Modeling the relationship between LVAD support time and gene expression changes in the human heart by penalized partial least squares

Xiaohong Huang¹, Wei Pan¹*, Soon Park², Xinqiang Han³, Leslie W. Miller³, Jennifer Hall³

Revised October 2003

¹ Division of Biostatistics, School of Public Health, University of Minnesota
² Cardiothoracic Surgery, University of Minnesota, Fairview Medical Center
³ Cardiovascular Division, Department of Medicine, Medical School, University of Minnesota
* Corresponding author. A460 Mayo Building (MMC 303), A460 Mayo Building (MMC 303), Minneapolis, MN 55455-0378, USA. Email: weip@biostat.umn.edu, phone: (612)626-2705, fax: (612)626-0660.
Modeling the relationship between LVAD support time and gene expression changes in the human heart by penalized partial least squares

Abstract
Heart failure affects more than 20 million people in the world. Heart transplantation is the most effective therapy, but the number of eligible patients far outweighs the number of available donor hearts. The left mechanical ventricular assist device (LVAD) has been developed as a successful substitution therapy that aids the failing ventricle while a patient is waiting for the donor heart. We obtained genomics data from paired human heart samples harvested at the time of LVAD implant and explant. The heart failure patients in our study were supported by the LVAD for various periods of time. The goal of this study is to model the relationship between the time of LVAD support and gene expression changes. To serve the purpose, we propose a novel penalized partial least squares (PPLS) method to build a regression model. Compared with partial least squares (PLS) and Breiman’s random forest method, PPLS gives the best prediction results for the LVAD data.

Key words: Heart failure; Left ventricular assist device (LVAD); Partial least square (PLS); Permutation; Penalized regression; Penalized partial least squares (PPLS).
Introduction

Heart failure is a condition that affects more than 20 million people around the world (nearly 5 million in the USA). Approximately 50% of people with heart failure die within 5 years of being diagnosed. Heart transplantation is currently the most effective therapy for heart failure (Hunt et al., 2002), but the number of patients far outweighs the number of donor hearts. The left ventricular assist device (LVAD) is a mechanical device that replaces the pumping of the ventricle in patients with severe refractory heart failure and has been shown to significantly decrease mortality (Altemose et al., 1997; Bruckner et al., 2001; Dipla et al., 1998; Levin et al., 1996; McCarthy et al., 1994; McCarthy et al., 1995a; McCarthy et al., 1995b; McCarthy and Smith, 2002; Rose et al., 2001). The device affords near total chamber decompression and reduction of wall stress, and achieves the most rapid and complete reverse remodeling of any model to date (Altemose et al., 1997; Rose et al., 2001). The response of the failing heart to mechanical unloading is not entirely clear, which limits our understanding of how best to utilize the device. We hope to use a genomic approach to better understand this response. Insertion of the device requires removal of a core of myocardial tissue (estimate 2 grams) at the apex of the heart to insert the drainage cannula. This tissue can be paired with tissue obtained from the explanted heart at the time of subsequent transplantation to define changes in gene expression in response to mechanical unloading. These paired samples provide a unique opportunity to define the critical genes regulated by workload and wall stress leading to myocardial remodeling. The paired sample design of this study is especially useful in reducing patient variability in gene expression and isolating those genes specifically regulated by mechanical unloading. Our goal is to study the temporal relationship between the time of the LVAD support and the gene expression changes. As a first step, our objective is to investigate whether we can predict the LVAD support time by gene expression changes. For this purpose, we have built a statistical regression model. However, because of the presence of a small number of samples as well as a large number of genes, many classical statistical methods cannot be directly applied. By combining the idea of the partial least squares (PLS) with that of penalized regression, we propose a novel
method, penalized partial least squares (PPLS), to build such a model. A permutation test is also proposed to assess the significance of such a model. The performance of PPLS method was compared to that of several other methods.

**Statistical methods**

A very special feature of oligonucleotide microarray gene expression data is the large number of genes (i.e. large \( p \)) and a small number of arrays (i.e. small \( n \)). In statistical words, we have a “large \( p \) and small \( n \)” problem (West, 2002). In many existing methods derived for the usual “\( p \leq n \)” problem, we can fit a multiple linear regression model by including all the genes and choose the model based on the statistical significance of estimated regression coefficients. For instance, when the number of genes \( p \) greatly exceeds the number of samples \( n \), it is impossible to fit a multiple linear regression model using ordinary least squares (OLS) by including all the genes. Even when \( p < n \) but \( p \) is close to \( n \), if we use OLS to fit a multiple linear regression model with all the genes, we are likely to get a model that fits the current data almost perfectly but that will fail to predict well for new data; this is the so-called over-fitting problem. Moreover, for some modeling methods that can handle larger numbers of genes, such as PLS, we cannot use the statistical significance to select genes because the significance test of the estimated regression coefficients is not available. Some form of variable selection, most notably based on a univariate gene ranking method, may be taken prior to model fitting, to greatly reduce the number of genes to be used in the model. However, such methods may not be optimal. On the other hand, partial least squares (PLS) is particularly suited for constructing a linear model under the situation with \( p \gg n \) (Hoskuldsson, 1988; Helland, 1988; Gaithwaite, 1994). Using PLS, we can fit a linear model with the presence of a large number of genes. However, we know a priori that most of the genes may not need to be included in the model. Because there is still no method existing to test the significance of the estimated regression coefficients in a PLS-fitted linear model, we need to seek other ways to trim a possibly too large model. Penalized regression (e.g. Tibshirani, 1996; Fan and Li, 2001) provides a viable approach; it shrinks the estimated coefficients to zero to realize variable selection in a multivariate model. We propose a new method, penalized partial least
squares method (PPLS), which uses PLS to fit a linear model and uses penalized regression
to do gene selection automatically.

**Penalized PLS**

**Model fitting**

Suppose that $X_{ij}$ is the expression change of gene $i$ and subject $j$, and $Y_j$ is the LVAD
support time of subject $j$, for $i = 1, \ldots, p$ and $j = 1, \ldots, n$. We denote column vectors
$X_i = (X_{i1}, \ldots, X_{im})^T$ and $Y = (Y_1, \ldots, Y_n)^T$, and denote the observed values of $Y$ and $X_i$
by column vectors $y = (y_1, \ldots, y_n)^T$ and $x_i = (x_{i1}, \ldots, x_{in})^T$ for $i = 1, \ldots, p$. Suppose that
each $X_i$ has been standardized to have sample mean 0 and standard deviation 1.

In PLS, we estimate the PLS components $T_1, \ldots, T_q$ (for details see Appendix; also see,
e.g., Garthwaite 1994; Huang and Pan 2003) and fit the OLS model

$$\hat{Y} = \gamma_0 + \gamma_1 T_1 + \gamma_2 T_2 + \cdots + \gamma_q T_q.$$  \hspace{1cm} (1)

Since each of the PLS components $T_1, \ldots, T_q$ is a linear combination of the variables $X_i$, we
can rewrite model (1) as

$$\hat{Y} = b_0 + \sum_{i=1}^{p} b_i X_i$$  \hspace{1cm} (2)

which established a linear relationship between the LVAD support time and the change of
gene expression.

Model (2) in general contains all the genes being used at the beginning. With a huge
number of genes, it is likely that many of them are not needed and they contribute nothing
but noise to the model. Hence it is desirable to shrink some $b_i$’s to 0, eliminating the corres-
ponding genes from the model. It has been shown in the statistics literature that penalized
regression with soft-thresholding is promising (Donoho and Johnstone, 1994; Donoho, 1995).

It shrinks some coefficients to zero through a continuous shrinkage process to select variables.
In the current context, the soft thresholding works as follows:

$$\hat{b}_i = \text{sign}(b_i)(|b_i| - \Delta)_+ \quad i = 1, \ldots, p,$$
where $\Delta$ is a shrinkage parameter to be determined (e.g. by cross-validation), $f_+ = f$ if $f > 0$ and $f_+ = 0$ if $f \leq 0$. We define a new explanatory variable $T = \sum_{i=1}^{p} \hat{b}_i X_i$ and fit the OLS regression model

$$
\hat{Y} = \theta_0 + \theta_1 T = \rho_0 + \sum_{i=1}^{p} \rho_i X_i
$$

(3)

Where $\rho_i = \theta_1 \hat{b}_i$ and $\rho_0 = \theta_0$. Note that gene $i$ will not contribute to the model if $\hat{b}_i = 0$.

There are two intuitive arguments motivating the use of shrinkage. First, gene expression data contain a lot of noises, and not all the genes are informative. Second, because we have a small sample size and a large number of variables, model regularization through parameter shrinkage can be productive, using the celebrated bias-variance trade-off argument. Although we did not compare soft thresholding with other shrinkage schemes, it seems that in the statistics literature, it is preferred by many. Furthermore, we found it working well here.

**Parameter selection**

Both the number of components $q$ and the shrinkage parameter $\Delta$ are selected using leave-one-out cross-validation (CV) with a training data set. Without loss of generality, suppose that there are $n$ samples in the training data, and we will only use the training data to select parameters in the following. First, we hold out sample $j = 1$, build a PLS model (with $q$ components) based on the remaining $n-1$ samples; we repeat this procedure for $j = 1, \ldots, n$ to build $n$ models. We set the range of the shrinkage parameter $\Delta$ based on the regression coefficients of these $n$ models. For each $\Delta$, we shrink the coefficients of a model, say the one with sample $j$ being left out, calculate the new explanatory variable $T(j)$ and fit model $\hat{Y}(j) = \theta_0 + \theta_1 T(j)$, where the subscript ($j$) means that the model is built without using sample $j$. The predictive power of a model at each $\Delta$ is measured by the value of predictive mean residual sum of squares (PMRSS),

$$
PMRSS = \frac{1}{n} \sum_{j=1}^{n} (Y_j - \hat{Y}(j))^2,
$$

(4)

where $\hat{Y}(j)$ is the predicted value of $Y_j$, the held sample. Let $PMRSS_q$ be the minimum $PMRSS$ across all $\Delta$’s with the PLS model containing $q$ components. The above procedures
are repeated with \( q = 1, \ldots, 8 \). Note that because we only have 14 samples, it seems a linear model with 8 components is large enough. The selected shrinkage parameter \( \hat{\Delta} \) is the one corresponding to \( \hat{PMR}SS_q = \min(\hat{PMR}SS_1, \ldots, \hat{PMR}SS_8) \). Finally, we fit a PLS linear model using all the \( n = 14 \) samples with the number of components \( \hat{q} \). The estimated regression coefficients are shrunken using the soft thresholding with the shrinkage parameter \( \hat{\Delta} \), leading to the final model (3). Usually in the final model there are only a few genes whose estimated coefficients are not shrunken to zero.

**Model comparison**

Note that the training error \( \hat{PMR}SS_q \) is a biased estimate of the true PMRSS for future data (Ambroise and McLachlan, 2002; Simon et al., 2003). If we have a separate test data set, the above fitted model can be applied to the test data to obtain a valid estimate of PMRSS. In practice, it is often to split a given data set into two parts, a training data set and a test data set. Here, because of the small sample size \( n = 14 \), we again use leave-one-out CV to estimate PMRSS. The formula is the same as (4) except that here, \( \hat{Y}_{(j)} \) is the predicted value of \( Y_j \) based on the PPLS model built on the data excluding sample \( j \); that is, we use the remaining \( n - 1 \) samples as the training data to select parameters for PPLS, as described in the previous section. Therefore, two nested levels of leave-one-out-CV are involved.

This procedure with honest CV is critical and applicable to any adaptive methods (i.e. methods with parameter selection). For example, as described later, this same procedure is applied to select the number of components and obtain an unbiased estimate of PMRSS for PLS. Such unbiased estimates of PMRSS can be used as criteria for model selection: we pick up the model with the smallest estimated PMRSS. In the following, we only present unbiased estimates of PMRSS.

**Results**

**Data**

Our microarray data set contained a total of 14 paired human heart samples obtained at the time of LVAD implant (pre-samples) and LVAD explant (post-samples). Initial analysis was completed in MAS 5.0 (MicroArray Suites 5.0). If the gene was considered absent in
both the pre- and post-samples, the gene was eliminated from the analysis. Thus, final
analysis included 16,383 probes. Since we were interested in the change of gene expression,
we took the difference of the expression levels between the pair of pre- and post-samples.
We centered within each sample by subtracting the mean of the sample. In order to make
different genes comparable, we standardized the profile of expression changes of each gene
such that it has mean 0 and variance 1. We denoted the standardized expression changes
of gene \( i \) as our covariate \( X_i \). The LVAD support time ranges from 7 days to as long as
681 days, with the mean 175.9 days, and the 25%, 50% and 75% quantiles 37.75, 104.5 and
211 days respectively. The log time (in days) of each patient under LVAD support is our
response variable \( Y \); we took a log transformation to alleviate the skewness of the support
times in the original scale.

**Univariate gene ranking**

To facilitate our later analysis with various numbers of genes included in starting models,
we have a preliminary ranking of all the genes using a usual F-statistic. This univariate
ranking is referred in the following, and obviously is by no means to be optimal. Specifically,
we fit a quadratic regression model for each gene \( i \) to take account of a possible nonlinear
relationship between the gene’s expression change and LVAD support time:

\[
\hat{Y} = \beta_{i0} + \beta_{i1}X_i + \beta_{i2}X_i^2, \quad i = 1, \ldots, p.
\]

We define the residual sum of squares

\[
RSS_i = \sum_{j=1}^{n} (Y_j - \hat{\beta}_{i0} - \hat{\beta}_{i1}X_{ij} - \hat{\beta}_{i2}X_{ij}^2)^2
\]

and

\[
RSS_{i0} = \sum_{j=1}^{n} (Y_j - \bar{Y})^2, \quad \bar{Y} = \frac{1}{n} \sum_{j=1}^{n} Y_j
\]

where \( \hat{\beta}_{i0}, \hat{\beta}_{i1} \) and \( \hat{\beta}_{i2} \) are the OLS estimates. The F-statistic for gene \( i \) is defined as

\[
F_i = \frac{(RSS_{i0} - RSS_i)/2}{RSS_i/(n - 3)}.
\]
We rank the genes based on their F-statistics; a gene with a larger F-statistic indicates a stronger relationship between its expression alterations and the LVAD support time, and therefore has a higher rank.

In leave-one-out CV, each gene is ranked according to its F-statistic calculated based on the \( n - 1 \) samples every time when a sample is taken out. Hence the top genes, say top 100, may vary every time within CV.

**Model fitting and model selection**

We start with various linear models by including different numbers of top ranked genes. For example, if our starting model included the top 100 ranked genes, we only use the top 100 ranked genes within cross-validation to carry out the PPLS fitting procedure; that is, during the procedure of estimating PMRSS (where the \( Y \) value of the left-out sample is predicted based on the PPLS model built on the remaining \( n - 1 \) samples through the second level CV), gene ranking and PPLS model fitting (including shrinkage parameter selection) are all based on the same \( n - 2 \) samples, while during the procedure of fitting the final model (when the number of components \( \hat{q} \) and shrinkage parameter \( \hat{\Delta} \) have been selected through the second level CV), gene ranking and PPLS model fitting are based on the same \( n - 1 \) samples. Finally, after the number of components \( \hat{q} \) and shrinkage parameter \( \hat{\Delta} \) are selected through the first level CV, the final model is fitted using the top 100 ranked genes based on all the \( n \) samples. We considered starting models including the top 50, 100, 200, 400, 800, 1600, 3200, 6400, 7400, 8400, 9400, 10400, 11400, 12400, 13400, 14400, 15400 and all 16383 genes respectively. The model fitting results, including the CV-estimated PMRSS and the numbers of genes contained in the final models, of using these starting models are listed in Table 1.

From Table 1, we can see that starting from different models, we obtain different PMRSS in cross-validation, ranging from 1.332 to 1.954. The model starting with the top 100 genes gives the smallest PMRSS = 1.332, and hence is the model of choice. The final PPLS model of choice contain only 21 genes (with non-zero regression coefficients). The identities of these 21 genes are listed in Table 2; the absolute value of the estimated regression coefficient \( |\hat{\beta}_i| \) for
gene $i$ indicates the partial correlation strength of the gene's expression change with LVAD support time.

We note that the genes with GenBank ID BG035151, BC003621, AL562282 were selected in all the 18 final PPLS models (starting with various numbers of top genes), while the gene with GenBank ID AF052100 was selected in the 17 final PPLS models, the genes AV702994, NM_001695, AA142942, AL390133 and AA580004 were selected by 15, 14, 12, 10 and 10 PPLS models respectively (Table 3). The number of selected genes in the 18 PPLS models are between 5 and 31. There are also many overlappings among the selected genes in the 18 PPLS models, in which sense we can say that the PPLS method is stable, at least for this data set.

Note that PPLS can be applied to the original data containing all the genes. Here we applied it with various numbers of top (univariately) ranked genes. One reason is that, as a by-product, we can show that the commonly used univariate gene ranking may not be effective (see also Table 3). As a comparison, gene selection through parameter shrinkage as implemented in PPLS is multivariate and can be more effective: it results in a much simpler model with a small number of genes while maintaining a high predictive ability (as measured by estimated PMRSS). It is clear that the few selected genes may not be among the top ones ranked by the univariate method.

**Statistical significance**

After obtaining our final PPLS model, we were interested in determining whether it is statistically significant or not; that is, whether the model was obtained by chance or not. We propose a permutation test to assess the significance of the model. Because $\hat{PMRSS}$ obtained in CV is a measure of the predictive ability of a model, we take it as the test statistic. We used only 200 permutations due to the intensive computing demand. In each permutation, the log LVAD support times are randomly permuted and assigned to the patients. A linear model with the top 100 genes within each CV iterate was refitted using PPLS to obtain its $\hat{PMRSS}$. The original $\hat{PMRSS}=1.332$ was compared to such obtained 200 $\hat{PMRSS}$’s. Note that ideally we should apply the whole PPLS procedure, including the
choice of the number of genes in the starting model, to permuted data; to save computing time, we fixed the number of top genes at 100, and this may over-estimate $P\tilde{M}RSS$ for permuted data.

Out of such 200 $P\tilde{M}RSS$’s from the permuted data, only 21 are smaller than 1.332, leading to a p-value=22/201=0.109. This indicates that, at the usual significance level 0.05, our final model does not establish a statistically significant relationship between the LVAD support time and gene expression changes; we suspect that this is related to a small sample size and thus reduced statistical power. Because of this, we should interpret our results in an exploratory way.

**Comparison with other methods**

To assess the performance of PPLS, we also applied PLS and Breiman’s random forest (Breiman, 2001) to the data. Breiman’s random forest, as bagging and boosting, is an ensemble or aggregated estimation method based on classification and regression trees. We used R implementation randomForest; all the parameters were set to default ones (for
Table 2: Genes and their corresponding regression coefficient estimates $\rho$ in the final PPLS model.

<table>
<thead>
<tr>
<th>GenBank</th>
<th>$\rho_i$</th>
<th>Gene name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BC003621</td>
<td>0.1780</td>
<td>Heterogeneous nuclear ribonucleoprotein U (scaffold attachment factor A)</td>
</tr>
<tr>
<td>BG035151</td>
<td>0.1299</td>
<td>Splicing factor proline/glutamine rich (polypyrimidine tract binding protein associated)</td>
</tr>
<tr>
<td>AA580004</td>
<td>0.1296</td>
<td>ADP-ribosylation factor 1</td>
</tr>
<tr>
<td>NM058004</td>
<td>0.1176</td>
<td>Ubiquitin specific protease 4 (proto-oncogene)</td>
</tr>
<tr>
<td>NM001695</td>
<td>0.0675</td>
<td>ATPase, H+ transporting, lysosomal 42kDa, V1 subunit C, isoform 1</td>
</tr>
<tr>
<td>BC002526</td>
<td>0.0509</td>
<td>Heat shock 70kDa protein 4</td>
</tr>
<tr>
<td>BC003129</td>
<td>0.0145</td>
<td>Non-POU domain containing, octamer-binding</td>
</tr>
<tr>
<td>AW664421</td>
<td>0.0050</td>
<td>Suppressor of cytokine signaling 5</td>
</tr>
<tr>
<td>BC005043</td>
<td>0.0023</td>
<td>Hypothetical protein MGC31957</td>
</tr>
<tr>
<td>AL662282</td>
<td>-0.3145</td>
<td>FAD-synthetase</td>
</tr>
<tr>
<td>AF052100</td>
<td>-0.1417</td>
<td>Vesicle-associated membrane protein 4</td>
</tr>
<tr>
<td>AV702994</td>
<td>-0.0755</td>
<td>HSPCO34 protein</td>
</tr>
<tr>
<td>AL390133</td>
<td>-0.0715</td>
<td>Homo sapiens mRNA; cDNA DKFZp5471147 (from clone DKFZp5471147)</td>
</tr>
<tr>
<td>A1935996</td>
<td>-0.0630</td>
<td>Nuclear receptor subfamily 4, group A, member 2</td>
</tr>
<tr>
<td>BC004517</td>
<td>-0.0584</td>
<td>Mitochondrial ribosomal protein L9</td>
</tr>
<tr>
<td>NM001512</td>
<td>-0.0428</td>
<td>Glutathione S-transferase A4</td>
</tr>
<tr>
<td>D83243</td>
<td>-0.0398</td>
<td>Nuclear protein, ataxia-telangiectasia locus</td>
</tr>
<tr>
<td>NM002874</td>
<td>-0.0397</td>
<td>RAD23 homolog B (S. cerevisiae)</td>
</tr>
<tr>
<td>NM006585</td>
<td>-0.0306</td>
<td>Chaperonin containing TCP1, subunit 8 (theta)</td>
</tr>
<tr>
<td>BF215487</td>
<td>-0.0299</td>
<td>Proteasome (prosome, macropain) 26S subunit, ATPase, 2</td>
</tr>
<tr>
<td>AA142942</td>
<td>-0.0211</td>
<td>ESTs, Weakly similar to neuronal thread protein (Homo sapiens) (H.sapiens)</td>
</tr>
</tbody>
</table>

11
Table 3: Genes selected by at least 10 final models.

<table>
<thead>
<tr>
<th>GenBank</th>
<th># of models</th>
<th>F rank</th>
<th>Gene name</th>
</tr>
</thead>
<tbody>
<tr>
<td>BG035151</td>
<td>18</td>
<td>1</td>
<td>Splicing factor proline/glutamine rich (polypyrimidine tract binding protein associated)</td>
</tr>
<tr>
<td>NM_001695</td>
<td>14</td>
<td>2</td>
<td>ATPase, H+ transporting, lysosomal 42kDa, V1 subunit C, isoform 1</td>
</tr>
<tr>
<td>NM_000696</td>
<td>12</td>
<td>4</td>
<td>Aldehyde dehydrogenase 9 family, member A1</td>
</tr>
<tr>
<td>AV702994</td>
<td>15</td>
<td>6</td>
<td>HSPCO34 protein</td>
</tr>
<tr>
<td>AL562282</td>
<td>18</td>
<td>11</td>
<td>FAD-synthetase</td>
</tr>
<tr>
<td>AA142942</td>
<td>12</td>
<td>12</td>
<td>ESTs, Weakly similar to neuronal thread protein (Homo sapiens) (H.sapiens)</td>
</tr>
<tr>
<td>BC003621</td>
<td>18</td>
<td>22</td>
<td>Heterogeneous nuclear ribonucleoprotein U (scaffold attachment factor A)</td>
</tr>
<tr>
<td>AL390133</td>
<td>10</td>
<td>55</td>
<td>Homo sapiens mRNA; cDNA DKFZp547I147 (from clone DKFZp547I147)</td>
</tr>
<tr>
<td>AF052100</td>
<td>17</td>
<td>82</td>
<td>Vesicle-associated membrane protein 4</td>
</tr>
<tr>
<td>AA580004</td>
<td>10</td>
<td>93</td>
<td>ADP-ribosylation factor 1</td>
</tr>
<tr>
<td>BE8677771</td>
<td>12</td>
<td>144</td>
<td>Hypothetical protein LOC220988</td>
</tr>
</tbody>
</table>
Table 4: Results from PLS and random forests (RF) starting from different numbers of top genes.

<table>
<thead>
<tr>
<th># of top genes</th>
<th>P(\hat{M}R)SS</th>
<th># of top genes</th>
<th>P(\hat{M}R)SS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PLS RF</td>
<td></td>
<td>PLS RF</td>
</tr>
<tr>
<td>50</td>
<td>1.903 1.895</td>
<td>8400</td>
<td>2.085 1.796</td>
</tr>
<tr>
<td>100</td>
<td>1.906 1.889</td>
<td>9400</td>
<td>2.034 1.796</td>
</tr>
<tr>
<td>200</td>
<td>1.927 1.855</td>
<td>10400</td>
<td>2.071 1.760</td>
</tr>
<tr>
<td>400</td>
<td>2.086 1.911</td>
<td>11400</td>
<td>2.073 1.886</td>
</tr>
<tr>
<td>800</td>
<td>2.077 1.876</td>
<td>12400</td>
<td>2.083 1.835</td>
</tr>
<tr>
<td>1600</td>
<td>2.092 1.882</td>
<td>13400</td>
<td>2.080 1.800</td>
</tr>
<tr>
<td>3200</td>
<td>2.038 1.761</td>
<td>14400</td>
<td>2.089 1.744</td>
</tr>
<tr>
<td>6400</td>
<td>2.081 1.755</td>
<td>15400</td>
<td>2.094 1.863</td>
</tr>
<tr>
<td>7400</td>
<td>2.065 1.802</td>
<td>16383</td>
<td>2.090 1.806</td>
</tr>
</tbody>
</table>
example, we used 500 trees to build a model). As before, leave-one-out CV was used to
calculate $\text{PMR}^{SS}$. The results are listed in Table 4. It can be seen that PPLS is by far the
best with smallest $\text{PMR}^{SS}$, and random forests also do better than PLS. It pinpoints the
importance of parameter shrinkage or model regularization with such a small sample size
and large number of variables. Furthermore, compared with a PLS model containing all the
genes, PPLS often leads to a much simpler model (with fewer genes), facilitating biological
interpretation of the results.

To further investigate the effects of variable selection, we also applied the forward selection
procedure to the LVAD data. The forward selection procedure starts with no variables in
a linear model (i.e., with intercept only). For each of the genes that are not included in
the current model, it calculates the p-value for the statistical significance of the gene being
added into the model. It picks up the smallest p-value and compares it to a parameter
called the significance level of entry $\alpha$. If it is larger than $\alpha$, then no covariate is selected
and the selection procedure stops; otherwise, the gene corresponding to the smallest p-value
is added into the model. This process is repeated until it stops. Again we use leave-one-
out CV to estimate $\text{PMR}^{SS}$, and consider $\alpha = 0.001$, 0.01 and 0.05. It can be seen from
Table 5 that starting with 200 or less genes with $\alpha = 0.05$ or 0.01, the forward selection
procedure performs even better than PPLS. However, its performance deteriorates quickly as
the number of genes considered increases. This unrobustness to the large number of genes in
microarray data is disturbing. Although it seems that only a few genes are needed in a model
for the LVAD data, which favors variable selection procedures such as forward selection, for
other microarray data requiring the use of many genes, the forward selection may not work
well because it cannot choose more than $n - 1$ genes in its final model. In contrast, PPLS is
more robust to the number of genes and virtually can be applied to data with any number
of genes. More evaluations using real data will be worthwhile.

A modified PPLS

In our PPLS method, we shrink the estimated regression coefficients $b_i$ towards zero. How-
ever, it may be argued that it is better to shrink the standardized coefficient estimates $\frac{b_i}{SE(b_i)}$
Table 5: Results of the forward selection by considering various numbers of top genes.

<table>
<thead>
<tr>
<th># of top genes</th>
<th>$PM\tilde{RSS}$</th>
<th># of top genes</th>
<th>$PM\tilde{RSS}$</th>
<th># of top genes</th>
<th>$PM\tilde{RSS}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\alpha = 0.05$</td>
<td>$\alpha = 0.01$</td>
<td>$\alpha = 0.001$</td>
<td>$\alpha = 0.05$</td>
<td>$\alpha = 0.01$</td>
</tr>
<tr>
<td>50</td>
<td>1.691</td>
<td>1.341</td>
<td>1.679</td>
<td>3200</td>
<td>2.001</td>
</tr>
<tr>
<td>100</td>
<td>0.941</td>
<td>1.261</td>
<td>1.620</td>
<td>6400</td>
<td>2.326</td>
</tr>
<tr>
<td>200</td>
<td>1.264</td>
<td>1.191</td>
<td>1.672</td>
<td>9400</td>
<td>2.138</td>
</tr>
<tr>
<td>400</td>
<td>2.235</td>
<td>2.107</td>
<td>1.881</td>
<td>12400</td>
<td>2.185</td>
</tr>
<tr>
<td>800</td>
<td>1.783</td>
<td>1.767</td>
<td>1.707</td>
<td>16383</td>
<td>2.213</td>
</tr>
<tr>
<td>1600</td>
<td>1.390</td>
<td>1.390</td>
<td>1.789</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6: Results with starting models including different numbers of top genes in a modified PPLS method.

<table>
<thead>
<tr>
<th># of top genes</th>
<th>$PM\tilde{RSS}$</th>
<th># of genes selected</th>
<th># of top genes</th>
<th>$PM\tilde{RSS}$</th>
<th># of genes selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>1.858</td>
<td>12</td>
<td>800</td>
<td>1.313</td>
<td>12</td>
</tr>
<tr>
<td>100</td>
<td>1.744</td>
<td>16</td>
<td>1600</td>
<td>1.326</td>
<td>13</td>
</tr>
<tr>
<td>200</td>
<td>2.341</td>
<td>15</td>
<td>3200</td>
<td>2.205</td>
<td>5</td>
</tr>
<tr>
<td>400</td>
<td>2.861</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
towards zero because in this way it takes account of possibly different variability of different regression coefficient estimates. In our approach, we have standardized gene expression change profiles before applying a weighted version of PLS to reduce the impact of the different variability of the parameter estimates, but it may not eliminate all the problems. A difficulty with this alternative approach is that we do not have a closed form solution for \( SE(b_i) \) in PLS. Here we propose using the jackknife method to estimate \( SE(b_i) \) (Efron, 1982). Due to the intensive computing involved, we only consider starting models with top 50, 100, 200, 400, 800, 1600, and 3200 genes. The new shrinkage rule is

\[
\hat{b}_i = SE(b_i) \text{sign}\left(\frac{b_i}{SE(b_i)}\right)(|\frac{b_i}{SE(b_i)}| - \Delta)_+ \quad i = 1, \ldots, p.
\]

Here \( SE(b_i) \) is estimated by the jackknife estimator (Efron, 1982):

\[
SE(b_i) = \sqrt{\frac{n' - 1}{n'} \sum_{j=1}^{n'} (\hat{b}_i^{(j)} - \hat{b}_i^{(j)})^2}, \quad \hat{b}_i^{(j)} = \frac{\sum_{j=1}^{n'} b_i^{(j)}}{n'},
\]

where \( n' \) is the number of samples in the estimation of the PLS coefficient \( b_i \) and \( b_i^{(j)} \) is the recomputed PLS coefficient with sample \( j \) deleted. Note that the jackknife procedure is conducted within the first level CV of calculating \( P\hat{M}RSS \), leading to two nested levels of CV. The results are somewhat wild (Table 6). Compared with Table 1, we see that the new method has some, though only minimum, improvement over the original PPLS when their \( P\hat{M}RSS \)'s are compared, but for many other cases the new method works much worse. This may indicate that no further normalization of the estimated regression coefficients using their SE's is needed, or that a better SE estimator than the jackknife is needed when using SE's to normalize.

**Discussion**

DNA microarray technology enables one to examine the expression levels of thousands of genes simultaneously. Meanwhile, it also creates many challenges in analyzing the resulting data. For example, it is no longer possible to fit a multiple linear regression model using OLS to correlate gene expression with a response variable (e.g., LVAD support time), due to the fact that the number of genes greatly exceeds the number of samples. Simple gene
selection methods based on univariate analysis are in general not optimal. Our proposed PPLS method is a contribution to this problem: The PLS method is first used to fit a linear model with many genes, then data-adapted shrinkage on the estimated regression coefficients automatically accomplishes gene selection and model regularization. In our analysis, starting from a model with 100 genes, PPLS selected a final model containing only 21 genes. It established a relationship between the LVAD support time and gene expression changes. Although the permutation test showed that the selected final model is not statistically significant, possibly due to the small sample size, the results show that the PPLS method is pretty stable based on the overlapping among the genes selected by different starting models. The results comparing PPLS to PLS and random forest also show that PPLS works much better than the other two methods. Therefore, these 21 genes selected may be the candidate genes for future research.

Finally we note that PLS is a very useful method. It has been applied in analyzing gene expression data in other contexts. Nguyen and Rocke (2002a, 2002b) and Hawkins et al (2003) used PLS to do dimension reduction before applying other discriminant analysis methods for cancer classification. Nguyen and Rocke (2002c) and Park et al. (2002) applied PLS to model survival times. Johansson et al. (2003) modeled gene expression cell cycles using PLS. Our recent work (Huang and Pan 2003) demonstrates that PPLS can improve over PLS in tumor classification; this is in line with the work of Tibshirani et al. (2002) showing that incorporating parameter shrinkage in a diagonalized linear discriminant method is productive. This is not surprising in light of the features of gene expression data with a small number of samples but a large number of genes. An attractive point of PPLS is its unified framework enabling its application to classification, regression, and possibly other types of gene expression data.

**Acknowledgements**

The authors are grateful to the reviewers for helpful comments. XH and WP were supported by an NIH grant and a Minnesota Medical Foundation grant. JH was supported by an AHA grant, the Lillehei Heart Institute and the Minnesota Medical Foundation.
References


Appendix

1. Partial least squares

We review PLS briefly. Suppose that we want to model a response variable $Y$ using some explanatory variables $X_i$'s. The central idea of PLS is to construct new explanatory variables (also called components) $T_1, \ldots, T_q$, each of which is a linear combination of the variables $X_i$ with $q \leq p$; then use $T_k$'s to model $Y$. An algorithm can be described as the following (Garthwaite, 1994). First, we let $Y^1$ and $X_1^1, \ldots, X_p^1$ represent the centered value of $Y$ and $X_1, \ldots, X_p$ respectively. We construct the PLS component

$$T_1 = \sum_{i=1}^{p} w_i^1 b_i^1 X_i^1,$$

where $b_i^1$ is the OLS estimator obtained by regressing $Y^1$ on $X_i^1$. We set the weight $w_i^1 = \frac{1}{\sqrt{\text{Var}(b_i^1)}}/\sum \frac{1}{\sqrt{\text{Var}(b_i^k)}}$ (see appendix for the estimation of $\text{Var}(b_i^k)$, $k = 1, \ldots, q$). Note that $\sum w_i^1 = 1$.

This procedure extends iteratively to give the PLS components $T_2, \ldots, T_q$. Briefly, in the $k_{th}$ step, let $Y^k$ be the residual of $Y^{k-1}$ regressed on $T_{k-1}$ and let $X_i^k$ be the residual of $X_i^{k-1}$ regressed on $T_{k-1}$, $k = 2, \ldots, q$. The $k_{th}$ PLS component is constructed to be

$$T_k = \sum_{i=1}^{p} w_i^k b_i^k X_i^k,$$

where $b_i^k$ is the OLS estimator of regressing $Y^k$ on $X_i^k$, and $w_i^k = \frac{1}{\sqrt{\text{Var}(b_i^k)}}/\sum \frac{1}{\sqrt{\text{Var}(b_i^k)}}$ is inversely proportional to the variance of the $b_i^k$'s. It is trivial that the correlation between any pair of components is 0 (Helland, 1988; Garthwaite, 1994).

After all the components $T_1, \ldots, T_q$ are determined, they are related to $Y$ using a linear regression model

$$Y = \gamma_0 + \gamma_1 T_1 + \gamma_2 T_2 + \cdots + \gamma_q T_q$$

with the regression coefficients $\gamma_0, \gamma_1, \ldots, \gamma_q$ determined by OLS. Since the components $T_1, \ldots, T_q$ are linear combinations of the variables $X_i$ (Garthwaite, 1994), we can rewrite
model (1) as

$$ Y = b_0 + \sum_{i=1}^{p} b_i X_i $$

which established a linear relationship between $Y$ and the original covariates $X_1, \ldots, X_p$.

2. **Derivation of the estimation of $Var(b^k_i)$, $k = 1, \ldots, q$.**

Since $Y^k$ is centered, our OLS model only contains a slope $b^k_i$

$$ Y^k = b^k_i X^k_i, $$

where because $X^k_i$ is also centered

$$ b^k_i = \frac{\sum_{j=1}^{n} x^k_{ij} Y^k_j}{\sum_{j=1}^{n} x^k_{ij}^2} $$

The variance of $b^k_i$ is

$$ Var(b^k_i) = Var\left(\frac{\sum_{j=1}^{n} x^k_{ij} Y^k_j}{\sum_{j=1}^{n} x^k_{ij}^2}\right) $$

$$ = \frac{1}{(\sum_{j=1}^{n} x^k_{ij})^2} Var\left(\sum_{j=1}^{n} x_{ij} Y^k_j\right) $$

$$ = \frac{1}{(\sum_{j=1}^{n} x^k_{ij})^2} \left(\sum_{j=1}^{n} Var(x_{ij} Y^k_j)\right) $$

$$ = \frac{1}{(\sum_{j=1}^{n} x^k_{ij})^2} \left(\sum_{j=1}^{n} x_{ij}^2 \sigma^k_i \right) $$

$$ = \frac{\sigma^k_i}{\sum_{j=1}^{n} x^k_{ij}^2} $$

where $\sigma^2_i$ is estimated by

$$ \hat{\sigma}^k_i = \frac{RSS^k}{n-2} $$

$$ = \sum_{j=1}^{n} (Y^k_j - b^k_i X^k_j)^2 $$

22