univariate regression coefficient w_j is

$$\mathrm{E}(w_j) = \beta_1 \sum_j \frac{\alpha_j}{\alpha_j^2 + \sigma_j^2}. \tag{26}$$

Now if $\sigma_j^2 = 0$, then the PLS direction $\sum_j w_j x_{ij}$ reduces to $\beta_1 \sum_j x_{ij}/\alpha_j$. But in that case, the latent factor U equals $\sum_j X_i/\alpha_j$, so the two solutions agree (in expectation).

Hence, after we isolate the block of important features, carrying out principal components regression or PLS is likely to give similar results. The main advantage of supervised principal components over the standard PLS procedure is the use of thresholding to estimate which features are important. PLS retains all features and can be adversely effected by the noise in

the unimportant features. We include PLS among the competitors in our comparisons in the next sections.

6.4 Mixed Variance–Covariance Criterion

The largest principal component is that normalized linear combination $\mathbf{z} = \mathbf{X}\mathbf{v}$ of the genes with the largest sample variance. Another way to supervise this would be to seek a linear combination $\mathbf{z} = \mathbf{X}\mathbf{v}$ having both large variance and a large (squared) covariance with $\mathbf{y}$, leading to the compromise criterion

$$\max_{\|\mathbf{v}\|=1} (1-\alpha)\,\mathrm{var}(\mathbf{z}) + \alpha\,\mathrm{cov}(\mathbf{z},\mathbf{y})^2, \quad \text{s.t. } \mathbf{z} = \mathbf{X}\mathbf{v}. \tag{27}$$

This is equivalent to

$$\max_{\|\mathbf{v}\|=1} (1-\alpha)\mathbf{v}^T\mathbf{X}^T\mathbf{X}\mathbf{v} + \alpha\mathbf{v}^T\mathbf{X}^T\mathbf{y}\mathbf{y}^T\mathbf{X}\mathbf{v}. \tag{28}$$

If $\mathbf{y}$ is normalized to unit norm, then the second term in (28) is a regression sum of squares (regressing $\mathbf{z}$ on $\mathbf{y}$) and has the interpretation "the variance of $\mathbf{z}$ explained by $\mathbf{y}$." The solution $\mathbf{v}$ can be efficiently computed as the first right singular vector of the augmented $(N+1) \times p$ matrix,

$$\mathbf{X}_a = \begin{pmatrix} (1-\alpha)^{1/2}\mathbf{X} \\ \alpha^{1/2}\mathbf{y}^T\mathbf{X} \end{pmatrix}. \tag{29}$$

By varying the mixing parameter α, we control the amount of supervision. Although the mixed criterion can guide the sequence of eigenvectors, all genes have non-0 loadings, which adds a lot of variance to the solution.

6.5 Supervised Gene Shaving

Hastie et al. (2000) proposed "gene shaving" as a method for clustering genes. The primary focus of their method was to find small clusters of highly correlated genes, whose average exhibited strong variance over the samples. They achieved this through an iterative procedure, which repeatedly computed the largest principal component of a subset of the genes, but after each iteration "shaved" away a fraction of the genes with small loadings. This produces a sequence of nested subsets of gene clusters, with successively stronger pairwise correlation and variance of the largest principal component.

They also proposed a supervised version of gene shaving, which uses precisely a mixed criterion of the form (28). Although this method has two tuning parameters, α and the subset size, here we fix α to the intermediate value of .5 and focus attention on the subset size. As in SPCA, for each subset the largest principal component is used to represent its genes.

This method is similar in flavor to SPCA; it produces principal components of subset of genes, where the choice of subset is supervised. Simultaneously searching for sparse components with high variance and correlation with $\mathbf{y}$ is an attempt to omit features that might slip through the SPCA screening step. Our experiments in the next section show that shaving can exhibit very similar performance to SPCA, the latter with the advantages of being simpler to define and having only one tuning parameter to select.

6.6 Another Mixed Criterion

The largest normalized principal component $\mathbf{u}_1$ is the largest eigenvector of $\mathbf{X}\mathbf{X}^T$. This follows easily from the SVD (1) and hence $\mathbf{X}\mathbf{X}^T = \mathbf{U}\mathbf{D}^2\mathbf{U}^T$. Intuitively, because

$$\mathbf{u}_1^T\mathbf{X}\mathbf{X}^T\mathbf{u}_1 = \sum_{j=1}^{p} \langle \mathbf{u}_1, \mathbf{x}_j \rangle^2, \tag{30}$$

we are seeking the vector $\mathbf{u}_1$ closest on average to each of the $\mathbf{x}_j$. A natural supervised modification is to perturb this criterion in a manner that encourages the leading eigenvector to align with $\mathbf{y}$,

$$\max_{\mathbf{u}_1, \|\mathbf{u}_1\|=1} (1-\alpha) \sum_{j=1}^{p} \langle \mathbf{u}_1, \mathbf{x}_j \rangle^2 + \alpha \langle \mathbf{u}_1, \mathbf{y} \rangle^2. \tag{31}$$

Solving (31) amounts to finding the largest eigenvector of

$$C(\mathbf{y};\alpha) = (1-\alpha)\mathbf{X}\mathbf{X}^T + \alpha\mathbf{y}\mathbf{y}^T. \tag{32}$$

Equivalently, one could form an augmented matrix $\mathbf{X}_a$ with $\mathbf{y}$ in the $(p+1)$st column. If we assign weights α to this row and $(1-\alpha)$ to the first p rows, then a weighted SVD of $\mathbf{X}_a$ is equivalent to an eigendecomposition of (31). We note that this is exactly the situation described in the errors-in-variables model (11)–(13) in Section 2.2. As mentioned there, the estimate $\mathbf{u}_1$ involves $\mathbf{y}$ as well as the $\mathbf{x}_j$, and so cannot be used directly with test data. We did not pursus this approach further.

6.7 Discussion of Methods

Figure 7 illustrates the methods discussed earlier on a simulation example with $N = 100$ samples and $p = 5{,}000$ features. The data are generated according to the latent variable model (35), where there are four dominant principal components, and the one associated with the response is ranked number 3 (when estimated from the data). The methods are identified in the figure caption. The leftmost M point corresponds to principal components regression using the largest principal component. SPCA and shaving do much better than the other methods.

Figure 8 gives us a clue to what is going on. Shown are the first 1,000 of 5,000 feature loadings for two of the methods demonstrated in Figure 7 (chosen at the best solution points). Both methods correctly identified the important component (the one related to y involving the first 50 features). In a regular SVD of $\mathbf{X}$, this important component was dominated by two other components. In detail, the training data from model (35) has four *built-in* components, with singular values computed as 99.9, 88.3, 80.9, and 80.5. Empirically, we verified that component three is identified with the response mechanism, but its singular value is just above the noise level (the fifth singular value was 79.2). However, the mixed criterion also brings with it noisy coefficients, somewhat smaller, for all of the other variables, whereas SPCA sets most of the other loadings to 0. The coefficients for shaving show a very similar pattern to SPCA, whereas those for ridge and PLS are very similar to the mixed criterion and are not shown here.

Our experience with many similar examples is much the same, although the shaving method occasionally gets the wrong component completely. SPCA tends to be more reliable and is simpler to define, and hence is our method of choice. The simulations in the next section also support this choice.

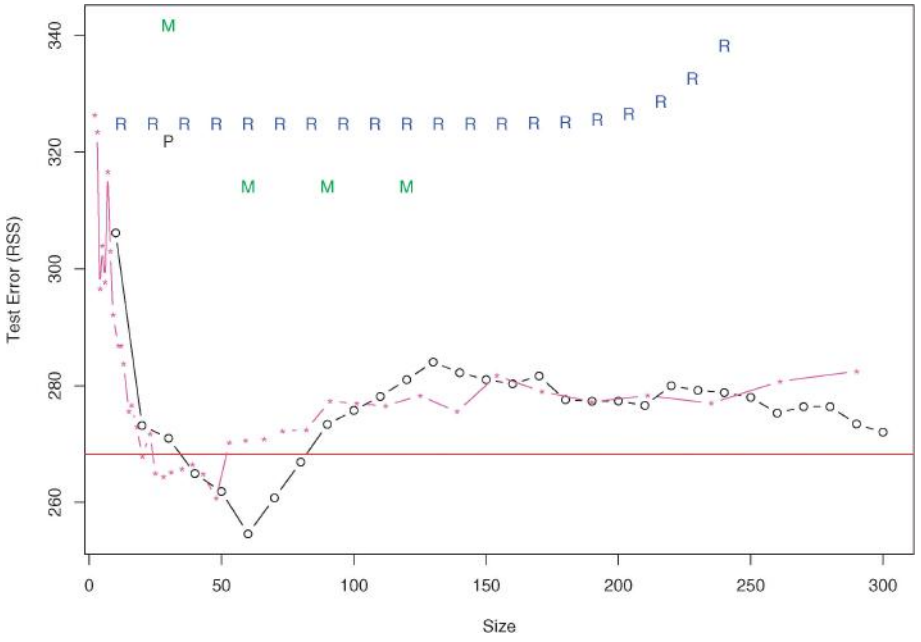

Figure 7. A Simulation Example Illustrating the Typical Behavior of the Different Methods. The data are generated according to the model (35) described in the next section, with N = 100 and p = 5,000. Ridge regression, PLS, and the mixed criterion all suffer from the very high dimensions. Although not shown, the regularization parameter λ for the ridge points increases to the right, as does the α for the mixed criterion, the leftmost value being 0. Both shaving and SPCA are indexed by subset size. The line labeled "truth" uses the known linear combination of 50 features as the regression predictor (○, SPCA; −, truth; M, mix; R, ridge; ⋆, shave; P, PLS).

7. SIMULATION STUDIES

We performed three simulation studies to compare the performance of the methods that we have considered. We describe the first two studies here, and the third one later. Each simulated dataset $\mathbf{X}$ consisted of 5,000 "genes" (rows) and 100 "patients" (columns). Let x_{ij} denote the "expression level" of the ith gene

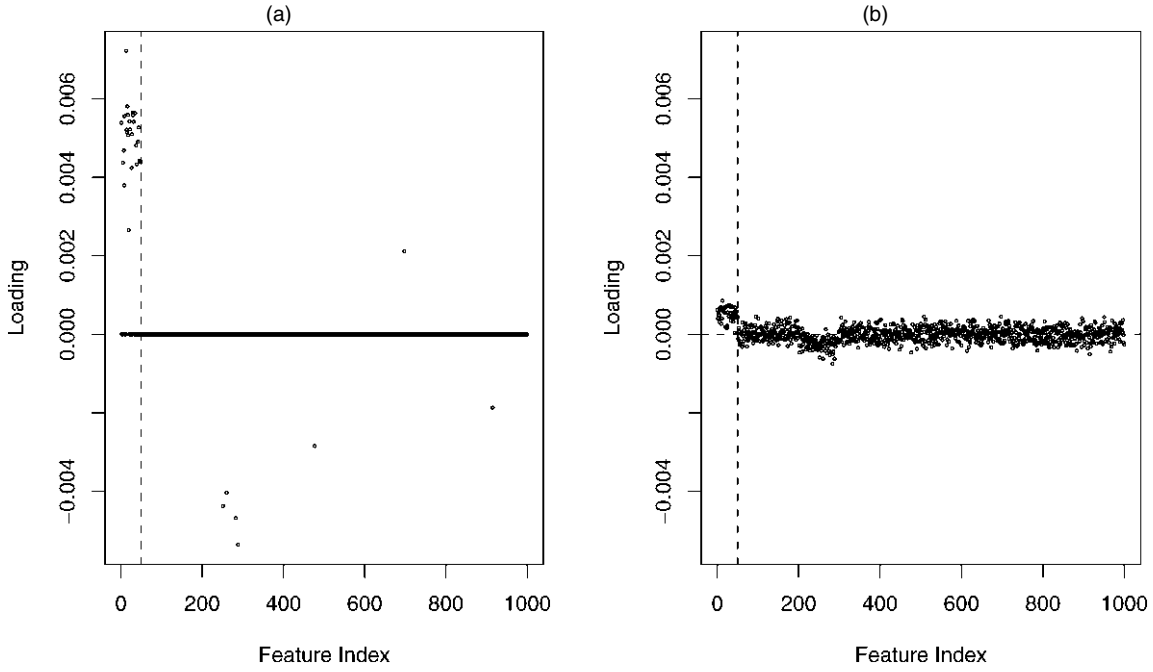

*Figure 8. Feature Loadings **w** for SPCA (a) and the Mixed Criterion (28) (b). The first 1,000 of 5,000 are shown, at the "best" solution point. The vertical line indicates that the first 50 variables generated the response. Whereas both of these methods were able to overwhelm the first two dominant principal components (which were unrelated to **y**), SPCA is able to ignore most of the variables, and the mixed criterion gives them all weight (albeit more weight to the first 50).*

and jth patient. In the first study we generated the data as

$$x_{ij} = \begin{cases} 3 + \epsilon_{ij} & \text{if } i \le 50, j \le 50 \\ 4 + \epsilon_{ij} & \text{if } i \le 50, j > 50 \\ 3.5 + \epsilon_{ij} & \text{if } i > 51, \end{cases} \tag{33}$$

where the ϵ_{ij}'s are independent normal random variables with mean 0 and variance 1. We also let

$$y_j = \frac{\sum_{i=1}^{50} x_{ij}}{25} + \epsilon_j, \tag{34}$$

where the ϵ_j's are independent normal random variables with mean 0 and standard deviation 1.5.

We designed this simulation so that there are two tumor "subclasses." Patients 1–50 belong to tumor class 1, and have slightly lower average expression levels in the patients with tumor class 2. Furthermore, because y is proportional to the sum of the expression level of the first 50 genes, $\mathbf{y}$ is slightly lower for patients with tumor class 1. The other 4,950 genes are unrelated to $\mathbf{y}$.

We applied eight methods to this simulated dataset: principal components regression, principal components regression using only the first principal component, PLS (one direction), ridge regression, lasso, supervised principal components, mixed variance–covariance, and gene shaving. We trained each of these models using a simulated dataset generated as described earlier. We select the optimal value of the tuning parameters for each method using 10-fold cross-validation. Then we used the same procedure to generate an independent test dataset and used the models that we built to predict $\mathbf{y}$ on the test dataset. We repeated this procedure 10 times and averaged the results. Table 2 gives the errors produced by each model.

We see that gene shaving and supervised principal components generally produce smaller cross-validation and test errors than any of the other methods, with the former holding a small edge. Principal components regression and PLS gave comparable results (although principal components regression performed slightly worse when restricted to one component).

Table 2. Results of the Simulation Study Based on the "Easy" Simulated Data

Method	*CV error*	*Test error*
PCR	$293.4_{(17.21)}$	$217.6_{(10.87)}$
PCR-1	$316.8_{(20.52)}$	$239.4_{(11.94)}$
PLS	$291.6_{(13.11)}$	$218.2_{(12.03)}$
Ridge regression	$298.0_{(14.72)}$	$224.2_{(12.35)}$
Lasso	$264.0_{(13.06)}$	$221.9_{(12.72)}$
Supervised principal components	$233.2_{(11.23)}$	$176.4_{(10.14)}$
Mixed variance–covariance	$316.7_{(19.52)}$	$238.7_{(10.24)}$
Gene shaving	$223.0_{(8.48)}$	$172.5_{(9.25)}$

NOTE: Each entry in the table represents the squared error of the test set predictions averaged over 10 simulations. The standard error of each error estimate is in parentheses. The prediction methods are: principal components regression (PCR), PCR restricted to using only one principal component (PCR-1), partial least squares (PLS), ridge regression, lasso, supervised principal components, mixed variance–covariance, and gene shaving.

Next, we generated a "harder" simulated dataset. In this simulation, we generated each x_{ij} as follows:

$$x_{ij} = \begin{cases} 3 + \epsilon_{ij} & \text{if } i \le 50, j \le 50 \\ 4 + \epsilon_{ij} & \text{if } i \le 50, j > 50 \\ 3.5 + 1.5 \cdot I(u_{1j} < .4) + \epsilon_{ij} & \text{if } 51 \le i \le 100 \\ 3.5 + .5 \cdot I(u_{2j} < .7) + \epsilon_{ij} & \text{if } 101 \le i \le 200 \\ 3.5 - 1.5 \cdot I(u_{3j} < .3) + \epsilon_{ij} & \text{if } 201 \le i \le 300 \\ 3.5 + \epsilon_{ij} & \text{if } i > 301. \end{cases} \tag{35}$$

Here the u_{ij} are uniform random variables on $(0, 1)$ and $I(x)$ is an indicator function. For example, for each of the genes 51–100, a single value u_{1j} is generated for sample j; if this value is larger than .4, then all of the genes in that block get 1.5 added. The motivation for this simulation is that there are other clusters of genes with similar expression patterns that are unrelated to $\mathbf{y}$. This is likely to be the case in real microarray data, because there are pathways of genes (that probably have similar expression patterns) that are not related to $\mathbf{y}$. Figures 7 and 8 illustrate some of the methods applied to a realization from this model.

We repeated the experiment described earlier using (35) to generate the datasets instead of (33). The results are given in Table 3. Most of the methods performed worse in this "harder" experiment. Once again, gene shaving and supervised principal components produced smaller errors than any of the competing methods; gene shaving shows much more variability than supervised principal components in this case.

Our third simulation study was quite different than the first two. We used the training and test expression datasets from Rosenwald et al. (2002), so as to obtain genes with "real-life" correlation. Fixing the expression data, we generated independent standard Gaussian coefficients θ_j, and finally generated a quantitative outcome $y_i = \sum_{j=1}^{p} x_{ij}\theta_j + \sigma Z$, with Z standard Gaussian. With $\sigma = 3$, about 30% of the variation in the outcome was explained by the true model. Multiple datasets were generated in this way, with expression data held fixed.

Ridge regression is the Bayes estimate in this setup, so we would expect it to perform the best. We were interested to see how other methods compared. Table 4 gives the results. Ridge regression is the best, followed in cross-validation error by PLS and in test error by the lasso. The other methods are substantially worse. Table 5 gives the average number of genes used by

Table 3. Results of the Simulation Study Based on the "Hard" Simulated Data

Method	*CV error*	*Test error*
PCR	$302.4_{(17.48)}$	$327.6_{(14.49)}$
PCR-1	$325.6_{(20.05)}$	$354.6_{(14.99)}$
PLS	$299.6_{(17.10)}$	$321.8_{(16.12)}$
Ridge regression	$301.0_{(18.47)}$	$328.0_{(16.38)}$
Lasso	$286.9_{(16.92)}$	$322.8_{(21.24)}$
Supervised principal components	$242.3_{(15.38)}$	$268.9_{(10.47)}$
Mixed variance–covariance	$322.5_{(19.64)}$	$349.8_{(16.02)}$
Gene shaving	$234.0_{(12.46)}$	$276.6_{(13.43)}$

NOTE: Each entry in the table represents the squared error of the test set predictions averaged over 10 simulations. The standard error of each error estimate is in parentheses. The prediction methods are the same as in Table 2.

Table 4. Third Simulation Study: Gaussian Prior for True Coefficients

Method	*CV error/ 1,000*	*Test error/ 1,000*
PCR	$399.423_{(16.617)}$	$194.489_{(16.298)}$
PCR-1	$559.708_{(29.637)}$	$283.356_{(24.320)}$
PLS	$322.513_{(11.142)}$	$203.375_{(16.978)}$
Ridge regression	$304.215_{(9.858)}$	$132.251_{(55.45)}$
Lasso	$356.886_{(15.281)}$	$169.266_{(10.217)}$
Supervised PC	$417.972_{(16.485)}$	$203.374_{(16.978)}$
Mixed covariance ($\mathbf{y}$)	$418.250_{(10.975)}$	$202.293_{(16.805)}$
Mixed covariance ($\hat{\mathbf{y}}$)	$551.924_{(26.251)}$	$286.255_{(23.149)}$
Gene shaving	$402.876_{(11.897)}$	$197.000_{(17.040)}$

supervised principal components and lasso in the three simulation studies. We see that lasso uses fewer genes than supervised principal components in each case. However, in the first two simulation studies, the number chosen by supervised principal components is closer to the actual number (50). In addition, if there are N samples and N is less than the total number of features p, then the lasso can never choose more than N features. This could be too restrictive, because there is no reason in general that the true number of important genes should be less than N.

8. APPLICATION TO VARIOUS SURVIVAL STUDIES

Here we compare several methods for performing survival analysis on real DNA microarray datasets. (Some of these results were also reported by Bair and Tibshirani 2004.) We applied the methods to four different datasets. First, we examined a microarray dataset consisting of patients with diffuse large B-cell lymphoma (Rosenwald et al. 2002). There are 7,399 genes, 160 training patients, and 80 test patients in this dataset. Second, we considered a breast cancer dataset (van't Veer et al. 2002) with 4,751 genes and 97 patients. We partitioned this dataset into a training set of 44 patients and a test set of 53 patients.

Next, we examined a lung cancer dataset (Beer et al. 2002) with 7,129 genes and 86 patients, which we partitioned into a training set of 43 patients and a test set of 43 patients. Finally, we considered a dataset of patients with acute myeloid leukemia (Bullinger et al. 2004), consisting of 6,283 genes and 116 patients. This dataset was partitioned into a training set of 59 patients and a test set of 53 patients.

In addition to supervised principal components, we examined the following methods: principal components regression, partial least squares, lasso, and two other methods that we call "median cut" and "clustering Cox," described by Bair and Tibshirani (2004). Both of these latter methods turn the problem into a two-class classification problem and then apply the nearest shrunken centroid classifier of Tibshirani, Hastie, Narasimhan, and Chu (2001). The median cut method stratifies the patients into highrisk or low risk, depending on whether they survived past the median survival time. The "clustering Cox" method is like supervised principal components, using two-means clustering applied to the genes with the highest Cox scores.

Table 5. Average Number of Genes (and standard deviation) for Supervised Principal Components and Lasso in Each of Three Simulation Studies

Method	*Simulation 1*	*Simulation 2*	*Simulation 3*
Supervised PC	$44.5_{(9.4)}$	$54.4_{(10.9)}$	$95.7_{(16.4)}$
Lasso	$32.8_{(8.7)}$	$23.1_{(6.2)}$	$42.9_{(5.5)}$

For PLS, ridge regression, and the lasso, we allowed the possibility of using more than one component, and chose this number by cross-validation. The results are given in Table 6. Overall, supervised principal components performs better than the competing methods. However, in the DLBCL example, the lasso does best. This is not surprising, because our use of the lasso as a post-processor for supervised principal components showed that only a few genes are needed for good prediction in this example.

9. THEORETICAL RESULTS

In this section we give details of our for supervised principal components in the Gaussian regression setting. Consistency results for survival data are discussed in the Appendix.

9.1 Setup

Suppose that the rows of $\mathbf{X}$ are iid. Then we can formulate a population model as follows. Denoting the rows by X_i^T $(i = 1, \ldots, N)$, we have the model

$$X_i \overset{\text{iid}}{\sim} \mathrm{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

where $\boldsymbol{\Sigma}$ $(p \times p)$ is the covariance matrix. Without loss of generality, we assume that $\boldsymbol{\mu} = \mathbf{0}$, because it can be quite accurately estimated from the data.

Suppose that $\mathbf{X}$ is partitioned as $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)$, where $\mathbf{X}_1$ is $N \times p_1$ and $\mathbf{X}_2$ is $N \times p_2$ with $p_1 + p_2 = p$. Assume that the corresponding partition of $\boldsymbol{\Sigma}$ is given by

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_1 & 0 \\ 0 & \boldsymbol{\Sigma}_2 \end{bmatrix}. \tag{36}$$

Suppose further that we can represent $\boldsymbol{\Sigma}_1$ $(p_1 \times p_1)$ as

$$\boldsymbol{\Sigma}_1 = \sum_{k=1}^{M} \lambda_k \boldsymbol{\theta}_k \boldsymbol{\theta}_k^T + \sigma^2 \mathbf{I}, \tag{37}$$

where $\boldsymbol{\theta}_k$ $(k = 1, \ldots, M)$ are mutually orthonormal eigenvectors and the eigenvalues $\lambda_1 \geq \cdots \geq \lambda_M > 0$. Here $\sigma^2 > 0$ represents the contribution of (isotropic) "background noise" that is unrelated to the interactions among genes. This model can be described as a covariance model for gene expressions that is an M-rank perturbation of identity. Here $1 \leq M \leq p_1 - 1$.

We can equivalently express the predictors through the following factor analysis model. Let $\mathcal{P}$ be the set of genes forming the columns of matrix $\mathbf{X}_1$. Then $|\mathcal{P}| = p_1$ and

$$X_{ij} = \sum_{k=1}^{M} \sqrt{\lambda_k} \theta_{jk} \eta_{ik} + \sigma w_{ij}, \qquad j \in \mathcal{P}, \tag{38}$$

represent the expression measurements for the genes in the set $\mathcal{P}$ of ith array (replicate), $i = 1, \ldots, N$ [cf. (10) in Sec. 2.2]. Here $\eta_{ik} \overset{\text{iid}}{\sim} \mathrm{N}(0, 1)$ and are independent of $w_{ij} \overset{\text{iid}}{\sim} \mathrm{N}(0, 1)$.

Our main assumption is that $\mathbf{X}_1$ is the matrix containing all of the columns whose variations are related to the variations in $\mathbf{y}$. First, assume that the selection procedure is such that it selects $\mathbf{X}_1$ with probability tending toward 1 as $N \to \infty$. In Section 10.4 we consider the more realistic scenario in which

Table 6. Comparison of the Different Methods on Four Different Datasets From Cancer Studies

	(a) DLBCL			(b) Breast cancer			(c) Lung cancer			(d) AML		
Method	R^2	*p value*	*NC*	R^2	*p value*	*NC*	R^2	*p value*	*NC*	R^2	*p value*	*NC*
(1) Median cut	.05	.047		.13	.0042		.15	.0016		.07	.0487	
(2) Clustering-Cox	.08	.006		.21	.0001		.07	.0499		.08	.0309	
(3) SPCA	.11	.003	2	.27	2.1×10^{-5}	1	.36	1.5×10^{-7}	3	.16	.0013	3
(4) PC regression	.01	.024	2	.22	.0003	3	.11	.0156	1	.08	.0376	1
(5) PLS	.10	.004	3	.18	.0003	1	.18	.0044	1	.07	.0489	1
(6) Lasso	.16	.0002	NA	.14	.001	NA	.26	.0001	NA	.05	.0899	NA

NOTE: The methods are (1) assigning samples to a "low-risk" or a "high-risk" group based on their median survival time, (2) using two-means clustering based on the genes with the largest Cox scores, (3) supervised principal components method, (4) principal components regression, (5) partial least squares regression, and (6) lasso. The table lists the R^2 (proportion of log-likelihood explained) and *p* values for the test set predictions, as well as the number of components used.

we estimate this subspace from data. Our key assumptions regarding the matrix $\mathbf{\Sigma}_1$ are given by conditions A1–A2 or, more generally, by conditions A1′ and A2′. We show in Section 10.2 that these conditions are sufficient for the consistency of the ordinary PCA-based estimators of θ_k and λ_k, $k = 1, \ldots, M$, when we perform such a PCA on the sample covariance matrix of $\mathbf{X}_1$. It follows from this that we can consistently estimate the parameters in the PC regression model for the response y described through (40); see Section 10.2 for details.

A1. The "signal" eigenvalues of $\mathbf{\Sigma}_1$ satisfy (identifiability condition for eigenvectors)

$$\lambda_1 > \cdots > \lambda_M > 0,$$

and M is a fixed positive integer.

A2. $p_1 \to \infty$ as N increases to infinity in such a way that $p_1/N \to 0$.

It may be possible that the noise variance σ^2 and the "signal" eigenvalues λ_k also vary with N. Under this setting, to guarantee consistency, we need to replace conditions A1 and A2 by the following:

A1′. The eigenvalues are such that $\lambda_k/\lambda_1 \to \rho_k$ for $k = 1, \ldots, M$ with $1 = \rho_1 > \rho_2 > \cdots > \rho_M > 0$ and $\lambda_1 \to c > 0$ as $N \to \infty$. Moreover, $\sigma^2 \to \sigma_0^2 \in [0, \infty)$ as $N \to \infty$.

A2′. p_1 varies with N in such a way that $\sigma^2 p_1/(N\lambda_1) \to 0$ as $N \to \infty$.

Notice that condition A1′ is an asymptotic identifiability condition for the eigenvectors $\theta_1, \ldots, \theta_M$. This is because if $\rho_k = \rho_{k+1}$ for some $1 \le k \le M-1$, then for large N, and for any 2×2 orthogonal matrix $\mathbf{C}$, the columns of the matrix $\mathbf{C}[\theta_k : \theta_{k+1}]$ are approximately the eigenvectors of $\mathbf{\Sigma}_1$ corresponding to eigenvalues λ_k and λ_{k+1}. This would imply a very special kind of inconsistency in the estimates of θ_k and θ_{k+1}, even though we still may be able to estimate the corresponding eigenspace consistently. To avoid the technicalities associated to this situation, we restrict ourselves to condition A1′. Note that the condition $\lambda_1 \to c > 0$, taken together with the first part of condition A1′, implies that all of the M eigenvalues λ_k converge to positive limits.

Remark 1. Conditions A1′ and A2′ allow for the possibility that $\lambda_1/\sigma^2 \to \infty$ and p_1/N converges to a positive limit. This particular facet becomes relevant when we try to connect to the scenario that we describe here. Consider the model (38) and suppose that $M = 1$. In this case, if $\sqrt{\lambda_1}\theta_{j1}$ is roughly of the same order of magnitude for all $j \in \mathcal{P}$, then $\lambda_1 \sim p_1$ for p_1 large. Even if otherwise, it is reasonable to believe that the "signal-to-noise ratio" λ_1/σ^2 is going to ∞ as $p_1 \to \infty$, because the presence of larger number of genes associated with a common latent factor yields a greater amount of information.

Suppose that the SVD of $\mathbf{X}_1$ is given by

$$\mathbf{X}_1 = \mathbf{U}\mathbf{D}\mathbf{V}^T, \quad \text{where}$$
$$\mathbf{U} \text{ is } N \times m, \mathbf{D} \text{ is } m \times m, \text{ and } \mathbf{V} \text{ is } p_1 \times m, \tag{39}$$
$$\text{with } m = \min(N, p_1).$$

Here N is the number of observations (patients) and p_1 is the dimension (number of genes). Let $\mathbf{u}_1, \ldots, \mathbf{u}_m$ denote the columns of $\mathbf{U}$ and let $\mathbf{v}_1, \ldots, \mathbf{v}_m$ denote the columns of $\mathbf{V}$. For obvious reasons, we set $\widehat{\boldsymbol{\theta}}_k = \mathbf{v}_k$, $k = 1, \ldots, M$. Also, we denote the diagonal elements of $\mathbf{D}$ by $d_1 > \cdots > d_m$.

The model for the response is

$$\mathbf{y} = \beta_0 \frac{1}{\sqrt{N}}\mathbf{1} + \sum_{k=1}^{K} \beta_k \frac{1}{\sqrt{N}}\boldsymbol{\eta}_k + \mathbf{Z}, \tag{40}$$

where $K \le M$, $\mathbf{1}$ is the vector with 1 in each coordinate, and $\mathbf{Z} \sim \mathrm{N}_N(\mathbf{0}, \frac{\tau^2}{N}\mathbf{I})$ independent of $\mathbf{X}$ for some $\tau \in [0, \infty)$.

It may seem from (40) that the parameters associated with the distribution of the sample size depend on the sample size N. But in reality the model (40) is an exact analog of the model for response given by (9) and (10). This is seen by dividing through (9) by $\sqrt{N}$ and taking $\alpha_{1jm} = \theta_{jm}$ in (10) and setting $\mathbf{U}_m = \boldsymbol{\eta}_m$ for $m = 1, \ldots, M$.

Remark 2. Note that we also could have described the model in terms of similar quantities for the full dataset, that is, $\mathbf{X}$ (correspondingly $\mathbf{\Sigma}$). There are two difficulties associated with this formulation. First, it is not at all likely that all the systematic variation in the gene expressions is associated with the variation in the response. So even if model (36)–(37) is true, there is no guarantee that the largest K eigenvalues of $\mathbf{\Sigma}$ are the largest K eigenvalues of $\mathbf{\Sigma}_1$. This will result in the addition of spurious (i.e., unrelated to the response $\mathbf{y}$) components to the model.

The second difficulty relates to the accuracy of estimation. Because typically p is very large (in fact much larger than, or at least comparable to, the sample size N), it is almost never going to be the case that assumption A2′ is satisfied (with p_1 replaced by p). But the assumption for p_1 is reasonable, because only a few genes are expected to be associated with a certain type of disease. Violation of this condition results in an inconsistency

in the estimates of $\boldsymbol{\theta}_k$ (see the next section for details). So the procedure of selecting the genes before performing the PCA regression is not only sensible, but also in effect necessary.

9.2 Results on Estimation of $\boldsymbol{\theta}_k$ and λ_k

To discuss consistency of the eigenvectors $\boldsymbol{\theta}_k$, we consider the quantity $\operatorname{dist}(\widehat{\boldsymbol{\theta}}_k, \boldsymbol{\theta}_k)$, where dist is a distance measure between two vectors on the p_1-dimensional unit sphere. We can choose either $\operatorname{dist}(\mathbf{a}, \mathbf{b}) = \angle(\mathbf{a}, \mathbf{b})$ (i.e., the angle between $\mathbf{a}$ and $\mathbf{b}$) or $\operatorname{dist}(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \operatorname{sign}(\mathbf{a}^T\mathbf{b}) \cdot \mathbf{b}\|_2$ for $\mathbf{a}, \mathbf{b} \in \mathbb{S}^{p_1}$.

First suppose we perform PCA on the full dataset $\mathbf{X}$ and estimate $\boldsymbol{\theta}_k$ by $\widetilde{\boldsymbol{\theta}}_k$, the restriction of the kth right singular vector of $\mathbf{X}$ to the coordinates corresponding to the set $\mathbf{X}_1$. Then the following result asserts that if p is very large, then we may not have consistency.

Theorem 1 (Lu 2002; Johnstone and Lu 2006). Suppose that (38) and condition A1 hold (and assume that σ^2 and λ_k are fixed) and that $p/N \to \gamma \in (0, \infty)$ as $N \to \infty$. Then

$$\operatorname{dist}(\widetilde{\boldsymbol{\theta}}_k, \boldsymbol{\theta}_k) \not\to 0 \quad \text{in probability as } N \to \infty;$$

that is, the usual PCA-based estimate of $\boldsymbol{\theta}_k$ is inconsistent.

Under the same conditions as in Theorem 1, the sample eigenvalues are also inconsistent estimates for the populations eigenvalues. Baik and Silverstein (2004) derived almost-sure limits of the sample eigenvalues in a similar setup under minimal distributional assumptions.

From now onward we treat exclusively the singular value decomposition of $\mathbf{X}_1$. We denote the PCA-based estimate of the kth largest eigenvalue of $\boldsymbol{\Sigma}_1$ by $\widehat{\ell}_k$, $k = 1, 2, \ldots, m$. Observe that $\widehat{\ell}_k = \frac{1}{N} d_k 2$. The corresponding population quantity is $\ell_k := \lambda_k + \sigma^2$.

A natural estimator of λ_k is $\widehat{\lambda}_k = \max\{\widehat{\ell}_k - \sigma^2, 0\}$ if σ^2 is known. But, if σ^2 is unknown, then we can estimate this by various strategies. One approach is to use the median of the diagonal elements of $\frac{1}{N}\mathbf{X}_1^T\mathbf{X}_1$ as a (usually biased) estimate of σ^2 and then define $\widehat{\lambda}_k = \max\{\widehat{\ell}_k - \widehat{\sigma}^2, 0\}$.

Now we establish the consistency for PCA restricted to the matrix $\mathbf{X}_1$. We do not give a complete proof of this result, because it is rather long and somewhat technical in nature. But in the Appendix we give an outline of the proof for the case $p_1/N \to 0$ and $\{\lambda_k\}_{k=1}^M$ and σ^2 fixed. The details have been given by Paul (2005).

Theorem 2. Let $\operatorname{dist}(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \operatorname{sign}(\mathbf{a}^T\mathbf{b}) \cdot \mathbf{b}\|_2$. Let $h(x) := \frac{x^2}{1+x}$ and $g(x, y) := \frac{(x-y)^2}{xy}$. Assume that (38) holds and that the set $\mathcal{P}$ is selected with probability tending toward 1 as $N \to \infty$.

- Suppose that conditions A1′ and A2′ hold. Then, for $1 \le k \le M$,

$$\mathbb{E}\operatorname{dist}^2(\widehat{\boldsymbol{\theta}}_k, \boldsymbol{\theta}_k) \le \left[\frac{p_1}{Nh(\lambda_k/\sigma^2)} + \frac{1}{N}\sum_{k \ne k'} \frac{1}{g(\lambda_k + \sigma^2, \lambda_{k'} + \sigma^2)}\right] \times (1 + o(1)). \tag{41}$$

  If, moreover, $\lambda_1/\sigma^2 \to \infty$, then $\widehat{\ell}_k = \lambda_k(1 + o_P(1))$.
- If σ^2 and the λ_k's are fixed and conditions A1 and A2 hold, then (41) holds and $\widehat{\ell}_k \xrightarrow{P} \ell_k = \lambda_k + \sigma^2$ as $N \to \infty$.

9.3 Estimation of β_k

In this section we discuss estimation of the parameters β_k, $k = 1, \ldots, K$. To simplify the exposition, we treat σ^2 and λ_k's as fixed and assume that conditions A1 and A2 hold. Recall that our model for the response variable is $\mathbf{y} = \hat{\boldsymbol{\beta}}_0 + \sum_{k=1}^K \hat{\beta}_k \hat{\boldsymbol{\mu}}_k$.

Suppose that either σ^2 is known or a consistent estimate $\widehat{\sigma}^2$ is available. Then define $\widehat{\lambda}_k = \max\{\widehat{\ell}_k - \widehat{\sigma}^2, 0\}$. Let $\mathbf{u}_k$ be as before and define $\widetilde{\mathbf{u}}_k$ as $\frac{1}{\sqrt{\widehat{\lambda}_k}} \frac{1}{\sqrt{N}} \mathbf{X}_1 \mathbf{v}_k$ if $\widehat{\lambda}_k > 0$, and as any fixed unit vector [say $(1, 0, \ldots, 0)^T$] otherwise. Define an estimate of β_k (for $1 \le k \le K$) as $\widetilde{\beta}_k = \widetilde{\mathbf{u}}_k^T \mathbf{y}$. We can compare its performance with another estimate $\widehat{\beta}_k = \mathbf{u}_k^T \mathbf{y}$ with $\mathbf{u}_k$ as before. Also, define $\widehat{\beta}_0 = \widetilde{\beta}_0 = \frac{1}{\sqrt{N}} \sum_{j=1}^N y_j$.

Observe that

$$\begin{aligned}\mathbf{u}_k &= \frac{1}{d_k}\mathbf{X}_1\mathbf{v}_k \\ &= (\widehat{\ell}_k)^{-1/2}\frac{1}{\sqrt{N}}\mathbf{X}_1\widehat{\boldsymbol{\theta}}_k \\ &= (\widehat{\ell}_k)^{-1/2}\left[\sum_{l=1}^M \sqrt{\lambda_l}(\boldsymbol{\theta}_l^T\widehat{\boldsymbol{\theta}}_k)\frac{1}{\sqrt{N}}\eta_l + \frac{\sigma}{\sqrt{N}}\mathbf{W}\widehat{\boldsymbol{\theta}}_k\right],\end{aligned}$$

where $\mathbf{W}$ is the $N \times p_1$ matrix whose rows are $\mathbf{w}_i^T$ ($i = 1, \ldots, N$). Then, because $\widehat{\boldsymbol{\theta}}_k = \boldsymbol{\theta}_k + \boldsymbol{\varepsilon}_k$ (as a convention assuming $\widehat{\boldsymbol{\theta}}_k^T\boldsymbol{\theta}_k > 0$), where $\|\boldsymbol{\varepsilon}_k\|_2 = O_P(\sqrt{p_1/N})$,

$$\begin{aligned}\mathbf{u}_k = \frac{\sqrt{\lambda_k}}{\sqrt{\lambda_k + \sigma^2}}\frac{1}{\sqrt{N}}\boldsymbol{\eta}_k(1 + o_P(1)) \\ + \frac{\sigma}{\sqrt{\lambda_k + \sigma^2}}\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{\theta}_k(1 + o_P(1)) + \boldsymbol{\delta}_k,\end{aligned} \tag{42}$$

where $\|\boldsymbol{\delta}_k\|_2 = O_P(\sqrt{p_1/N})$. To prove this last statement, we need only use Theorem 2 together with the fact that $\|\frac{1}{N}\mathbf{W}^T\mathbf{W}\|_2 = 1 + o_P(1)$, because

$$2\left\|\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{\varepsilon}_k\right\|_2 \le 2\left\|\frac{1}{N}\mathbf{W}^T\mathbf{W}\right\|_2 \|\boldsymbol{\varepsilon}_k\|_2 = O_P\left(\frac{p_1}{N}\right)$$

and

$$|\boldsymbol{\varepsilon}_k^T\boldsymbol{\theta}_l| \le \|\boldsymbol{\varepsilon}_k\|_2\|\boldsymbol{\theta}_l\|_2 = O_P\left(\sqrt{\frac{p_1}{N}}\right) \quad \text{for } 1 \le l \ne k \le M,$$

and, finally, $\|\boldsymbol{\eta}_l\|_2 = \sqrt{N}(1 + o_P(1))$ for all $l = 1, \ldots, M$.

From, this it follows that

$$\widetilde{\mathbf{u}}_k = \frac{1}{\sqrt{N}}\boldsymbol{\eta}_k(1 + o_P(1)) + \frac{\sigma}{\sqrt{\lambda_k}}\frac{1}{\sqrt{N}}\mathbf{W}\boldsymbol{\theta}_k(1 + o_P(1)) + \widetilde{\boldsymbol{\delta}}_k, \tag{43}$$

where $\|\widetilde{\boldsymbol{\delta}}_k\|_2 = O_P(\sqrt{p_1/N})$. Note that the vectors $\{\mathbf{W}\boldsymbol{\theta}_k : k = 1, \ldots, M\}$ are independent $\mathrm{N}_N(\mathbf{0}, \mathbf{I})$ and independent of $\{\boldsymbol{\eta}_k : k = 1, \ldots, M\}$, because the $\boldsymbol{\theta}_k$'s are mutually orthonormal.

To establish consistency of $\widetilde{\beta}_k$, $1 \le k \le K$, note that [by (43)]

$$\begin{aligned}\widetilde{\beta}_k = \beta_0\frac{1}{\sqrt{N}}\widetilde{\mathbf{u}}_k^T\mathbf{1} \\ + \sum_{l=1}^K \beta_l \frac{1}{N}\left[\left(\boldsymbol{\eta}_k + \frac{\sigma}{\sqrt{\lambda_k}}\mathbf{W}\boldsymbol{\theta}_k\right)(1 + o_P(1)) + \sqrt{N}\widetilde{\boldsymbol{\delta}}_k\right]^T\boldsymbol{\eta}_l \\ + \widetilde{\mathbf{u}}_k^T\mathbf{Z}\end{aligned}$$

$$= \beta_0\left(O_P\left(\frac{1}{\sqrt{N}}\right) + o_P(1)\right) + \beta_k(1 + o_P(1) + \widetilde{\boldsymbol{\delta}}_k^T \frac{1}{\sqrt{N}}\boldsymbol{\eta}_k\Big)$$
$$+ \sum_{l\neq k}\beta_l\left(O_P\left(\frac{1}{\sqrt{N}}\right) + \widetilde{\boldsymbol{\delta}}_k^T \frac{1}{\sqrt{N}}\boldsymbol{\eta}_l\right) + O_P\left(\frac{1}{\sqrt{N}}\right)$$
$$= \beta_k(1 + o_P(1)),$$

because $\frac{1}{N}\boldsymbol{\eta}_k^T\boldsymbol{\eta}_l = O_P(1/\sqrt{N})$ if $k \neq l$ and $\frac{1}{N}\boldsymbol{\eta}_l^T\mathbf{W}\boldsymbol{\theta}_k = O_P(1/\sqrt{N})$ for all k, l (by independence), $\|\widetilde{\boldsymbol{\delta}}_k\|_2 = o_P(1)$, and $\widetilde{\mathbf{u}}_k^T\mathbf{Z} = \|\widetilde{\mathbf{u}}_k\|_2\langle\frac{\widetilde{\mathbf{u}}_k}{\|\widetilde{\mathbf{u}}_k\|_2}, \mathbf{Z}\rangle$. Note that the second term in the last product is a $\mathrm{N}(0, \frac{\tau^2}{N})$ random variable, and the first term is $\sqrt{(\lambda_k + \sigma^2)/\lambda_k}(1 + o_P(1))$ by (43).

It is easy to verify that $\widehat{\beta}_0 = \beta_0(1 + o_P(1))$. But, from the foregoing analysis, it is clear that the estimator $\widehat{\beta}_k = \mathbf{u}_k^T\mathbf{y}$, for $1 \leq k \leq K$, is not consistent in general. In fact, $\widehat{\beta}_k = \sqrt{\frac{\lambda_k}{\lambda_k+\sigma^2}}\beta_k(1 + o_P(1))$ when the λ_k's and σ^2 are fixed. However, as we indicated in Remark 1, it is reasonable to assume that $\lambda_1/\sigma^2 \to \infty$ as $p_1, N \to \infty$. This will ensure (via the first part of Thm. 2) that the factor $\sqrt{\lambda_k/\widehat{\ell}_k} \to 1$ in probability as $N \to \infty$ when conditions A1′ and A2′ hold. Therefore, we have $\widehat{\beta}_k = \beta_k(1 + o_P(1))$ for $1 \leq k \leq K$. This in a way validates the claim that having more genes (i.e., larger p_1) associated with the response gives better predictability.

9.4 Consistency of the Coordinate Selection Scheme: Regression Model

In this section we describe some situations under which SPCA will consistently select the set $\mathcal{P}$ of coordinates (genes) whose variability is associated with that of the response through the model given by (36), (38), and (40). Here we work under the assumption that $p_1 = O(N^\alpha)$ for some $\alpha \in (0, 1)$ and $\log p \asymp \log N$. The second assumption covers a wide range of possible situations. The key point that we emphasize is that to be able to recover the set $\mathcal{P}$ of predictors associated with the response, we may need some identifiability conditions on this set. Our method may work under more general circumstances, but here we restrict our attention to cases that are analytically tractable and relatively simple to interpret.

First, observe that we can write the vector of univariate scores as $\mathbf{s} = \mathbf{H}_\mathbf{X}^{-1}\mathbf{X}^T\mathbf{y}$, where $\mathbf{H}_\mathbf{X} = \mathrm{diag}(\|\mathbf{x}_1\|, \ldots, \|\mathbf{x}_p\|)$. Because the rows of $\mathbf{X}_2$ are independent $\mathrm{N}_{p_2}(\mathbf{0}, \boldsymbol{\Sigma}_2)$ r.v. independent of $\mathbf{X}_1$, invoking (38), we can express the nonnormalized score vector $\widetilde{\mathbf{s}} := \mathbf{X}^T\mathbf{y}$ in the form

$$\widetilde{\mathbf{s}} = \begin{bmatrix} (\sum_{k=1}^M \sqrt{\lambda_k}\boldsymbol{\theta}_k\boldsymbol{\eta}_k^T + \sigma\mathbf{W}^T)\mathbf{y} \\ \boldsymbol{\Sigma}_2^{1/2}\mathbf{C}\mathbf{y} \end{bmatrix}, \tag{44}$$

where $\mathbf{C}$ is a $p_2 \times N$ matrix whose entries are iid $\mathrm{N}(0, 1)$ independent of $\mathbf{X}_1$ and $\mathbf{Z}$ (and hence $\mathbf{y}$). Observe that $\mathbf{W}^T$ is independent of $\mathbf{y}$. For expositional purposes, we work with $\widetilde{\mathbf{s}}$ rather than with $\mathbf{s}$.

This shows that if we consider the jth element of $\widetilde{\mathbf{s}}$ for $j \in \mathcal{P}$, then

$$\frac{1}{\sqrt{N}}\widetilde{s}_j = \frac{1}{N}\left(\sum_{k=1}^M \sqrt{\lambda_k}\boldsymbol{\theta}_{jk}\boldsymbol{\eta}_k^T\right)$$
$$\times \left(\beta_0\mathbf{1} + \sum_{k'=1}^K \beta_{k'}\boldsymbol{\eta}_{k'} + \sqrt{N}\mathbf{Z}\right) + \frac{\sigma}{\sqrt{N}}(\mathbf{W}^T\mathbf{y})_j$$
$$= \beta_0\sum_{k=1}^M \sqrt{\lambda_k}\theta_{jk}O_P\left(\frac{1}{\sqrt{N}}\right)$$
$$+ \sum_{k=1}^K \beta_k\sqrt{\lambda_k}\theta_{jk}\left(1 + O_P\left(\frac{1}{\sqrt{N}}\right)\right)$$
$$+ \sum_{k=1}^M \sqrt{\lambda_k}\theta_{jk}\sum_{k'\neq k}^K \beta_{k'}O_P\left(\frac{1}{\sqrt{N}}\right)$$
$$+ \sigma\left(\sum_{k=0}^K \beta_k\right)O_P\left(\frac{1}{\sqrt{N}}\right)$$
$$= \sum_{k=1}^K \beta_k\sqrt{\lambda_k}\theta_{jk} + O_P\left(\frac{1}{\sqrt{N}}\right).$$

But on the other hand, if $j \notin \mathcal{P}$, then, assuming that $\|\boldsymbol{\Sigma}_2\|_2$ is bounded above,

$$\frac{1}{\sqrt{N}}\widetilde{s}_j = \frac{1}{\sqrt{N}}\left(\boldsymbol{\Sigma}_2^{1/2}\mathbf{C}\mathbf{y}\right)_j = \left(\boldsymbol{\Sigma}_2^{1/2}\right)_j^T\frac{1}{\sqrt{N}}\mathbf{C}\mathbf{y} = O_P\left(\frac{1}{\sqrt{N}}\right).$$

Thus, for that the "signal" $\zeta_j^K := \sum_{k=1}^K \beta_k\sqrt{\lambda_k}\theta_{jk}$ to be detectable, it must be $\gg 1/\sqrt{N}$. Large deviation bounds suggest that we can recover with sufficient accuracy only those coordinates j for which $|\zeta_j^K| \geq c_0\sqrt{\log N/N}$ for some constant $c_0 > 0$ (which depends on σ, the λ_k's and β_k's and $\|\boldsymbol{\Sigma}_2\|_2$).

Potentially, many ζ_j^K's could be smaller than that, and hence those coordinates will not be selected with a high probability. If we make the threshold too small, then we will include many "spurious" coordinates (i.e., those with $j \notin \mathcal{P}$), which can cause problems in estimation in various ways that we discussed already.

If $K = 1$, then the jth component of the signal vector $\boldsymbol{\zeta}^K$ is proportional to $\sqrt{\lambda_1}\theta_{j1}$. So the scheme will select only those coordinates j for which $\sqrt{\lambda_1}|\theta_{j1}|$ is big. This may not exhaust the set $\{1, \ldots, p_1\}$, but as far as consistent estimation of θ_1 and λ_1 is concerned, it is adequate. Thus, when $K = 1$, the coordinate selection scheme is consistent.

In the case where $K > 1$, we may encounter a problem. This is because the method that we described relies on a fixed linear functional of the vector $\mathbf{t}_j = (\sqrt{\lambda_1}\theta_{j1}, \ldots, \sqrt{\lambda_K}\theta_{jK})$, viz., ζ_j^K. Thus even if at least one entry of $\mathbf{t}_j$ is quite big, we may miss that coordinate j. In other words, when $K > 1$, in general there is no guarantee that the coordinate selection scheme is consistent. A closer look suggests that in general we do not have sufficient identifiability constraints on the set of predictors $\mathcal{P}$. One way to impose this constraint is to say that $|\zeta_j^K|$ is above a threshold of the form $c_0\sqrt{\log N/N}$ whenever $\|\mathbf{t}_j\|$ is above a threshold $c_1\sqrt{\log N/N}$ for some constants $c_0, c_1 > 0$. Even though this condition may not be satisfied exactly, it turns out that we only need the following, somewhat weaker constraint:

A3. The set of variables $\mathcal{P}$ determining $\mathbf{X}_1$ is such that if $\mathcal{P}_{N,\beta}$ denotes the set of all $j \in \mathcal{P}$ with

$$\left|\left\langle \mathbf{t}_j, \frac{\boldsymbol{\beta}}{\|\boldsymbol{\beta}\|}\right\rangle\right| \geq c_0\sqrt{\frac{\log N}{N}} \tag{45}$$

for some constant $c_0 > 0$ independent of $\boldsymbol{\beta}$, then

$$\sum_{j \in \mathcal{P} \setminus \mathcal{P}_{N,\beta}} \frac{1}{\lambda_1} \|\mathbf{t}_j\|^2 \to 0 \quad \text{as } N \to \infty.$$

Observe that A3 is a constraint on the entire model, not just on the distribution of the predictors $\mathbf{X}$. The physical meaning of this constraint is as follows: The predictors in the class $\mathcal{P}$ that have significant variance because the first K components of variation in the model (38) are highly correlated with the response. This is because $\sigma^2 + \|\mathbf{t}_j\|^2 + \sum_{k=K+1}^{M} \lambda_k \theta_{jk}^2$ is the variance of the jth predictor, and $\frac{1}{\sqrt{N}} \zeta_j^K$ is the covariance of $\mathbf{X}_j$ and $\mathbf{y}$. It also means that the coordinates that we may fail to pick have negligible contribution to the overall variability associated with the first K components of variation. Note that this condition is automatically satisfied when $K = 1$.

A different way to impose identifiability on the set of predictors $\mathcal{P}$ is to impose a constraint on the parameter $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)$. In this way we require that a specific $K \times K$ matrix $\mathbf{H}(\boldsymbol{\beta})$, whose entries are polynomials in β_k's, has small condition number. We define $\mathbf{H}$ as follows. The first column of $\mathbf{H}(\boldsymbol{\beta})$ is $\boldsymbol{\beta}$ itself. We can use this constraint to ensure that we select all of the big coordinates even when $K \gg 1$. We could generalize the selection scheme as follows.

For integers $r = 1, 2, \ldots, K$, define the set J_r to be the set of coordinates j such that $|s_j^{(r)}| > \alpha_j^{(r)}$ where $\alpha_j^{(r)}$ is a threshold of the order $\sqrt{\log N}$ and $s_j^{(r)}$ is the jth coordinate of $\frac{1}{\sqrt{N}}(\mathbf{X}^T \mathbf{y}^{(r)})$ where the lth coordinate of $\mathbf{y}^{(r)}$ is $(\sqrt{N} y_l)^{2r-1}$. In particular $\mathbf{y}^{(1)} = \sqrt{N}\mathbf{y}$, so that $\mathbf{s}^{(1)} = \mathbf{s}$, as defined earlier. Finally, take the union $J := \bigcup_{r=1}^{K} J_r$ and take J to be the final selection. An analysis of this scheme shows that for $j \in \mathcal{P}$, $\frac{1}{\sqrt{N}} s_j^{(r)} = \mathbf{t}_j^T \mathbf{H}_r(\boldsymbol{\beta}) + O_P(1/\sqrt{N})$, where $\mathbf{H}_r(\boldsymbol{\beta})$ is the rth column of H. Then, by the constraint on the matrix $\mathbf{H}(\boldsymbol{\beta})$, for any $j \in \mathcal{P}$, we have $j \notin J$ if and only if $\|\mathbf{t}_j\|$ is "small" (meaning smaller than a certain threshold of the form $c_2\sqrt{\log N / N}$ for some constant $c_2 > 0$).

Remark 3. Of the two methods of imposing identifiability constraints, the second one is admittedly rather ad hoc and does not have a meaningful generalization beyond the regression setting. However, the first constraint may often be satisfied in practice, because some part of the variability in the predictors may be directly linked to the variability in the response. This is likely to be true if, for example, there is a causal relationship.

10. SOME PRACTICAL ISSUES AND GENERALIZATIONS

Here we mention some ways in which the supervised principal components can be applied in practice.

Joint Fitting With Other Covariates. Typically, there may be covariates measured on each of the cases, and it might be of interest to adjust for these. For example, in gene expression survival studies, in addition to the predictors $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_p$, we might have available covariates $\mathbf{z} = (z_1, z_2, \ldots, z_k)$, such as tumor stage and tumor type. There might be interest in finding gene expression predictors that work independently of stage and tumor; that is, having adjusted for these factors, the gene expression predictor is still strongly related to survival.

To compare the supervised principal component predictor to competing predictors, one can simply fit them together in a predictive model for the test set. In the lymphoma example, the International Prognostic Index (IPI) (low, medium or high) is a widely used clinical predictor of survival. We fit both the supervised principal component predictor and IPI to the test set, then determined the p values when each from removed separately from the joint model. These were .001 for the supervised principal component and .05 for IPI. Thus the effect of the supervised principal components predictor is strongly independent of IPI, whereas IPI is only moderately independent of the supervised principal components predictor.

We can also explicitly encourage the supervised principal components PC predictor to look for variation that is independent of competing predictors. To do this, we do a linear regression of each gene on the competing predictors, replacing each gene's measurements by the residuals from this process. We then apply the supervised principal components procedure to the residual matrix. This process decorrelates the gene expression and competing predictors and forces the principal components to be orthogonal to the competing predictors. This same approach can be used with other methods, such as PLS.

Use of Unlabelled Samples. In some settings, we have available both "labeled" data (e.g., gene expression profiles with a measured survival times) and unlabeled data (just gene expression profiles). In fact, one might have many unlabeled samples and only a few labeled ones, because obtaining outcome information can be more difficult. In this setting it might be helpful to use the unlabeled data in some way, because they contain information about the correlation between the features. Because of the simple form of the supervised principal components predictor, there is an easy way to do this. Suppose that the feature matrices for the labeled and unlabeled data are $\mathbf{X}^L$ and $\mathbf{X}^U$. In the first step, we use just $\mathbf{X}^L$ (and the outcome) to choose the features. Then we use the full set of features $(\mathbf{X}^L, \mathbf{X}^U)$ to compute principal components. The added information provided by the unlabeled samples can potentially improve the accuracy of the supervised principal components.

Application to Other Data Types. The supervised principal components idea can be applied to other types of outcome measures, such as classification outcomes. In that case, we could choose features having the largest between-class to within-class variation, then compute the principal components of the selected data. Then the principal component would be fit in a multiple logistic regression to predict the class label. Although this procedure seems promising, we have not yet found examples where it improves on methods such as the nearest shrunken centroid approach (Tibshirani et al. 2001). The explanation may lie in the soft-thresholding inherent in nearest shrunken centroids; it may have the same beneficial effect as the thresholding in supervised principal components.

11. DISCUSSION AND LIMITATIONS

Supervised principal components represents a promising tool for prediction in regression and generalized regression problems. It is a simple idea that has probably been tried many times in practice. Here we have explored its application to gene expression studies.

Regression is an important and difficult problem in statistics; it is especially difficult when the number of features p greatly exceeds the number or observations N. Overfitting can occur with even moderately complex models, and identifying the important features is fraught with danger because of the large number of features, many of which are often highly correlated. Despite the difficulty in identifying important features, however, this is a high priority for biologists in gene expression studies.

Supervised principal components approaches this difficult problem through a semisupervised strategy, looking for gross structure in the data that aligns itself with the outcome. Only later in the process does it try to pare down the set of features to a much smaller list (through its importance scores). A crucial practical aspect of this importance score is the fact that is provides a *fixed* ordering of features. Thus we start with a list of 200 features and ask for a submodel containing just 20 features; the constructed model consists of the 20 features among the 200 with the largest importance scores. In contrast, using a method like the lasso, the following could happen: We deliver a model having 200 features to our collaborator, who then asks for smaller model, containing just 20 features. So we change the lasso bound to achieve this and obtain a new model containing 20 features, some or none of which were in the original list of 200! This seems like an unsatisfactory approach to model selection in this setting.

Despite the encouraging performance of supervised principal components, the high-dimensional regression problem is very difficult and should be approached with caution. There are many issues that need further development and careful study. Many of these were pointed out by the editors and referees. We list some here:

- The ability of cross-validation to select the "correct" set of genes has not been established theoretically. In practice, it seems to perform reasonably well but can sometimes exhibit large variability, especially when the sample sizes are small. When the number of principal components K is >1, the condition needed to ensure election of the correct variables is very difficult to verify in practice. It would be useful to explore other approaches for multiple components. With large numbers of highly correlated features, it is important to learn when we can and cannot isolate the important underlying features.
- The latent variable model used in this article is a reasonable starting point, but may not be realistic in practice. One might have a situation in which the response is marginally independent of the active predictors, and yet jointly dependent on them. Another situation would have all predictors marginally dependent on the response, whereas one set is independent of the response given the rest of the predictors. In these cases, the supervised principal component procedure would fail.
- The response model considered in this article is a simple linear or (generalized linear) model. It would be useful to examine whether supervised principal components can perform well when the response is a more complex function of the latent factors. One could use linear correlation thresholding (as described in the article) but then use a spline basis (instead of a linear basis) in the response model. In practice, we have found that a natural cubic spline basis with two or three knots can capture simple nonlinearities in the response function. We have not yet implemented this in our software, but plan to explore it further.
- A key aspect of the method is the preselection of features according to their correlation with the outcome. This alleviates the effect of a larger number of noisy features on the prediction model. It is likely that this preselection can be used effectively with other regression methods, such as partial least squares and ridge regression. We have focused on supervised principal components because of its striking simplicity.
- Further work is needed in the Cox model setting, because our results there are not yet rigorous.
- Supervised principal components is attractive because of its simplicity. However, as mentioned earlier, other methods, such as partial least squares and the lasso, could be applied after thresholding of the genes. These might also perform well and are worth investigating. In addition, there are other closely related methods that should be considered and compared with supervised principal components. These include the sliced inversed regression approach of Duan and Li (1991) and Li (1992) and the sufficient dimension reduction approaches used by Chiaromonte, Cook, and Li (2002) and Cook (2004). The first of the latter two articles relates to covariate adjustment, whereas the second treats predictor selection. Related applications to microarray data include those of Chiaromonte and Martinelli (2002), Antoniadis, Lambert-Lacroix, and Leblanc (2003), and Bura and Pfeiffer (2003).

An application of supervised principal components in a medical setting is discussed in Zhao et al. (2006). We have written Excel (PAM) and R language packages (superpc) implementing supervised principal components for survival and regression data. These are freely available on Tibshirani's website (*http://www-stat.stanford.edu/~tibs/superpc*).

APPENDIX: OUTLINE OF PROOF OF THEOREM 2

As already stated, we prove the result under assumptions A1 and A2. To prove Theorem 2, we need the following lemma about the perturbation of eigenvectors of a symmetric matrix under symmetric perturbation.

Lemma A.1. For some $m \in \mathbb{N}$, let $\mathbf{A}$ and $\mathbb{B}$ be two symmetric $m \times m$ matrices. Let the eigenvalues of matrix $\mathbf{A}$ be denoted by $\lambda_1(\mathbf{A}) \geq \cdots \geq \lambda_m(\mathbf{A})$. Set $\lambda_0(\mathbf{A}) = \infty$ and $\lambda_{m+1}(\mathbf{A}) = -\infty$. For any $r \in \{1, \ldots, m\}$, if $\lambda_r(\mathbf{A})$ is an eigenvalue of multiplicity 1, that is, $\lambda_{r-1}(\mathbf{A}) > \lambda_r(\mathbf{A}) > \lambda_{r+1}(\mathbf{A})$, then denoting by $\mathbf{p}_r$ the eigenvector associated with the rth largest eigenvalue,

$$\mathbf{p}_r(\mathbf{A}+\mathbf{B}) - \operatorname{sign}\big(\mathbf{p}_r(\mathbf{A}+\mathbf{B})^T \mathbf{p}_r(\mathbf{A})\big)\mathbf{p}_r(A) = -\mathbf{H}_r(\mathbf{A})\mathbf{B}\mathbf{p}_r(\mathbf{A}) + \mathbf{R}_r, \quad \text{(A.1)}$$

where $\mathbf{H}_r(\mathbf{A}) := \sum_{s \neq r} (\lambda_s(\mathbf{A}) - \lambda_r(\mathbf{A}))^{-1} \mathbf{P}_{\mathcal{E}_s}(\mathbf{A})$ and $\mathbf{P}_{\mathcal{E}_s}(\mathbf{A})$ denotes the projection matrix onto the eigenspace $\mathcal{E}_s$ corresponding to the eigenvalue $\lambda_s(\mathbf{A})$, (possibly multidimensional). Further, the residual

term $\mathbf{R}_r$ can be bounded by

$$\|\mathbf{R}_r\| \le \begin{cases} \|\mathbf{H}_r(\mathbf{A})\mathbf{B}\mathbf{p}_r(\mathbf{A})\|\left[\frac{2\Delta_r(1+\Delta_r)}{1-2\Delta_r(1+\Delta_r)} + \frac{\|\mathbf{H}_r(\mathbf{A})\mathbf{B}\mathbf{p}_r(\mathbf{A})\|}{(1-2\Delta_r(1+\Delta_r))2}\right] \\ \qquad \text{if } \Delta_r < \frac{\sqrt{5}-1}{2} \\ 10\Delta_r^2 \quad \text{always,} \end{cases} \tag{A.2}$$

where

$$\Delta_r = \frac{\|\mathbf{B}\|_2}{\min_{1\le s\ne r\le m}|\lambda_s(\mathbf{A}) - \lambda_r(\mathbf{A})|}. \tag{A.3}$$

Proof. This follows from a refinement of the argument given in the proof of lemma A.1 of Kneip and Utikal (2001).

Lemma A.1 gives a first-order expansion of the eigenvector of a perturbed matrix. Now we can take as matrix $\mathbf{A}$ the matrix $\mathbf{\Sigma}_1$, the covariance matrix of $\{\mathbf{X}_j : j \in \mathcal{P}\}$, and then we can take $\mathbf{B}$ to be the difference $\mathbf{S}_1 - \mathbf{\Sigma}_1$, where $\mathbf{S}_1 = \frac{1}{N}\mathbf{X}_1^T\mathbf{X}_1$. Notice that

$$\mathbf{H}_r(\mathbf{\Sigma}_1) = \sum_{1\le s\ne r\le M} \frac{1}{\lambda_s - \lambda_r}\boldsymbol{\theta}_s\boldsymbol{\theta}_s^T - \frac{1}{\lambda_r}\left(\mathbf{I} - \sum_{s=1}^{M}\boldsymbol{\theta}_s\boldsymbol{\theta}_s^T\right),$$

and

$$\mathbf{p}_r(\mathbf{\Sigma}_1) = \boldsymbol{\theta}_r.$$

By Lemma A.1, we only need probabilistic bounds for the quantities $\|\mathbf{H}_r(\mathbf{A})\mathbf{B}\mathbf{p}_r(\mathbf{A})\|$ and $\|\mathbf{S}_1 - \mathbf{\Sigma}_1\|$. The first involves some lengthy but straightforward calculation, and for the second we need a bound for the term $\|\frac{1}{N}\mathbf{W}^T\mathbf{W} - \mathbf{I}\|$. For this, we use the following lemma, the proof of which uses large deviation inequalities for quadratic forms of Gaussian random variables.

Lemma A.2. Suppose that $n, L \to \infty$ s.t. $L/n \to 0$. Let $\mathbf{Z}$ be denote an $L \times n$ matrix with iid N(0, 1) entries. Denote by l_1 and l_L the largest and the smallest eigenvalues of $\frac{1}{n}\mathbf{Z}\mathbf{Z}^T$. We have

$$\mathbb{P}\left(l_1 - 1 > 2\left(\sqrt{\log(n/L)} + \pi\right)\sqrt{\frac{L}{n}}\right) \le \Delta_{nL}^{-1}(L/n)^{(L/2)(1+o(1))}(1 + o(1))$$

and

$$\mathbb{P}\left(l_L - 1 < -2\left(\sqrt{\log(n/L)} + \pi\right)\sqrt{\frac{L}{n}}\right) \le (1 + \Delta_{nL}^{-1})(L/n)^{(L/2)(1+o(1))}(1 + o(1)),$$

where $\Delta_{nL} = \sqrt{2L}\sqrt{\log(n/L)}$.

Proof. For $a \in \mathbb{S}^L$ (L-dimensional unit sphere), define $g(\mathbf{a}, \mathbf{Z}) = \frac{1}{n}\mathbf{a}^T\mathbf{Z}\mathbf{Z}^T\mathbf{a}$. As a function of $\mathbf{a}$, $g(\mathbf{a}, \mathbf{Z})$ is Lipschitz-1 with Lipschitz constant $2\|\frac{1}{n}\mathbf{Z}\mathbf{Z}^T\| = 2l_1$. This is because $g(\mathbf{a}, \mathbf{Z}) - g(\mathbf{b}, \mathbf{Z}) = (\mathbf{a} - \mathbf{b})^T\frac{1}{n}\mathbf{Z}\mathbf{Z}^T(\mathbf{a} + \mathbf{b})$. Let $\mathcal{F}_\delta$ be a minimal covering of the sphere $\mathbb{S}^L$ by balls of radius $\delta < 1$. Then a simple argument shows that

$$\left(\frac{1}{\delta}\right)^{L-1} \le |\mathcal{F}_\delta| \le 2\left(\frac{\pi}{\delta}\right)^{L-1}. \tag{A.4}$$

For a proof of this result, refer to proposition 2 of von Luxburg, Bousquet, and Schölkopf (2002). Now, by definition,

$$l_1 = \max_{\mathbf{a}\in\mathbb{S}^L} g(\mathbf{a}, \mathbf{Z}) \qquad \text{and} \qquad l_L = \min_{\mathbf{a}\in\mathbb{S}^L} g(\mathbf{a}, \mathbf{Z}).$$

Hence by the covering of $\mathbb{S}^L$ by spheres of radius δ centered at points in $\mathcal{F}_\delta$ and the Lipschitz bound on g, it follows that

$$\begin{aligned} &\max_{\mathbf{a}\in\mathcal{F}_\delta} g(\mathbf{a}, \mathbf{Z}) \le l_1 \le \max_{\mathbf{a}\in\mathcal{F}_\delta} g(\mathbf{a}, \mathbf{Z}) + 2\delta l_1 \qquad \text{and} \\ &\min_{\mathbf{a}\in\mathcal{F}_\delta} g(\mathbf{a}, \mathbf{Z}) - 2\delta l_1 \le l_L \le \min_{\mathbf{a}\in\mathcal{F}_\delta} g(\mathbf{a}, \mathbf{Z}). \end{aligned} \tag{A.5}$$

Now we use the fact that if $\mathbf{a} \in \mathbb{S}^L$ and entries of the $L \times n$ matrix $\mathbf{Z}$ are iid N(0, 1), then $\mathbf{Z}^T\mathbf{a}$ has iid N(0, 1) entries and so $g(\mathbf{a}, \mathbf{Z}) \sim \chi^2_{(n)}/n$. Finally, we recall some large-deviation inequalities for χ^2 random variables. Johnstone (2001) showed that

$$\mathbb{P}\left(\chi^2_{(n)} > n(1+\epsilon)\right) \le e^{-3n\epsilon^2/16}, \qquad 0 < \epsilon < \frac{1}{2}, \tag{A.6}$$

$$\mathbb{P}\left(\chi^2_{(n)} < n(1-\epsilon)\right) \le e^{-n\epsilon^2/4}, \qquad 0 < \epsilon < 1, \tag{A.7}$$

and

$$\mathbb{P}\left(\chi^2_{(n)} > n(1+\epsilon)\right) \le \frac{\sqrt{2}}{\epsilon\sqrt{n}}e^{-n\epsilon^2/4} \qquad 0 < \epsilon < n^{1/16}, n \ge 16. \tag{A.8}$$

Let $0 < t < 1$ and $0 < \delta < \frac{t}{2(1+t)}$; then, by the first inequality in (A.5), for $n \ge 16$,

$$\begin{aligned} \mathbb{P}(l_1 - 1 > t) &\le \mathbb{P}\left(\max_{\mathbf{a}\in\mathcal{F}_\delta} g(\mathbf{a}, \mathbf{Z}) \cdot (1-2\delta)^{-1} - 1 > t\right) \\ &= \mathbb{P}\left(\max_{\mathbf{a}\in\mathcal{F}_\delta}[g(\mathbf{a}, \mathbf{Z}) - 1] > t(1-2\delta) - 2\delta\right) \\ &\le |\mathcal{F}_\delta|\mathbb{P}\left(\frac{\chi^2_{(n)}}{n} - 1 > t(1-2\delta) - 2\delta\right) \\ &\le 2\left(\frac{\pi}{\delta}\right)^{N-1}\frac{\sqrt{2}}{\sqrt{n}(t(1-2\delta) - 2\delta)} \\ &\qquad \times \exp\left[-\frac{n}{4}(t(1-2\delta) - 2\delta)^2\right], \end{aligned}$$

by (A.4) and (A.8). Now choose $\delta := \delta_n = \pi\sqrt{L/n}$ and $t := t_n = (2\sqrt{\log(n/L)} + 2\pi)\sqrt{\frac{L}{n}}$, which satisfy the restrictions for n sufficiently large. Then

$$\begin{aligned} t(1-2\delta) - 2\delta &= 2\sqrt{\log\left(\frac{n}{L}\right)}\sqrt{\frac{L}{n}}\left(1 - 2\pi\sqrt{\frac{L}{n}}\right) - 4\pi^2\frac{L}{n} \\ &= 2\sqrt{\log\left(\frac{n}{L}\right)}\sqrt{\frac{L}{n}}(1 - \varepsilon_n), \end{aligned}$$

where $\varepsilon_n = 2\pi\sqrt{L/n}(1 + \pi(\log(\frac{n}{L}))^{-1/2})$. Because $\varepsilon_n = o(1)$ as $n \to \infty$,

$$\begin{aligned} &\mathbb{P}\left(l_1 - 1 > \left(2\sqrt{\log\left(\frac{n}{L}\right)} + 2\pi\right)\sqrt{\frac{N}{n}}\right) \\ &\le \frac{\sqrt{2}}{2\sqrt{L}\sqrt{\log(n/L)}(1-\varepsilon_n)}\left(\frac{n}{L}\right)^{(L-1)/2} \\ &\qquad \times \exp\left[-L\log\left(\frac{n}{L}\right)(1-\varepsilon_n)^2\right] \\ &= \frac{1}{\sqrt{2L}\sqrt{\log(n/L)}(1+o(1))}\left(\frac{L}{n}\right)^{(L/2)(1+o(1))} \qquad \text{as } n \to \infty. \end{aligned} \tag{A.9}$$

Next, using the second inequality in (A.5), for $t = t_n$ and $\delta = \delta_n$ as chosen earlier,

$$\begin{aligned} &\mathbb{P}(l_L - 1 < -t) \\ &\le \mathbb{P}\left(\min_{\mathbf{a}\in\mathcal{F}_\delta} g(\mathbf{a}, \mathbf{Z}) - 1 < -t + 2\delta l_1\right) \\ &\le \mathbb{P}\left(\min_{\mathbf{a}\in\mathcal{F}_\delta}[g(\mathbf{a}, \mathbf{Z}) - 1] < -t + 2\delta(1+t)\right) + \mathbb{P}(l_1 - 1 > t) \\ &\le |\mathcal{F}_\delta|\mathbb{P}\left(\frac{\chi^2_{(n)}}{n} - 1 < -(t(1-2\delta) - 2\delta)\right) + \mathbb{P}(l_1 - 1 > t). \end{aligned}$$

Then, using (A.4) and (A.7) exactly as before to bound the first term on the right side and then using (A.9), we get, as $n \to \infty$,

$$\mathbb{P}\left(l_L - 1 < -\left(2\sqrt{\log\left(\frac{n}{L}\right)} + 2\pi\right)\sqrt{\frac{L}{n}}\right) \leq \left(1 + \frac{1}{\sqrt{2L}\sqrt{\log(n/L)}(1+o(1))}\right)\left(\frac{L}{n}\right)^{(L/2)(1+o(1))}. \quad \text{(A.10)}$$

[Received November 2004. Revised May 2005.]

REFERENCES

Alter, O., Brown, P., and Botstein, D. (2000), "Singular Value Decomposition for Genome-Wide Expression Data Processing and Modeling," *Proceedings of the National Academy of Sciences USA*, 97, 10101–10106.

Antoniadis, A., Lambert-Lacroix, S., and Leblanc, F. (2003), "Effective Dimension Reduction Methods for Tumor Classification Using Gene Expression Data," *Bioinformatics*, 19, 563–570.

Baik, J., and Silverstein, J. W. (2004), "Eigenvalues of Large Sample Covariance Matrices of Spiked Population Models," *arXiv:math.ST*.

Bair, E., and Tibshirani, R. (2004), "Semi-Supervised Methods to Predict Patient Survival From Gene Expression Data," *PLoS Biology*, 2, 511–522.

Beer, D. G., Kardia, S. L., Huang, C.-C., Giordano, T. J., Levin, A. M., Misek, D. E., Lin, L., Chen, G., Gharib, T. G., Thomas, D. G., Lizyness, M. L., Kuick, R., Hayasaka, S., Taylor, J. M., Iannettoni, M. D., Orringer, M. B., and Hanash, S. (2002), "Gene-Expression Profiles Predict Survival of Patients With Lung Adenocarcinoma," *Nature Medicine*, 8, 816–824.

Bullinger, L., Döhner, K., Bair, E., Fröhling, S., Schlenk, R., Tibshirani, R., Döhner, H., and Pollack, J. R. (2004), "Gene Expression Profiling Identifies New Subclasses and Improves Outcome Prediction in Adult Myeloid Leukemia," *New England Journal of Medicine*, 350, 1605–1616.

Bura, E., and Pfeiffer, R. M. (2003), "Graphical Methods for Class Prediction Using Dimension Reduction Techniques on DNA Microarray Data," *Bioinformatics*, 19, 1252–1258.

Chen, S. S., Donoho, D. L., and Saunders, M. A. (1998), "Atomic Decomposition by Basis Pursuit," *SIAM Journal on Scientific Computing*, 20, 33–61.

Chiaromonte, F., Cook, R., and Li, B. (2002), "Sufficient Dimension Reduction in Regressions With Categorical Predictors," *The Annals of Statistics*, 30, 475–497.

Chiaromonte, F., and Martinelli, J. (2002), "Dimension Reduction Strategies for Analyzing Global Gene Expression Data With a Response," *Mathematical Biosciences*, 176, 123–144.

Cook, R. (2004), "Testing Predictor Contributions in Sufficient Dimension Reduction," *The Annals of Statistics*, 32, 1062–1092.

Duan, N., and Li, K.-C. (1991), "Slicing Regression: A Link-Free Regression Method," *The Annals of Statistics*, 19, 505–530.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004), "Least Angle Regression," *The Annals of Statistics*, 32, 407–499.

Frank, I., and Friedman, J. (1993), "A Statistical View of Some Chemometrics Regression Tools" (with discussion), *Technometrics*, 35, 109–148.

Ghosh, D. (2002), "Singular Value Decomposition Regression Models for Classification of Tumors From Microarray Experiments," *Pacific Symposium on Biocomputing*, 7, 18–29.

Hastie, T., and Tibshirani, R. (2003), "Efficient Quadratic Regularization for Expression Arrays," technical report, Stanford University.

Hastie, T., Tibshirani, R., Botstein, D., and Brown, P. (2001), "Supervised Harvesting of Expression Trees," *Genome Biology*, 2, 1–12.

Hastie, T., Tibshirani, R., Eisen, M., Alizadeh, A., Levy, R., Staudt, L., Botstein, D., and Brown, P. (2000), "Identifying Distinct Sets of Genes With Similar Expression Patterns via 'Gene Shaving'," *Genome Biology*, 1, 1–21.

Hastie, T., Tibshirani, R., and Friedman, J. (2001), *The Elements of Statistical Learning; Data Mining, Inference and Prediction*, New York: Springer-Verlag.

Hi, H., and Gui, J. (2004), "Partial Cox Regression Analysis for High-Dimensional Microarray Gene Expression Data," *Bioinformatics*, 5, 1208–1215.

Hoerl, A. E., and Kennard, R. (1970), "Ridge Regression: Biased Estimation for Nonorthogonal Problems," *Technometrics*, 12, 55–67.

Huffel, S. V., and Lemmerling, P. (eds.) (2002), *Total Least Squares and Errors-in-Variables Modeling*, Dordrecht: Kluwer.

Jiang, H., Deng, Y., Chen, H.-S., Tao, L., Sha, Q., Chen, J., Tsai, C.-J., and Zhang, S. (2004), "Joint Analysis of Two Microarray Gene-Expression Data Sets to Select Lung Adenocarcinoma Marker Genes," *BMC Bioinformatics*, 5, 1–12.

Johnstone, I. (2001), "Chi-Square Oracle Inequalities," in *Festschrift for William R. van Zwet*, eds. M. de Gunst, C. Klaassen, and A. van der Waart, Hayward, CA: IMS, pp. 399–418.

Johnstone, I., and Lu, A. Y. (2006), "Sparse Principal Components Analysis," *Journal of the American Statistical Association*, to appear.

Kneip, A., and Utikal, K. J. (2001), "Inference for Density Families Using Functional Principal Component Analysis," *Journal of the American Statistical Association*, 96, 519–542.

Li, K.-C. (1992), "Sliced Inverse Regression for Dimension Reduction" (with discussion), *Journal of the American Statistical Association*, 86, 316–342.

Lu, A. Y. (2002), "Sparse Principal Components Analysis for Functional Data," technical report, Stanford University.

Mardia, K., Kent, J., and Bibby, J. (1979), *Multivariate Analysis*, New York: Academic Press.

Miller, R. G. (1986), *Beyond Anova: Basics of Applied Statistics*, New York: Wiley.

Nguyen, D., and Rocke, D. (2002), "Partial Least Squares Proportional Hazard Regression for Application to DNA Microarrays," *Bioinformatics*, 18, 1625–1632.

Paul, D. (2005), "Nonparametric Estimation or Parametric Components," Ph.D. thesis, Stanford University.

Rosenwald, A., Wright, G., Chan, W. C., Connors, J. M., Campo, E., Fisher, R. I., Gascoyne, R. D., Muller-Hermelink, H. K., Smeland, E. B., and Staudt, L. M. (2002), "The Use of Molecular Profiling to Predict Survival After Chemotherapy for Diffuse Large B-Cell Lymphoma," *The New England Journal of Medicine*, 346, 1937–1947.

Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society*, Ser B., 58, 267–288.

——— (1997), "The Lasso Method for Variable Selection in the Cox Model," *Statistics in Medicine*, 16, 385–395.

Tibshirani, R., Hastie, T., Narasimhan, B., and Chu, G. (2001), "Diagnosis of Multiple Cancer Types by Shrunken Centroids of Gene Expression," *Proceedings of the National Academy of Science*, 99, 6567–6572.

van't Veer, L. J., Dai, H., van de Vijver, M. J., He, Y. D., Hart, A. A. M., Mao, M., Peterse, H. L., van der Kooy, K., Marton, M. J., Witteveen, A. T., Schreiber, G. J., Kerkhoven, R. M., Roberts, C., Linsley, P. S., Bernards, R., and Friend, S. H. (2002), "Gene Expression Profiling Predicts Clinical Outcome of Breast Cancer," *Nature*, 415, 530–536.

von Luxburg, U., Bousquet, O., and Schölkopf, B. (2002), "A Compression Approach to Support Vector Model Selection," technical report, Max-Planck-Institut für Biologische Kybernetik.

Wold, H. (1975), "Soft Modelling by Latent Variables: The Nonlinear Iterative Partial Least Squares (NIPALS) Approach," in *Perspectives in Probability and Statistics, in Honor of M. S. Bartlett*, ed. J. Gani, pp. 117–144.

Zhao, H., Ljungberg, B., Grankvist, K., Rasmuson, T., Tibshirani, R., and Brooks, J. (2006), "Gene Expression Profiling Predicts Survival in Conventional Renal Cell Carcinoma," *PLoS Medicine*, 3, 1–10.