Projection Pursuit method for the Small Sample Size with the Large Number of Variables

Eun-kyung Lee\textsuperscript{1} and Dianne Cook\textsuperscript{2}

\textsuperscript{1} Department of Statistics, Ewha Womans University, 11-1 Daehyun-dong, Seodaemun-gu, Seoul, 120-750, Korea
\textsuperscript{2} Department of Statistics, Iowa State University, Ames, IA 50011, USA

Summary

In high-dimensional data, one often seeks a few interesting low-dimensional projections which reveal important aspects of the data. Projection pursuit for exploratory supervised classification is for finding separable class structure. Even though the projection pursuit method can bypass the curse of dimensionality, when we have the small number of observations relative to the number of variables, the class structure of optimal projection can be biased seriously. In this situation, most classical multivariate analysis methods have problems. We discuss how the sample size and dimensionality are related, and we propose a new projection pursuit index that considers the penalty for the projection coefficients and overcomes the problem of small sample size.

Keywords: The curse of dimensionality; Gene expression data analysis; Multivariate data; Penalized discriminant analysis; Projection pursuit
1 Introduction

This paper is about the exploratory data analysis of small samples with a large number of variables, especially in supervised classification. If the classifier obtained for a given training set is inadequate, it is natural to consider adding new variables, particularly ones that will help separate the difficult to classify cases. If the new variables provide any additional information, the performance of the classifier must improve. Unfortunately, beyond a certain point, additional variables will make the classifier worse. The problem arises when the sample size is small or the variables are highly correlated. When the training set is relatively small compared to the number of variables, the statistical parameters estimated on this training set are not accurate and they are unstable. A quite different classifier may be obtained when a different training set is used.

A small sized sample with the very large number of variables is the typical situation of gene expression data analysis. In this paper, we focus on leukemia data from two types of leukemia, acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). This data set consists of 25 cases of AML and 47 cases of ALL (38 cases of B-cell ALL and 9 cases of T-cell ALL). After preprocessing, we have 3571 human genes (Golub et al. 1999).

In Chapter 2, we discuss how the sample size and dimensionality are related and how they affect the supervised classification. Chapter 3 introduces a new projection pursuit method when the sample size is small and the number of variables are large and describes its properties and apply our new projection index to leukemia data. We explain how this new projection pursuit index can be applied to the gene selection method and compare to other gene selection methods in Chapter 4.

2 Problems of high dimensionality

To see how the number of variables affects the classification methods that use separating hyperplanes, we investigate linear discriminant analysis with leukemia data ($n=72$ and $p=3571$). For gene selection, we use the ratio of between-group to within-group sums of squares.

$$BW(j) = \frac{\sum_{i=1}^{n} \sum_{k=1}^{g} I(y_i = k)(\bar{x}_{k,j} - \bar{x}_{..,j})^2}{\sum_{i=1}^{n} \sum_{k=1}^{g} I(y_i = k)(x_{i,j} - \bar{x}_{k,j})^2}$$

(1)

where $\bar{x}_{..,j} = (1/n) \sum_{i=1}^{n} x_{i,j}$ and $\bar{x}_{k,j} = (\sum_{i=1}^{n} I(y_i = k)x_{i,j}) / (\sum_{i=1}^{n} I(y_i = k))$. First, we sample a 2/3 training set ($n_{train} = 48$) and calculate BW values for each gene using this training set. After then we select the $p$ variables that have larger BW values. Using this training set with $p$ variables, we
build the classifier using Linear Discriminant Analysis (LDA) and compute the training error and the test error. Repeat this 200 times. Median and upper quantiles of training and test errors are summarized in Table 1. For various $p$, training errors are almost 0. That is, the training sets are perfectly separated regardless of $p$. But test errors are increased as $p$ is increased. As $p$ approaches $n$, the test error gets worse.

<table>
<thead>
<tr>
<th></th>
<th>True Class</th>
<th></th>
<th>Permut ed Class</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Training error</td>
<td>Test error</td>
<td>Training error</td>
<td>Test error</td>
</tr>
<tr>
<td>$p$</td>
<td>$Q_2$</td>
<td>$Q_3$</td>
<td>$Q_2$</td>
<td>$Q_3$</td>
</tr>
<tr>
<td>40</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

$Q_2$ : median $Q_3$ : upper quantile

It gets more interesting when we scramble the class id’s using permutation and then the class separations are spurious. Table 1 shows the results of the same procedure outlined above using permuted classes. We might suspect that a classifier will not accurately separate these spurious classes. Surprise! When $p = 40$, the training error is 0. This result can be explained by the capacity of a separating plane. When the training set has $p = 40$ and $n = 48$, the probability that $n$ sample points in $p$-dimensional space are linearly separable is close to 1 (Ripley, 1996). Therefore there exists a separating hyperplane purely by chance. When $p$ gets smaller, the training error with the permuted class gets larger. The test errors are consistent, independent of $p$.

These results can be seen visually using the LDA projection pursuit index (Lee, 2003). Figure 2 shows the 2-dimensional optimal projections using the LDA PP index with the training set for both true class (top row) and the permuted class (bottom row). 1, 2, and 3 represent AML, B-cell ALL and T-cell ALL in the training set, and the symbols $\bigcirc$, $\bigtriangleup$, and $\ast$ represent AML, B-cell ALL and T-cell ALL in the test set. After finding the 2-dimensional optimal projection with the training set, we project both training and test sets onto this optimal projection. For the true class, the training set with $p = 40$ is more separable and has smaller within-class variance than $p = 10$, but the test set shows quite different group means and more larger within-class variance. Notice that the test set is not separable on this projection. When $p$ is smaller, the training set has larger within-class variance and the test set has more similar structure to the training set.

For the permuted class, when $p = 40$, we can find separated class structure with permuted class for the training set. When $p = 30$, the training set is
still separated. As $p$ is smaller, class structure for the training set weakens. For all $p$, the test sets don’t reveal any class structure. These results support LDA errors in Table 1. From these results, we can conclude that when $p$ is large, the LDA classifier is biased too much. Therefore we need to choose the number of variables carefully.

Many classical multivariate analysis methods need to calculate the inverse of covariance matrix. If $n \geq p + 1$ or the variables are highly correlated, $\Sigma$ will be close to singular which will result in numerical instability in calculating the inverse. It is necessary to estimate for the variance-covariance matrix differently. If there is prior information about this covariance, then we can use a Bayesian or pseudo-Bayesian estimate $\hat{\Sigma} = (1 - \lambda)\Sigma + \lambda\Omega$, where $\Omega$ is a pre-determined matrix from prior information or assumption. If $\Omega$ is diagonal, it will help avoid numerical problems. For the extreme assumption that all variables are independent, we can use $\hat{\Sigma} = \text{diag}(\Sigma)$. Even though the assumption is incorrect, the resulting heuristic estimates can provide better performance than the MLE.

We investigate LDA in this point of view. LDA finds a projection $a$ by maximizing $a^T \Sigma_B a / a^T \Sigma_W a$, where $\Sigma_B$ is the between-class covariance matrix and $\Sigma_W$ is the within-class covariance matrix. When sample size is small and the number of variables are large, LDA is usually too flexible and sometimes
3 PDA projection pursuit index

3.1 Index definition

We propose a new projection pursuit index which is the extension of the LDA PP index (Lee, 2003). The main purpose is to (1) prevent the problems with the small number of observations and the large number of variables and (2)
find projections that contain class separations in a reasonable manner. We use \( \hat{\Sigma}(\lambda) = (1 - \lambda) \hat{\Sigma} + \lambda \cdot diag(\hat{\Sigma}) \) as our estimate of variance-covariance matrix. As \( \lambda \) is increased, \( \hat{\Sigma} \) tends to be \( diag(\hat{\Sigma}) \). When the data is standardized, it reduces to \( \hat{\Sigma}(\lambda) = (1 - \lambda) \hat{\Sigma} + \lambda I \).

Let \( X_{ij} \) be the \( p \)-dimensional vector of the \( j \)th observation in the \( i \)th class, \( i = 1, \ldots, g \), \( j = 1, \ldots, n_i \), \( g \) is the number of classes, \( n_i \) is the number of observations in class \( i \), and \( n = \sum_{i=1}^{g} n_i \). Let \( \bar{X}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} X_{ij} \) be the \( i \)th class mean and \( \bar{X}_i = \frac{1}{n} \sum_{i=1}^{g} \sum_{j=1}^{n_i} X_{ij} \) be the total mean. For convenience, we assume that \( X_{ij} \)'s are standardized. Let

\[
B = \sum_{i=1}^{g} n_i (\bar{X}_i - \bar{X}_.) (\bar{X}_i - \bar{X}_.)^T : \text{between-class sums of squares},
\]

\[
W = \sum_{i=1}^{g} \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_i) (X_{ij} - \bar{X}_i)^T : \text{within-class sums of squares}.
\]

Here, \( B + W = n \hat{\Sigma} \) and \( \hat{\Sigma} \) is the correlation matrix. Then, the PDA projection pursuit index is

\[
I_{PDA}(\A, \lambda) = 1 - \frac{|A^T [(1 - \lambda) W + n \lambda I_p] A|}{|A^T [(1 - \lambda)(B + W) + n \lambda I_p] A|} \quad (2)
\]

where \( A \) is an orthonormal projections onto \( k \)-dimensional space and \( \lambda \in [0, 1] \) is a predetermined parameter. Let \( B^* = (1 - \lambda) B \) and \( W^* = (1 - \lambda) W + \lambda n I_p \). Then, the PDA index has a same form as the LDA index and when \( \lambda = 0 \), the PDA index is same as the LDA index.

**Proposition 1.** Let \( \Phi^* = (1 - \lambda)(B + W) + \lambda n I_p = B^* + W^* \). Then,

\[
0 \leq 1 - \prod_{i=1}^{k} \lambda_i^* \leq I_{PDA}(\A, \lambda) \leq 1 - \prod_{i=p}^{k+1} \lambda_i^* \leq 1 \quad (3)
\]

where \( \lambda_i^* \geq \lambda_2^* \geq \cdots \geq \lambda_p^* \geq 0 \) : eigenvalues of \( \Phi^* \cdot \frac{1}{\sqrt{2}} W^* \Phi^* \cdot \frac{1}{\sqrt{2}} 
\),

\( e^*_{1,1}, e^*_{1,2}, \cdots, e^*_{p} \) : corresponding eigenvectors of \( \Phi^* \cdot \frac{1}{\sqrt{2}} W^* \Phi^* \cdot \frac{1}{\sqrt{2}} 
\),

\( f^*_{1,1}, f^*_{2,2}, \cdots, f^*_{p} \) : eigenvectors of \( \Phi^* \cdot \frac{1}{\sqrt{2}} B^* \Phi^* \cdot \frac{1}{\sqrt{2}} 
\).

In (3), the right equality holds when \( \A = \Phi^* \cdot \frac{1}{\sqrt{2}} [e_{\cdot 1}^{*} \ e_{\cdot 2}^{*} \ \cdots \ e_{\cdot p}^{*}] \) and the left equality holds when \( \A = \Phi^* \cdot \frac{1}{\sqrt{2}} [f_{\cdot 1}^{*} \ f_{\cdot 2}^{*} \ \cdots \ f_{\cdot k}^{*}] \) and the left equality holds when \( \A = \Phi^* \cdot \frac{1}{\sqrt{2}} [e_{\cdot k}^{*} \ e_{\cdot k+1}^{*} \ \cdots \ e_{\cdot 1}^{*}] = \Phi^* \cdot \frac{1}{\sqrt{2}} [f_{\cdot 1}^{*} \ f_{\cdot 2}^{*} \ f_{\cdot k+2}^{*} \ \cdots \ f_{\cdot p}^{*}] \).

Proof of this proposition is same as Proposition 1 in Lee, et al (2004). To explain the difference between the PDA and LDA index, we use the principal
components. Let

$$\Phi = B + W = QDQ^T = \sum_{i=1}^{p} d_i q_i q_i^T,$$ \quad (4)

where \( Q = [q_1, q_2, \ldots, q_p] \) is the eigenvector matrix of \( \Phi \), \( D = \text{diag}(d_1, d_2, \ldots, d_p) \) is the eigenvalues of \( \Phi \), \( d_i = 0 \) for all \( i = r + 1, \ldots, p \) and \( \text{rank}(\Phi) = r. \) Then, \( tr(\Phi) = tr(\Phi^*) = np, \) and

$$\Phi^* = \sum_{i=1}^{p} [(1 - \lambda) d_i + \lambda n] q_i q_i^T \quad (5)$$

Therefore, \( \Phi \) and \( \Phi^* \) have same principal component directions and total variance. The difference between these two variance matrices is the proportion of total variance due to the \( k \)th principal component. For the LDA index, we use the original principal component of \( \Phi \). The PDA index keeps the direction of \( \Phi \)'s principal component and the total variance, but changes the proportion of total variance explained by each direction. When the proportion due to the \( k \)th principal component is larger than \( 1/p \), the PDA index uses the shrunk proportion of the total variance due to this direction. Otherwise, the PDA index uses the increased proportion of the total variance due to this direction. For the non-significant principal component, the PDA index put \( \lambda/p \) as a proportion of the total variance on that principal component. Therefore if \( p \) is large and if we want to keep the amount of shrinkage, we need to use larger \( \lambda \).

### 3.2 Examples

A toy example from Marron, et al (2002) is used to demonstrate the difference between the PDA index and the LDA index. There are two classes in the 39-dimensional space. Each class has 20 data points. They are generated from the standard Gaussian distribution, except that the mean of the first variable is shifted to 2.2 and -2.2 for two classes, respectively. Therefore, the separation of two classes only depends on the first variable. Figure 3 (a)-(e) show the histograms of the 1-dimensional optimal projected data using the PDA PP index with various \( \lambda \) values. As we mentioned before, when \( \lambda = 0 \) (Figure 3-b), the PDA PP index is same as the LDA index and the projected data has very small within-group variance. As \( \lambda \) is increased, the within-group variances of the projected data also get larger and the projected data have more reasonable class structure.

To see the difference between the LDA index and the PDA index in detail, we compared the optimal projection coefficients of the LDA index and the PDA index with \( \lambda = 0.9 \). Figure 3 (b-1) shows the absolute values of the
optimal projection coefficients using the LDA index. All the coefficients have small values and from these coefficients, it is hard to decide which variables are more important than the others to separate two classes. On the other hand, the coefficient of the first variable has very large value and the others are very small (Figure 3 (e-1)). From this result, we can conclude that the LDA index focuses only on the projection having small within-class variance relative to the total variance and leads us to the projection that is biased too much, therefore cannot be useful when the sample size is small and the number of variables are large. On the other hand, the PDA index can lead us to a quite reasonable projection and its coefficients can be used as a guideline to select important variables.

We apply the PDA index to the Leukemia data with three classes: AML, B-cell ALL and T-cell ALL. Figure 4 shows the results using the PDA index with various $\lambda$ values. After finding the 2-dimensional optimal projection using the PDA index on the training set, we project both training and test sets onto this optimal projection.

When $\lambda = 0$, the training set and the test set have different class structure on this projection. The training set has very small within-class variances. On the other hand, the test set has large within-class variance. When $\lambda = 0.1$, the training set and the test set have similar class structures. But as $\lambda$ is increased, the within-class variance of the training set is increased too much and beyond a certain point, the PDA index can be biased in the other way of the LDA index, that is, the within-class variance of the training set has larger than the within-class variance of the test set. Therefore we need to select $\lambda$ value that can keep the within-class variance of the training set in the reasonable amount.

For selecting $\lambda$ value, we suggest to use $S(\lambda) = tr(W)/n$, where $W$ is the within-class sum of squares of the optimal projected data using the PDA index with $\lambda$. If we use standardized data, the optimal value of $S(\lambda)$ is 1. If $S(\lambda)$ is smaller than 1, we suggest to increase your $\lambda$ value. If $S(\lambda)$ is larger than 1, we suggest to decrease your $\lambda$ value. Figure 5 shows the plot of $S(\lambda)$ and $\lambda$. For the leukemia data with 40 genes, the best $\lambda$ value is around 0.2. In the plot of 2-dimensional projected data using the PDA index with $\lambda = 0.2$, we can see that training set and test set have similar within-group variances.

We examine the training and test error in classification using the LDA and the PDA projection pursuit methods. After finding the optimal 2-dimensional projection using the LDA and PDA index with $\lambda = 0.1$, we build a classifier using the rule (11-65) in Johnson and Wichern (2002), and compute the training and test errors. This is repeated 200 times. The median and upper quantile of the test errors are summarized in Table 2. The results from the LDA PP method is same as Table 1. For the PDA PP method, the training
Figure 3. Toy example: \( p = 39 \), \( n = 40 \). (a) - (e) The histograms of 1D optimal projected data using the PDA index with various \( \lambda \), (b-1) (e-1) the projection pursuit coefficient values for corresponding variables.
Figure 4. Leukemia data ($p=40$) - 2D projections using PDA index with various $\lambda$

Figure 5. $\lambda$ selection for the Leukemia data ($p=40$) - Plot of $S(\lambda)$ vs. $\lambda$ and the optimal 2-dimensional projected data using the PDA index with $\lambda=0.2$

Figure 6. Leukemia data ($p=3571$) - 2-dimensional projections using PDA index with various $\lambda$
errors are the same as linear discriminant classification with the original data (all 0), but the test errors are much lower. For various numbers of variables (p), the test errors of the PDA PP method are smaller than the errors of the LDA PP method. Also the PDA PP method shows very consistent test errors for all p. From this result, we can conclude that the PDA index helps find less biased and more reasonable projections.

Table 2. Training and test error for the LDA PP method and the PDA PP method with \( \lambda = 0.2 \).

<table>
<thead>
<tr>
<th></th>
<th>LDA PP</th>
<th>PDA PP with ( \lambda = 0.1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Training error</td>
<td>Test error</td>
</tr>
<tr>
<td>p</td>
<td>( Q_2 )</td>
<td>( Q_3 )</td>
</tr>
<tr>
<td>40</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 6 shows the 2-dimensional optimal projections using the PDA index with Leukemia data with all genes. For all \( \lambda \) values, the between-class variance structure are similar, the means of three classes form a triangle shape, but the within-class structure are quite different. When \( \lambda = 0 \), the within-class sums of squares of the projected data is a singular matrix, which would suggest the 1-dimensional projection is enough to use linear discriminant analysis and this optimal projection will not be useful for showing separations in new samples. When \( \lambda = 0.1 \), the within-variance is very small, but it is nonsingular and the 2-dimensional projection has more information than the 1-dimensional projection. As \( \lambda \) is increased, the within-class sums of square of the projected data gets larger and the training and test set have more similar within-class variances. Using \( S(\lambda) \), we suggest to use larger \( \lambda \) value, around 0.7 (Figure 7).

4 Application : Gene selection

In the previous sections, we used the BW values to select genes that are useful for separating classes. The BW values are calculated for each gene and there is no consideration of the correlation between genes in their calculation. But most of genes are highly correlated and some genes work together to separate classes, even though they have small BW values. In this sense, the projection coefficients from the PDA index can provide a better gene selection method. As we saw in the toy example, these coefficients from the PDA index tend to be more reasonable than the coefficients from the LDA index. These
coefficients can be used to explain how important the corresponding variables are to separate classes. We compare these coefficients to the BW values in terms of separating classes.

To see how the projection coefficients from the PDA index and the BW values are related, we start from the Leukemia data \((p = 3571, n = 72)\) with 2 classes, AML and ALL. Figure 8 shows the plot of the BW and the projection coefficients from the PDA projection method with \(\lambda = 0.9\). For most genes, the BW values are less than 0.5 and the projection coefficients are in between -0.04 and 0.04. Most genes with large BW values have larger than 0.04 or smaller than -0.04 for the projection coefficient value. On the other hand, the BW values of most genes with high projection coefficients are spread out very widely. Some of them are less than 0.5.

\textit{Table 3. Comparison between BW and projection coefficients.}

<table>
<thead>
<tr>
<th>From large projection coeff.</th>
<th>From large BW</th>
</tr>
</thead>
<tbody>
<tr>
<td>\textbf{genes}</td>
<td>\textbf{BW}</td>
</tr>
<tr>
<td>U34877_at</td>
<td>0.16</td>
</tr>
<tr>
<td>M27891_at</td>
<td>2.66</td>
</tr>
<tr>
<td>M84526_at</td>
<td>3.01</td>
</tr>
<tr>
<td>X95735_at</td>
<td>1.80</td>
</tr>
<tr>
<td>HG1612-HT1612_at</td>
<td>1.03</td>
</tr>
</tbody>
</table>

We select 5 genes from large BW values: M84526\_at, M27891\_at, U46499\_at,
Figure 8. BW vs PDA(λ = 0.9) projection coefficients: Leukemia data with two classes, AML and ALL (p=5571, n=72)

M23197 at and X95735 at, and large projection coefficients: U34877 at, M27891 at, M84526 at, X95735 at and HG1612-HT1612 at, respectively and compare them. Table 3 shows the BW values and projection coefficients for each gene. All genes from larger projection coefficients have large BW values except one, gene U34877 at. All genes from larger BW values have large projection coefficients. Figure 8 shows the scatter plot matrices of 5 genes from large BW and projection coefficients. 5 genes from large BW values show quite separable group means, but any pairs of these genes are not clearly separable. At least one or two cases are misclassified if we use one separating hyperplane. On the other hand, 5 genes from larger projection coefficients show more mixed structure. In the plot of X95735 at and HG1612-HT1612 at, we can find a separating hyperplane that 2 classes are clearly separable.

Whenever we optimize the PDA index, we can get different projection coefficients, especially when we use very large number of variables. It is from the curse of dimensionality. Because most of high dimensional space is empty, especially when we have small number of observations, we can separate classes in many different ways. Therefore with the projection pursuit method, we can explore various projections that provide the separated class structure.

To show how this works, we choose another projection from maximizing the
Figure 8. Scatter plot matrices: 5 genes from large BW values and 5 genes from large projection coefficients with the PDA index ($\lambda = 0.9$) : Leukemia data with two classes, AML(○) and ALL(+) ($p=3571, n=72$)
PDA index (λ = 0.9) and select 10 genes with large projection coefficients. Table 4 shows the summary of 10 genes from larger projection coefficients. All 10 genes from larger projection coefficients have small BW values (less than 1) and some of them have very small BW values, even less than 0.1.

Table 4. Comparison between BW and projection coefficients.

<table>
<thead>
<tr>
<th>genes</th>
<th>BW</th>
<th>PP coeff.</th>
</tr>
</thead>
<tbody>
<tr>
<td>M82809_at</td>
<td>0.1241</td>
<td>-0.0673</td>
</tr>
<tr>
<td>X51521_at</td>
<td>0.5920</td>
<td>0.0563</td>
</tr>
<tr>
<td>S68616_at</td>
<td>0.0932</td>
<td>-0.0555</td>
</tr>
<tr>
<td>L02426_at</td>
<td>0.0005</td>
<td>-0.0554</td>
</tr>
<tr>
<td>AF006087_at</td>
<td>0.0187</td>
<td>0.0543</td>
</tr>
<tr>
<td>U30255_at</td>
<td>0.3871</td>
<td>-0.0541</td>
</tr>
<tr>
<td>U41654_at</td>
<td>0.1560</td>
<td>0.0525</td>
</tr>
<tr>
<td>M84371_at</td>
<td>0.7359</td>
<td>0.0522</td>
</tr>
<tr>
<td>L10373_at</td>
<td>0.2444</td>
<td>0.0519</td>
</tr>
<tr>
<td>U10868_at</td>
<td>0.4401</td>
<td>-0.0518</td>
</tr>
</tbody>
</table>

Figure 10 shows the histograms of the projected data onto the LDA optimal projection with selected genes from larger BW values and larger projection coefficients in Table 4. In Figure 10(a-1), 5 genes from larger BW values can separate AML and ALL clearly except 5 cases (2 cases in ALL and 3 cases in AML). As we increase the number of genes up to 10, the two groups are more separable, but with 10 genes from larger BW values, we still have a misclassified case.

Figure 10(b-1) is the histogram of the projected data onto the LDA optimal projection with 5 selected genes from large projection coefficients. Lots of cases are misclassified. However, as we increase the number of genes up to 10, the performance of separating two classes are improved very quickly. When we use 10 genes from larger projection coefficients, two classes are clearly separable.

As we can see in Figure 10, adding more genes to classification is not much helpful to separate classes beyond a certain point. As a guideline to decide the number of separated genes, we recommend to use the LDA index value. Figure 11 shows plots of the optimal LDA index value versus the number of genes. In Figure 11(a), we use the same projection coefficient as in Table 3 and optimize these selected genes with the 1-dimensional LDA index. After p = 5, the LDA index value isn’t much increased. Therefore, 5 selected genes in Table 3 are enough to separate AML and ALL. In Figure 11(b), genes are selected from larger BW values. After one or two genes are selected, LDA index value is increased very slowly. Figure 11(c) is the LDA index plot of the selected genes from larger projection coefficients that are used for Table 4. As we expected from very low BW values, the LDA index values are very
low when $p$ is small. But as $p$ is increased, the LDA index value is increased rapidly and after $p = 12$, it stays steady.

5 Discussion

We have looked at the problems in a high dimensional space, especially when we have a small number of observations, and have proposed a new projection pursuit index that adds a penalty term for high dimensionality or multicollinearity.

The PDA index works well to separate classes in reasonable manner when data have multicollinearity or very high dimensionality relative to the sample size. To use the PDA index, we need to choose $\lambda$. In the original PDA, cross-validation can be used to select $\lambda$. But the main purpose of projection pursuit is exploratory data analysis. Therefore cross-validation cannot be a good approach to select $\lambda$ for our PDA index. This is the main reason we keep $\lambda$ in $[0,1)$. One guideline to select $\lambda$ is to use larger $\lambda$ for large $p$.

The PDA index can be used to select important variables that are helpful to separate classes. In gene expression data analysis, this application is useful to select important genes that work differently in each class. It can be extended
to cluster genes.

To optimize this PDA index, we used the modified simulated annealing method (Lee, 2003). We have used the R language for this research and the PDA index is included in the classPP package (available at CRAN).

References


Statistical Learning, Springer.


