Projection Pursuit for Exploratory Supervised Classification

EUN-KYUNG LEE¹, DIANNE COOK², SIGBERT KLINKE³, and THOMAS LUMLEY⁴

1 Department of Statistics, Ewha Womans University
2 Department of Statistics, Iowa State University
3 Institute for Statistics and Econometrics, Humboldt-Universität zu Berlin
4 Department of Biostatistics, University of Washington

ABSTRACT In high-dimensional data, one often seeks a few interesting low-dimensional projections which reveal important features of the data. Projection pursuit is a procedure for searching high-dimensional data for interesting low-dimensional projections via the optimization of a criterion function called the projection pursuit index. Very few projection pursuit indices incorporate class or group information in the calculation. Hence, they cannot be adequately applied in supervised classification problems to provide low-dimensional projections revealing class differences in the data. We introduce new indices derived from linear discriminant analysis that can be used for exploratory supervised classification.

Key Words: Data mining; Exploratory multivariate data analysis; Gene expression data; Discriminant analysis;

1. Introduction

This paper is about methods for finding interesting projections of multivariate data when the observations belong to one of several known groups. The type of data is denoted as a p-dimensional vector $X_{ij}$ representing the $j$th observation of the $i$th class, $i = 1, \ldots, g$, $g$ is the number of classes, $j = 1, \ldots, n_i$, and $n_i$ is the number
of observations in class \(i\). Let \(\bar{X}_i = \sum_{j=1}^{n_i} X_{ij} / n_i\) be the \(i\)th group mean and \(\bar{X} = \sum_{i=1}^{g} \sum_{j=1}^{n_i} X_{ij} / n\) be the total mean, where \(n = \sum_{i=1}^{g} n_i\). Interesting projections correspond to views where there are the biggest
difference between the observations from different classes, that is, the classes are clustered in the view. In
this paper, the approach to finding interesting projections uses the measures of between group variation,
relative to within-group variation. These new methods are important for exploratory data analysis and data
mining purposes when the task is to (1) examine the nature of clustering in the space of the data due to
class information, and (2) to build a classifier for predicting the class of new data.

Projection pursuit is a method to search for interesting linear projections by optimizing some pre-
determined criterion function, called a projection pursuit index. This idea originated with Kruskal (1969),
and Friedman and Tukey (1974) first used the term “projection pursuit” describing technique for exploratory
analysis of multivariate data. It is useful for an initial data analysis, especially when data is in high dimen-
sional space. A problem many multivariate analysis techniques face is “the curse of dimensionality”, that is,
most of high dimensional space is empty. Projection pursuit methods help for exploring multivariate data
in interesting low dimensional spaces. The definition of “interesting” projection depends on the projection
pursuit index.

Many projection pursuit indices have been developed to define interesting projections. Because most
low-dimensional projections are approximately normal (Huber, 1985), most of projection pursuit indices are
focused on non-normality: for example, the entropy index and the moment index (Jones and Sibson, 1987),
the Legendre index (Friedman, 1987), the Hermite index (Hall, 1989), and the Natural Hermite index (Cook,
et al, 1993).

Visual inspection of high dimensional data using projections is helpful to understand data, especially
when it is combined with dynamic graphics. GGobi is an interactive and dynamic software system for data
visualization and projection pursuit is implemented in it dynamically (Swayne, et al, 2003). The Holes index
and the central mass index in GGobi are helpful in finding projections with few observations in the center
and projections containing an abundance of points in the center, respectively (Cook, et al, 1993).

As the data mining area has grown, projection pursuit methods are used in classification and clustering to escape the curse of dimensionality. Posse (1992) suggested the projection pursuit discriminant analysis for two groups. He used the kernel estimation of the projected data instead of the original data and used the total probability of misclassification of the projected data as a projection pursuit index. Polzehl (1995) considered the cost of misclassification and used the expected overall loss as a projection pursuit index. Flick, et al (1990) proposed the projection pursuit method for classification. They used a basis function expansion and minimized a measure of scatter to find the best projection.

However, most methods in supervised classification focus on finding accurate classifiers. Projection pursuit methods for classification focus on 1-dimensional projections provides ways for visual inspection of high-dimensional data. It is hard to extend these to k-dimensional projections.

The methods presented in this paper start from a well-known classification method, linear discriminant analysis (LDA). This approach is extended to provide new projection pursuit indices for exploratory supervised classification. This indices use Fisher’s linear discriminant ideas and expand on Huber’s ideas on projection pursuit for classification. These new indices are helpful for building understanding about how class structure relates to measured variables and they can be used to provide graphics to assess and verify supervised classification results. This indices are implemented as an R package, and for dynamic graphics, these indices are available in GGobi.

Chapter 2 introduces the new projection pursuit indices and describes their properties. The optimization method is discussed in Chapter 3. Chapter 4 describes how to apply these indices using two gene expression data sets.

2. Index Definition

2.1 LDA projection pursuit index
The first index is derived from classical linear discriminant analysis (LDA). The approach, first developed by Fisher (1938), finds linear combinations of the data which have large between-group sums of squares relative to within-group sums of squares. (For detailed explanations, see Johnson and Wichern, 1992.) Let

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

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

Dimension reduction is achieved by finding the linear projection, \( \mathbf{a} \), that maximizes \( \mathbf{a}^T \mathbf{B} \mathbf{a} / \mathbf{a}^T \mathbf{W} \mathbf{a} \), which approach leads to the natural definition of a projection pursuit index. \( \mathbf{a}^T \mathbf{B} \mathbf{a} / \mathbf{a}^T \mathbf{W} \mathbf{a} \) ranges between 0 and 1, where low values correspond to projections that display little class differences and high values correspond to projections that have large differences between the classes. To extend to an arbitrary-dimensional projection, we consider a test statistic used in multivariate analysis of variance (MANOVA), Wilks \( \Lambda^* = |\mathbf{W}| / |\mathbf{W} + \mathbf{B}| \).

This quantity also ranges between 0 and 1, although the interpretation of numerical values are reversed from the 1-dimensional measure defined above. Small values of \( \Lambda^* \) correspond to large difference between the classes.

Let \( \mathbf{A} = [\mathbf{a}_1 \ \mathbf{a}_2 \ \cdots \ \mathbf{a}_k] \) define an orthogonal projection onto \( k \)-dimensional space. In projection pursuit the convention is that interesting projections are the ones that maximize the projection pursuit index, so we use the negative value of Wilks Lambda and add 1 to keep this index between 0 and 1. Then higher values are the projections with larger differences between classes. This gives the LDA projection pursuit index (LDA index) as:

\[
I_{LDA}(\mathbf{A}) = \begin{cases} 
1 - \frac{|\mathbf{A}^T \mathbf{W} \mathbf{A}|}{|\mathbf{A}^T (\mathbf{W} + \mathbf{B}) \mathbf{A}|} & \text{for } |\mathbf{A}^T (\mathbf{W} + \mathbf{B}) \mathbf{A}| \neq 0 \\
0 & \text{for } |\mathbf{A}^T (\mathbf{W} + \mathbf{B}) \mathbf{A}| = 0
\end{cases}
\]

Low index values correspond to little difference between classes and high values correspond to large differences between classes. The next proposition quantifies the minimum and maximum values. For simplicity, we denote \( \mathbf{W} + \mathbf{B} \) as \( \Phi \).
**Proposition 1.** Let \( \text{rank}(\Phi) = p, \ k \leq \min(p, g) \). Then,

\[
1 - \prod_{i=1}^{k} \lambda_i \leq I_{LDA}(A) \leq 1 - \prod_{i=p-k+1}^{p} \lambda_i
\]  

where \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0 \) : eigenvalues of \( \Phi^{-1/2}W\Phi^{-1/2} \),

\( e_1, e_2, \cdots, e_p \) : corresponding eigenvectors of \( \Phi^{-1/2}W\Phi^{-1/2} \),

\( f_1, f_2, \cdots, f_p \) : eigenvectors of \( \Phi^{-1/2}B\Phi^{-1/2} \).

In (4), the right equality holds when \( A = \Phi^{-1/2}[e_p \ e_{p-1} \ \cdots \ e_{p-k+1}] = \Phi^{-1/2}[f_1 \ f_2 \ \cdots \ f_k] \) and the left equality holds when \( A = \Phi^{-1/2}[e_k \ e_{k-1} \ \cdots \ e_1] = \Phi^{-1/2}[f_{p-k+1} \ f_{p-k+2} \ \cdots \ f_p] \).

A problem arises for LDA when \( \text{rank}(W) = r < p \). We need to remove collinearity before applying LDA by variable selection. Otherwise, we need to modify the \( W^{-1} \) calculation, for example, to use the pseudo-inverse (pseudo LDA : Fukunaga, 1990) , or to use a ridge estimate instead of \( W \) such as regularized discriminant analysis (Friedman, 1989). This is not a problem encountered. For projection pursuit, because we make calculations in \( k \)-dimensional space instead of \( p \)-dimensional space, we can find interesting projections without an initial dimension reduction or modified \( W \) calculation. The next proposition show how the LDA index works when \( \text{rank}(W) < p \).

**Proposition 2.** Let \( \text{rank}(\Phi) = r < p, \ k \leq \min(r, g) \). Then,

\[
1 - \prod_{i=1}^{k} \delta_i \leq I_{LDA}(A) \leq 1 - \prod_{i=r-k+1}^{r} \delta_i
\]  

\[ (5) \]
Figure 1. (a) Huber’s plot (1990) using $I_{LDA}$ on data simulated from two bivariate normal population. The symbols . $\bigcirc$ and $+$ represent two different classes. The solid line represents $I_{LDA}$ value for all 1-dimensional projections, and the dashed line is a guide set at the median $I_{LDA}$ value. The straight dotted line (b) represents the optimal projection, showing the middle plot (b) as a histogram and the dotted line denoted as (c) represents the projection corresponding to, shown as a histogram in (c).

where \[ \Phi = \begin{bmatrix} P & Q \end{bmatrix} \begin{bmatrix} \Lambda & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} P^T \\ Q^T \end{bmatrix} = P \Lambda P^T : \text{spectral decomposition of } \Phi, \]

$P : k \times r$ matrix, $P^T P = I_r$,

$Q : k \times (k-r)$ matrix, $Q^T Q = I_{k-r}$,

$\Lambda = \text{diag}[\delta_1, \delta_2, \cdots, \delta_r] : r \times r$ diagonal matrix,

$\delta_1, \delta_2, \cdots, \delta_r : \text{eigenvalues of } \Lambda^{-1/2}P^T W \Lambda^{-1/2}$,

$e_1, e_2, \cdots, e_r : \text{corresponding eigenvectors of } \Lambda^{-1/2}P^T W \Lambda^{-1/2}$.

In (5), the right equality holds when $A = P \Lambda^{-1/2}[e_r \ e_{r-1} \ \cdots \ e_{r-k+1}]$, and the left equality holds when $A = P \Lambda^{-1/2}[e_k \ e_{k-1} \ \cdots \ e_1]$.

The proofs of these two propositions are provided in Lee (2003). To illustrate the behavior of the LDA PP index (Figure 1), we use a type of plot that was introduced by Huber (1990). In one-dimensional projections
from a 2-dimensional space, for \( \theta = 0^\circ, \cdots, 179^\circ \), the projection pursuit index is calculated using projection \( \mathbf{a}_\theta = (\cos \theta, \sin \theta) \) and displayed radially as a function of \( \theta \). In each figure, the data points are plotted in the center. The solid line represents the index value, \( I_{LDA} \), plotted at distances relative to the center. The dotted circle is a guide line plotted at the median index value.

Figure 1 shows how the LDA PP index works. Data are simulated from two normal distributions with the same variance, \( \Sigma = \begin{pmatrix} 1 & 0.95 \\ 0.95 & 1 \end{pmatrix} \), and different means, \( \mu_1 = \begin{pmatrix} -1 \\ 0.6 \end{pmatrix} \) and \( \mu_2 = \begin{pmatrix} 1 \\ -0.6 \end{pmatrix} \). Each group has 50 samples. Figure 1(a) shows that the LDA PP index function (solid line) is smooth and has a maximum value when the projected data reveals two separated classes. Figure 1(b) and figure 1(c) are the histograms of the optimal projected data using the LDA PP index and the projected data onto the first principal component. The LDA PP index finds separated class structure. Principal component analysis is commonly used to find revealing low-dimensional projections, but it really doesn’t work well in classification problems. Here, principal component analysis solves a different problem: It finds the most spread data structure.

The LDA PP index works well generally, but it has some problems in special circumstances. One special situation is 2-dimensional data generated from a uniform mixture of three Gaussian distributions, with identity variance-covariance matrices and centers at the vertices of an equilateral triangle. Figure 2(a) shows the theoretical case where three classes have the exact same variance-covariance matrix and three class means are the vertices of an equilateral triangle. In this case, all directions have the same LDA index values. The best projection is the full 2-dimensional data space. Figure 2(b) shows data simulated from this distribution. Because of the sampling, variances are slightly different in each class and the three means don’t lie exactly on an equilateral triangle. Therefore the optimal direction (the dotted straight line in Figure 2(b)) depends on the sampling variation. If a new sample is generated, a completely different optional projection will occur. This is not what we want in exploratory methods. We would like to be able to find all the interesting data structures, which in this case would be the three 1-dimensional projections revealing each group separated
Figure 2. A problem situation for $I_{\text{LDA}}$ (a) The theoretical case where the three classes have the exact same variance and the three class means come from the vertices of an equilateral triangle. All direction have exactly same $I_{\text{LDA}}$ values (solid circle). The best projection is really the full 2-dimensional data space! What happens in practice : (b) The generated data from three normal distributions with different means and same identity variance-covariance.

Three means are set at the vertices of an equilateral triangle. The optimal projection changes with simple variation from the other two groups. We extend this problem of the LDA PP index to define a new index that is able to detect interesting structures in this situation.

### 2.2 LDA extended projection pursuit index using $L_r$-norm

We start from the 1-dimensional index. Let $y_{ij} = a^T X_{ij}$ be a projected data onto a 1-dimensional space. In the LDA PP index, we use $a^T B a$ and $a^T W a$ as the measures of between-group and within-group variations, respectively. These two measures can be explained as the square of $L_2$ vector norm.

$$a^T B a = \sum_{i=1}^{g} \sum_{j=1}^{n_i} (\bar{y}_{ij} - \bar{y}_i)^2 = \{||M\bar{y}_g - 1_n\bar{y}_r||_2\}^2$$  \hspace{1cm} (6)

$$a^T W a = \sum_{i=1}^{g} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2 = \{||y - M\bar{y}_g||_2\}^2$$  \hspace{1cm} (7)

$$a^T \Phi a = \sum_{i=1}^{g} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2 = \{||y - 1_n\bar{y}_r||_2\}^2 = \{||M\bar{y}_g - 1_n\bar{y}_r||_2\}^2 + \{||y - M\bar{y}_g||_2\}^2$$  \hspace{1cm} (8)
where \( \mathbf{M} = \text{diag}(1_{n_1}, \cdots, 1_{n_g}) \), \( \mathbf{y}_g = [\bar{y}_1, \bar{y}_2, \cdots, \bar{y}_g]^T \), \( \mathbf{y} = [y_1^T, y_2^T, \cdots, y_g^T]^T \), and \( \mathbf{y}_i = [y_{i1}, y_{i2}, \cdots, y_{in_i}]^T \), and \( 1_n = [1, 1, \cdots, 1]^T : n \times 1 \) vector. This total sum of squares \( \Phi \) can be represented as the additive of \( \mathbf{B} \) and \( \mathbf{W} \). We extend to the \( L_r \) norm. Let

\[
\mathbf{B}_r = \left\{ \| \mathbf{M}\bar{y}_g - 1_n\bar{y}_. \|_r \right\}^r = \sum_{i=1}^{g} \sum_{j=1}^{n_i} |\bar{y}_i - \bar{y}_.|^r \quad (9)
\]

\[
\mathbf{W}_r = \left\{ \| \mathbf{y} - \mathbf{M}\bar{y}_g \|_r \right\}^r = \sum_{i=1}^{g} \sum_{j=1}^{n_i} |y_{ij} - \bar{y}_i|^r. \quad (10)
\]

Then

\[
\left\{ \| \mathbf{y} - 1_n\bar{y}_. \|_r \right\}^r = \sum_{i=1}^{g} \sum_{j=1}^{n_i} |y_{ij} - \bar{y}_.|^r \leq \sum_{i=1}^{g} \sum_{j=1}^{n_i} |\bar{y}_i - \bar{y}_.|^r + \sum_{i=1}^{g} \sum_{j=1}^{n_i} |y_{ij} - \bar{y}_i|^r = \mathbf{B}_r + \mathbf{W}_r. \quad (11)
\]

Even though the additivity does not hold for the \( L_r \) vector norm, \( \mathbf{B}_r \) and \( \mathbf{W}_r \) can be substitutes for the measures of between-group and within-group variabilities. We use these measures to define our new index.

The 1-dimensional \( L_r \) projection pursuit index (\( L_r \) PP index) is defined by

\[
I_{L_r}(\mathbf{a}) = \left( \frac{\mathbf{B}_r}{\mathbf{W}_r} \right)^{1/r} = \frac{\| \mathbf{M}\bar{y}_g - 1_n\bar{y}_. \|_r}{\| \mathbf{y} - \mathbf{M}\bar{y}_g \|_r} \quad (12)
\]

\[
= \left( \frac{\sum_{i=1}^{g} \sum_{j=1}^{n_i} (\bar{y}_i - \bar{y}_.)^r}{\sum_{i=1}^{g} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^r} \right)^{1/r}. \quad (13)
\]

Taking the ratio to the \( 1/r \) power, prevents this index value from getting too big. The 1-dimensional LDA PP index is a special case of this index when \( r = 2 \).

For a \( k \)-dimensional projection \( \mathbf{A} \), let \( \mathbf{Y}_{ij} = \mathbf{A}^T \mathbf{X}_{ij} = [y_{ij1}, y_{ij2}, \cdots, y_{ijk}]^T \) be a projected data onto the \( k \) dimensional space spanned by \( \mathbf{A} \). Then

\[
[A^T \mathbf{B} \mathbf{A}]_{lm} = \sum_{i=1}^{g} \sum_{j=1}^{n_i} (\bar{y}_{i,l} - \bar{y}_.) (\bar{y}_{i,m} - \bar{y}_.), \quad (14)
\]

\[
[A^T \mathbf{W} \mathbf{A}]_{lm} = \sum_{i=1}^{g} \sum_{j=1}^{n_i} (y_{ijl} - \bar{y}_{i,l}) (y_{ijm} - \bar{y}_{i,m}) \quad (15)
\]

where \( l, m = 1, 2, \cdots, k \). The diagonals of these matrices represent the variances of the between (or within) group for each variable and the off-diagonals represent covariances between variables. We take only the
Figure 3. Huber’s plot showing behavior of $I_{L_r}$ index on the special data that caused problems for $I_{LDA}$.  (a) $I_{L_1}$: The optimal projections separate each class from the other two.  (b) $I_{L_5}$: The optimal projections separate all three classes.  (c) $I_{L_5}$: The optimal projections separate each class from the other two.  When $r=2$ and $r=4$, the index is the same as $I_{LDA}$, shown in Figure 2(a).

diagonal parts of these between-group and within-group variance and extend these sums of squares to $L_r$ norms.  Then,

$$I_{L_r}(A) = \left( \frac{\sum_{i=1}^{k} \sum_{j=1}^{g} \sum_{l=1}^{n_i} (\mathbf{y}_{l,i} - \bar{\mathbf{y}}_{i,j})^r}{\sum_{i=1}^{k} \sum_{j=1}^{g} \sum_{l=1}^{n_j} (\mathbf{y}_{l,j} - \bar{\mathbf{y}}_{i,j})^r} \right)^{1/r}. \quad (16)$$

For detailed explanations, see Lee (2003).

Figure 3 shows how the new index $I_{L_r}$ ($r = 1, 2, 3$) for the special situation that caused problem for $I_{LDA}$.  When $r = 1$, all three optimal projections separate one class from the other two classes.  When $r = 3$, the optimal projections separate the three classes.  With the $L_5$ index, we found the same optimal projections as the $L_1$ index but the index function is smoother than the $L_1$ index.  When $r$ is 2 and 4, these indices have the same value for all directions, just like the LDA index.

The LDA index and the $L_r$ index ($r \geq 2$) are usually sensitive to outliers, mainly due to use distance based measures.  Even though the $L_1$ index is based on distances, this index is more robust to outliers than other indices.  Figure 4 shows how these indices work in the presence of an outlier.  In each plot, there are two classes (1 and 2).  The class 1 has 21 observations with one outlier and the class 2 has 20 observations.
Figure 4. The behavior of the $I_{L_r}$ in the presence of an outlier, using simulated data with 2 classes, where class 1 has an outlier. (a) Huber’s plot of $I_{L_1}$, (a-1) Histogram of the projected data onto the $L_1$ optimal projection (b) Huber’s plot of $I_{L_2}$, (b-1) Histogram of the projected data onto the $L_2$ optimal projection (c) Huber’s plot of $I_{L_3}$, (c-1) Histogram of the projected data onto the $L_3$ optimal projection.

The histogram of the optimal 1-dimensional projected data using the $L_1$ index (Figure 4 (a-1)) shows that the outlier is separated from two groups and the best projection is not affected by this outlier. When $r \geq 2$, the best projections are leveraged towards the direction of the outlier. With the exception of the outlier, the $L_1$ index provides a more separated view of the two classes than the best projection of the $L_r (r \geq 2)$ index.

3. Optimization

A good optimization procedure is an important part of projection pursuit. The purpose of projection pursuit optimization is to find all of the interesting projections, not only to find one global maximum, because sometimes the local maximum can reveal unexpectedly interesting data structure. For this reason, the projection pursuit optimization algorithm needs to be flexible enough to find global and local maxima.

Posse (1990) compared the several optimization procedures, and suggest a random search for finding
the global maximum of a projection pursuit index. Cook, et al (1995) use a grand tour alternated with a simulated annealing optimization of a projection pursuit index, to creating a continuous stream of projections that are displayed for exploratory visualization of multivariate data. Klein and Dubes (1989) showed that simulated annealing can produce results as good as those obtained by conventional optimization methods and this method performs well for large data sets.

Simulated annealing was first proposed by Kirkpatrick, et al (1983) as a method to minimize criterion functions that have many variables. The fundamental idea of simulated annealing methods is that a re-scaling parameter, called the “temperature”, allows control of the speed of convergence to an optimal value, whether the optimal value is global and local. For a criterion function \( h(\theta) \), called the “energy”, we start from the initial value \( \theta_0 \). A value, \( \theta^* \) is generated in the neighborhood of \( \theta_0 \). Then, \( \theta^* \) is accepted as a new value with probability \( \rho \), defined by the temperature and the energy difference between \( \theta_0 \) and \( \theta^* \). This probability \( \rho \) guards against getting trapped into a local minimum allowing the algorithm to visit a local maximum and then jump out and explore for other maxima. For detailed explanations, see Bertsimas and Tsitsiklis (1993).

For out projection pursuit optimization, we use two different temperatures, one \( (D_i) \) is for neighborhood definition, and the other \( (T_i) \) is for the probability \( \rho \). \( D_i \) is re-scaled by the predetermined cooling parameter \( c \) and \( T_i \) is defined by \( T_0 / \log(i + 1) \). Before we start, we need to choose the cooling parameter, \( c \), and the initial temperature, \( T_0 \). The cooling parameter, \( c \), decides how many iterations are needed to converge to the optimal value and whether the optimal value is a local maximum or a global maximum. The initial temperature, \( T_0 \), also controls the speed of convergence. If we use small \( c \), the optimal value can be found in a small number of iterations, but the optimal value might be a local maximum. If \( c \) is large, we need more iterations to get an optimal value, but this optimal value is more likely to be the global maximum.
Therefore this algorithm is quite flexible for finding local or global maxima.

**Simulated Annealing Optimization Algorithm for Projection Pursuit**

1. Set an initial projection, $A_0$, and calculate the initial projection pursuit index value $I_0 = I(A_0)$.

   For the $i$th iteration,

2. Generate a projection $A_i$ from $N_{D_i}(A_0)$,

   where $D_i = c^i$, $c$ is the predetermined cooling parameter in the range $(0,1)$,

   $N_{D_i}(A_0) = \{ A : A$ is an orthonormal projection with direction $A_0 + D_iB, \forall$ random projections $B \} $

3. Calculate $I_i = I(A_i), \Delta I_i = I_i - I_0, T_i = \frac{T_0}{\log(i+1)}$.

4. Set $A_0 = A_i$ and $I_0 = I_i$ with probability $\rho = \min\left(\exp\left(\frac{\Delta I_i}{T_i}\right), 1\right)$ and increase $i$ to $i+1$.

Repeat 2-4 until $\Delta I_i$ is small.

4. Application

DNA microarray technologies provide a powerful tool for analyzing thousands of genes simultaneously. Comparison of gene expression levels between samples is quite useful to obtain information about important genes and their functions. Because microarrays contain large number of genes on each chip but typically few chips are used, analyzing DNA microarray data usually means dealing with large $p$, small $n$ challenges. A recent publication compares classification methods for gene expression data (Dudoit, et al., 2002) has focused on the classification error. We will use the same data sets to demonstrate the use of new PP indices.

4.1 Data sets

**Leukemia** This data set originated from a study of gene expression in two types of acute leukemias, acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). The data set consists of $n_1 = 25$ cases of AML and $n_2 = 47$ cases of ALL (38 cases of B-cell ALL and 9 cases of T-cell ALL), giving $n = 72$. After pre-
processing, we have \( p = 3571 \) human genes. This data set is available at http://www-genome.wi.mit.edu/mpr
and was described by Golub, et al. (1999).

**NCI60** This data set consists of 8 different tissue types where cancer was found: \( n_1 = 9 \) cases from breast, 
\( n_2 = 5 \) cases from central nervous system (CNS), \( n_3 = 7 \) cases from colon, \( n_4 = 8 \) cases from leukemia, 
\( n_5 = 8 \) cases from melanoma, \( n_6 = 9 \) cases from non-small-cell lung carcinoma (NSCLC), \( n_7 = 6 \) cases from ovarian, and \( n_8 = 9 \) cases from renal, and \( p = 6830 \) human genes. Missing values are imputed by a 
simple \( k \) nearest-neighbor algorithm \( (k = 5) \). We use this data to show how to use exploratory projection 
pursuit classification when the number of classes is large. This data set is available at http://genome-
www.stanford.edu/sutech/download/nci60/index.html and was described by Ross, et al. (2000).

**Standardization and Gene Selection** The gene expression data were standardized so that each observa-
tion has mean 0 and variance 1. For gene selection, we use the ratio of between-group to within-group sums 
of squares.

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

(17)

where \( \bar{x}_{.,j} = (1/n) \sum_{i=1}^{n} x_{i,j} \) and \( \bar{x}_{k,j} = (\sum_{i=1}^{n} I(y_i = k) x_{i,j})/(\sum_{i=1}^{n} I(y_i = k)) \). At the beginning, we follow 
the original study (Dudoit, et al, 2002) and start with \( p = 40 \) for the leukemia data and \( p = 30 \) for the 
NCI60 data and discuss different numbers of genes later.

**4.2 Results**

1-dimensional projection

Figure 5 displays the histograms of the projected data onto the optimal 1-dimensional projections. For 
this application, we choose a very large cooling parameter \( 0.999 \) which gives us the global maximum. In 
the Leukemia data, when \( r=1 \) (Figure 5-a), the B-cell ALL class is separated from the other classes except 
for one case. When \( r = 2 \) (Figure 5-b), the three classes are almost separable when the \( L_2 \) index is used, 
which is the same result as for the LDA index. As \( r \) is increased, the index tends to separate the T-cell ALL
Figure 5. Leukemia data: 1-dimensional projection ($p=40$) (a) the histogram of the optimal projected data using $I_{L_1}$ (b) the histogram of the optimal projected data using $I_{L_2}$ (c) the histogram of the optimal projected data using $I_{L_3}$

The NCI60 data is a quite challenging example. For such a small number of observations, there are too many classes. For this data, we try an isolation method that applies projection pursuit iteratively and takes off one class at a time (Friedman and Tukey, 1974). The 8 classes are too many to separate with a single 1-dimensional projection. After finding one split, we apply projection pursuit to each partition. Usually one class is peeled off from the others in each step. The tree diagram in Figure 6 illustrates the steps. In the first step (Figure 6-a), we separate the Leukemia class from the others. At the second step, Colon class is separated (Figure 6-b). Then, the Renal, the Breast, the NSCLC, and the Melanoma classes are separated sequentially. Finally, the Ovarian and the CNS classes are separated.

2-dimensional projection

Figures 7 and 8 show the plot of the data projected onto the optimal 2-dimensional projections for the Leukemia data. All three classes separate easily using the LDA index. Using the $L_1$ index, the B-cell ALL class is separated with one exception - the same outlier of the result of the 1-dimensional projection in Figure
Figure 6. NCI60 data: the 1-dimensional projection ($p=30$). (a) The histogram of the optimal projection using the LDA PP index. Leukemia group is separated from the others - peel off Leukemia group. (b) Colon group is separated (c) Renal group is separated (d) Breast group is separated.
1. AML 2: B-cell ALL 3: T-cell ALL

Figure 7. Leukemia data: 2-dimensional projection (p=40). (a) $I_{LDA}$: The three classes are separated. (b) $I_{r_1}$: The B-cell ALL class is separated from the other two except for one case. (c) $I_{r_2}$: The three classes are separated, although the gap between classes 2 and 3 is small.

1: Breast 2: CNS 3: Colon 4: Leukemia 5: Melanoma 6: NSCLC 7: Ovarian 8: Renal

Figure 8. NCI60: the 2-dimensional projection (p=30). (a) $I_{LDA}$: The Leukemia and Colon classes are separated from the others. (b) $I_{r_1}$: The Leukemia class and Colon classes are separated from the others, but Colon class is not clearly separated. (c) Only the Leukemia class is separated from the others.
5(c). In the 2-dimensional case, the LDA index is only the same as the $L_2$ index if $B$ and $W$ are diagonal matrices. The best result is obtained using the $I_{LDA}$ index, where all three classes are clearly separated. In the NCI60 data, the Leukemia class is clearly separated from the others for all indices (Figure 8).

Classification

Table 1. Test set Error. Median and Upper quantile of the misclassified samples from 200 replications. ($n_{test} = 24$)

<table>
<thead>
<tr>
<th>Method</th>
<th>Median</th>
<th>Upper quantile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fisher’s Linear Discriminant Analysis (LDA)</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Diagonal linear discriminant analysis (DLDA)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Diagonal quadratic discriminant analysis (DQDA)</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>LDA PP method</td>
<td>2.5</td>
<td>4</td>
</tr>
<tr>
<td>$L_1$ PP method</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Even though our projection pursuit indices are developed for the exploratory data analysis, especially for the visual inspection, they can be used for classification. For comparison to the other methods, we show the results of Dudoit, et al(2001) on the Leukemia data with two classes: AML and ALL. For a 2/3 training set ($n_{train} = 48$), we calculate BW(Equation 17) values for each gene and select the 40 genes with the larger BW values. Using this 40 gene training set, find the optimal projection, then build a classifier using the rule (11-65) in Johnson and Wichern(2002), and compute the test error. This is repeated repeat this 200 times. The median and upper quantile of the test errors are summarized in Table 1. The results of Fisher’s LDA, DLDA, and DQDA are from Dudoit, et al (2001). As we expect, $I_{LDA}$ has similar results to Fisher’s LDA. The $L_1$ compares well with the other methods.

5. Discussion

We have proposed new projection pursuit indices for exploratory supervised classification and explored
their properties. In most applications, the LDA index works well to find a projection that has well-separated class structure. The \( L_r \) PP index can lead us to projections that have special features. With the \( L_1 \) index, we can get a projection that is robust to outliers. This index is useful for discovering outliers. As \( r \) is increased, the \( L_r \) PP index tends to be more sensitive to outliers. For exploratory supervised classification, we need to use several PP indices (at least LDA and \( L_1 \) PP indices) and examine different results. These indices can be used for obtaining a better understanding of the class structure in the data space and their projection coefficients help find the important variables that best separate classes (Lee, 2003). The insights learned from plotting the PP projections are useful when building a classifier and for assessing classifiers.

Projection pursuit methods can be applied to multivariate tree methods. Several authors have considered the problem of constructing tree-structured classifiers that have linear discriminants at each node. Friedman (1977) reported that applying Fisher’s linear discriminants, instead of univariate features, at some internal nodes was useful in building better trees. This is a similar approach to the isolation method that we applied to NCI 60 data (Figure 6).

A major issue revealed by the gene expression application is that when there are too few cases for variables the reliability of the classifications is questionable. When the number of genes is larger than the sample size, most of high dimensional space is empty and we can find a separating hyperplane that divides groups purely by chance (see Ripley, 1996). For more detailed discussion, see Lee (2003).

For a large number of variables, our simulated annealing optimization algorithm for PP is quite slow to find the global optimal projection. A faster annealing algorithm described by Ingber (1989) may be better.

Finally we have used the \( R \) language for this research and provide the classPP package (available at CRAN). These indices are also available for the guided tour in the software \( GGobi \) (http://www.ggobi.org).

**REFERENCE**


Statistics - Simulation and Computation, 19, 1143-1164.


