Mizuta, Masahiro

Working Paper

Dimension Reduction Methods

Papers, No. 2004,15

Provided in Cooperation with:
CASE - Center for Applied Statistics and Economics, Humboldt University Berlin

Suggested Citation: Mizuta, Masahiro (2004) : Dimension Reduction Methods, Papers, No. 2004,15, Humboldt-Universität zu Berlin, Center for Applied Statistics and Economics (CASE), Berlin

This Version is available at:
https://hdl.handle.net/10419/22189

Dimension Reduction Methods

Masahiro Mizuta

Information Initiative Center, Hokkaido University, Sapporo 060-0811, (Japan)
mizuta@cims.hokudai.ac.jp

1 Introduction

One characteristic of computational statistics is the processing of enormous amounts of data. It is now possible to analyze large amounts of high-dimensional data through the use of high-performance contemporary computers. In general, however, several problems occur when the number of dimensions becomes high. The first problem is an explosion in execution time. For example, the number of combinations of subsets taken from p variables is 2^p; when p exceeds 20, calculation becomes difficult pointing terms of computation time. When p exceeds 25, calculation becomes an impossible no matter what type of computer is used. This is a fundamental situation that arises in the selection of explanatory variables during regression analysis. The second problem is the sheer cost of surveys or experiments. When questionnaire surveys are conducted, burden is placed on the respondent because there are many questions. And since there are few inspection items to a patient, there are few the burdens on the body or on cost. The third problem is the essential restriction of methods. When the number of explanatory variables is greater than the data size, most methods are incapable of directly dealing with the data; microarray data are typical examples of this type of data.

For these reasons, methods for dimension reduction without loss of statistical information are important techniques for data analysis. In this chapter, we will explain linear and nonlinear methods for dimension reduction; linear methods reduce dimension through the use of linear combinations of variables, and nonlinear methods do so with nonlinear functions of variables. We will also discuss the reduction of explanatory variables in regression analysis. Explanatory variables can be reduced with several linear combinations of explanatory variables.

2 Linear Reduction of High-Dimensional Data

The p-dimensional data can be reduced into q-dimensional data using q linear combinations of p variables. The linear combinations can be considered as linear projection. Most methods for reduction involve the discovery of linear combinations of variables under set criterion. Principal component analysis (PCA) and projection pursuit are typical methods of this type. These methods will be described in the following subsections.

2.1 Principal Component Analysis

Suppose that we have observations of p variables size n; $\{\boldsymbol{x}_i; i = 1, 2, \cdots, n\}$ (referred to as X hereafter). PCA is conducted for the purpose of constructing linear combinations of variables so that their variances are large under certain conditions. A linear combination of variables is denoted by $\{\boldsymbol{a}^\top \boldsymbol{x}_i; i = 1, 2, \cdots, n\}$ (simply, $\boldsymbol{a}^\top \boldsymbol{X}$), where $\boldsymbol{a} = (a_1, a_2, \cdots, a_p)^\top$.

Then, the sample variance of $\boldsymbol{a}^\top \boldsymbol{X}$ can be represented by

$$V(\boldsymbol{a}^\top \boldsymbol{x}) = \boldsymbol{a}^\top \hat{\Sigma} \boldsymbol{a},$$

where $\hat{\Sigma} = V(\boldsymbol{X})$. $\boldsymbol{a}^\top \hat{\Sigma} \boldsymbol{a}$ is regarded as a p variable function of $(a_1, a_2, \cdots, a_p)$: $\phi(a_1, a_2, \cdots, a_p) = \boldsymbol{a}^\top \hat{\Sigma} \boldsymbol{a}$. To consider the optimization problem for ϕ, $\boldsymbol{a}$ is constrained to $\boldsymbol{a}^\top \boldsymbol{a} = 1$. This promblem is solved using Lagrange multipliers. The following Lagrange function is defined as

$$\begin{aligned} L(a_1, a_2, \cdots, a_p) &= \phi(a_1, a_2, \cdots, a_p) - \lambda_1 (\sum_{i=1}^{p} a_i^2 - 1) \\ &= \boldsymbol{a}^\top \hat{\Sigma} \boldsymbol{a} - \lambda_1 (\boldsymbol{a}^\top \boldsymbol{a} - 1), \end{aligned}$$

where λ is the Lagrange multiplier. L is partially differentiated with respect to $\boldsymbol{a} = (a_1, a_2, \cdots, a_p)^\top$ and λ_1, and the derivatives are equated to zero. We therefore obtain the simultaneous equations:

$$\begin{cases} 2\hat{\Sigma}\boldsymbol{a} - 2\lambda_1 \boldsymbol{a} = 0 \\ \boldsymbol{a}^\top \boldsymbol{a} - 1 = 0. \end{cases}$$

This is an eigenvector problem; the solution to this problem for $\boldsymbol{a} = (a_1, a_2, \cdots, a_p)^\top$ is a unit eigenvector of $\hat{\Sigma}$ corresponding to the largest eigenvalue. Let $\boldsymbol{a}$ be an eigenvector and let λ be an eigenvalue. We then have

$$\phi(a_1, a_2, \cdots, a_p) = V(\boldsymbol{a}^\top \boldsymbol{x}) = \boldsymbol{a}^\top \hat{\Sigma} \boldsymbol{a} = \lambda \boldsymbol{a}^\top \boldsymbol{a} = \lambda.$$

The eigenvector is denoted as $\boldsymbol{a}_1$. Then $\boldsymbol{a}_1^\top \boldsymbol{x}_i; i = 1, 2, \cdots, n$ are referred to as the first principal components. The first principal components are one-dimensional data that are the projection of the original data with the maximum variance. If all of the information for the data can be represented by the

first principal components, further calculation is unnecessary. However, the first principal components usually exhibit the "size factor" only, whereas we would like to obtain another projection, namely the second principal components $\boldsymbol{a}_2^\top \boldsymbol{x}_i$.

The second principal components serve to explain the maximum variance under the constraint and the fact that they are independent of the first principal components. In other words, the second principal components $\boldsymbol{a}_2^\top \boldsymbol{X}$ take the maximum variance under the constraints $\boldsymbol{a}_1^\top \boldsymbol{a}_2 = 0$ and $\boldsymbol{a}_2^\top \boldsymbol{a}_2 = 1$. The second principal components can also be derived with Lagrange multipliers;

$$L(a_1, a_2, \cdots, a_p, \lambda, \lambda_2) = \boldsymbol{a}^\top \hat{\Sigma} \boldsymbol{a} - \lambda \boldsymbol{a}_1^\top \boldsymbol{a} - \lambda_2 (\boldsymbol{a}^\top \boldsymbol{a} - 1).$$

L is partially differentiated with respect to $\boldsymbol{a} = (a_1, a_2, \cdots, a_p)^\top$, λ and λ_2, and the derivatives are equted to zero. The simultaneous equations below are obtained:

$$\begin{cases} 2\hat{\Sigma}\boldsymbol{a} - \lambda \boldsymbol{a}_1 - 2\lambda_2 \boldsymbol{a} = 0 \\ \boldsymbol{a}_1^\top \boldsymbol{a} = 0 \\ \boldsymbol{a}_2^\top \boldsymbol{a}_2 - 1 = 0. \end{cases}$$

We can obtain $\lambda = 0$ and λ_2 is another eigenvalue (not equal to λ_1). Since the variance of $\boldsymbol{a}_2^\top X$ is λ_2, the $\boldsymbol{a}_2$ must be the second largest eigenvalue of $\hat{\Sigma}$. $\{\boldsymbol{a}_2^\top \boldsymbol{x}_i; i = 1, 2, \cdots, n\}$ are referred to as the second principal components. The third principal components, fourth principal components, $\cdots$, and the p-th principal components can then be derived in the same manner.

Proportion and Accumulated Proportion

The first principal components through the p-th principal components were defined in the discussions above. As previously mentioned, the variance of the k-th principal components is λ_k. The sum of variances of p variables is $\sum_{j=1}^{p} \hat{\sigma}_j = trace(\hat{\Sigma})$, where $\hat{\Sigma} = (\hat{\sigma}_{ij})$. It is well known that $trace(\hat{\Sigma}) = \sum_{j=1}^{p} \lambda_j$; the sum of the variances coincides with the sum of the eigenvalues. The proportion of the k-th principal components is defined as the proportion of the entire variance to the variance of the k-th principal components:

$$\frac{\lambda_k}{\sum_{j=1}^{p} \lambda_j}.$$

The first principal components through the k-th principal components are generally used consecutively. The total variance of these principal components is represented by the accumulated proportion :

$$\frac{\sum_{j=1}^{k} \lambda_j}{\sum_{j=1}^{p} \lambda_j}.$$

We have explained PCA as an eigenvalue problem of covariance matrix. However, the results of this method are affected by units of measurements

or scale transformations of variables. Thus, another method is to employ a correlation matrix rather than a covariance matrix. This method is invariant under units of variables, but does not take the variances of the variables into account.

2.2 Projection Pursuit

PCA searches a lower dimensional space that captures the majority of the variation within the data and discovers linear structures in the data. This method, however, is ineffective in analyzing nonlinear structures, *i.e.* curves, surfaces or clusters. In 1974, Friedman and Tukey (1974) proposed projection pursuit to search for linear projection onto the lower dimensional space that robustly reveals structures in the data. After that, many researchers developed new methods for projection pursuit and evaluated them (e.g. Huber, 1985; Friedman, 1987; Hall, 1989; Iwasaki, 1991; Nason, 1995; Koyama et al., 1998). The fundamental idea behind projection pursuit is to search linear projection of the data onto a lower dimensional space their distribution is "interesting"; "interesting" is defined as being "far from the normal distribution", *i.e.* the normal distribution is assumed to be the most uninteresting. The degree of "far from the normal distribution" is defined as being a projection index, the details of which will be described later.

Algorithm

The use of a projection index makes it possible to execute projection pursuit with the projection index. Here is the fundamental algorithm of k-dimensional projection pursuit.

1. Sphering $\boldsymbol{x}$: $\boldsymbol{z}_i = \hat{\Sigma}_{xx}^{-\frac{1}{2}}(\boldsymbol{x}_i - \hat{\boldsymbol{x}})$ $(i = 1, 2, \cdots, n)$, where $\hat{\Sigma}$ is the sample covariance matrix and $\hat{\boldsymbol{x}}$ is the sample mean of $\boldsymbol{x}$.
2. Initialize the project direction: $\alpha = (\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots, \boldsymbol{\alpha}_k)$.
3. Search the direction α that maximizes the projection index.
4. Project the data onto the lower dimensional space and display or analyze them.
5. Change the initial direction and repeat steps 3 and 4, if necessary.

Projection Indexes

The goal of projection pursuit is to find a projection that reveals *interesting* structures in the data. There are various standards for interestingness, and it is a very difficult task to define. Thus, the normal distribution is regarded as uninteresting, and uninterestingness is defined as a degree that is "far from the normal distribution."

Projection indexes are defined as of this degree. There are many definitions for projection indexes. Projection pursuit searches projections based on the

projection index; methods of projection pursuit are defined by the projection indexes.

Here we will present several projection indexes. It is assumed that $\boldsymbol{Z} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)$ is the result of sphering $\boldsymbol{X}$; the mean vector is a zero vector and the covariance matrix is an identity matrix.

Friedman's Index

Friedman (1987) proposed the following projection index:

$$I = \frac{1}{2}\sum_{j=1}^{J}(2j+1)\left[\frac{1}{n}\sum_{i=1}^{n}P_j(2\Phi(\boldsymbol{\alpha}^\top \boldsymbol{Z}_i)-1)\right]^2,$$

where $P_j(\cdot)$ are Legendre polynomials of order j and $\Phi(\cdot)$ is the cumulative distribution function of the normal distribution and J is a user-defined constant number, *i.e.* the degree of approximation.

In the case of two-dimensional projection pursuit, the index is represented by

$$\begin{aligned} I = &\sum_{j=1}^{J}(2j+1)E^2[P_j(R_1)]/4 \\ &+\sum_{k=1}^{J}(2k+1)E^2[P_k(R_2)]/4 \\ &+\sum_{j=1}^{J}\sum_{k=1}^{J-j}(2j+1)(2k+1)E^2[P_j(R_1)P_k(R_2)]/4, \end{aligned}$$

where

$$\begin{aligned} X_1 = \boldsymbol{\alpha}_1^\top \boldsymbol{Z}, \quad X_2 = \boldsymbol{\alpha}_2^\top \boldsymbol{Z} \\ R_1 = 2\Phi(X_1) - 1, \quad R_2 = 2\Phi(X_2) - 1. \end{aligned}$$

Moment Index

The third and higher cumulants of the normal distribution vanish. The cumulants are sometimes used for the test of normality, *i.e.* they can be used for the projection index. Jones and Sibson (1987) proposed a one-dimensional projection index named the "moment index," with the third cumulant $k_3 = \mu_3$ and the fourth cumulant $k_4 = \mu_4 - 3$:

$$I = k_3^2 + \frac{1}{4}k_4^2.$$

For two-dimensional projection pursuit, the moment index can be defined as

$$I = (k_{30}^2 + 3k_{21}^2 + 3k_{12}^2 + k_{03}^2) + \frac{1}{4}(k_{40}^2 + 4k_{31}^2 + 6k_{22}^2 + 4k_{13}^2 + k_{04}^2).$$

Hall's Index

Hall (1989) proposed the following projection index:

$$I = [\theta_0(\boldsymbol{\alpha}) - 2^{-1/2}\pi^{-1/4}]^2 + \sum_{j=1}^{J} \theta_j^2(\boldsymbol{\alpha}),$$

where

$$\theta_j(\boldsymbol{\alpha}) = n^{-1} \sum_{i=1}^{n} P_j(\boldsymbol{\alpha}^\top \boldsymbol{Z}_i)\phi(\boldsymbol{\alpha}^\top \boldsymbol{Z}_i),$$

$$P_j(z) = \frac{\sqrt{2}}{\sqrt{j!}}\pi^{1/4} H_j(2^{1/2}z),$$

$\phi(z)$ is the normal density function and $H_j(z)$ are the Hermite polynomials of degree j. J is a user-defined constant number. Hall's index is much more robust for outliers than Freidman's index.

Relative Projection Pursuit

The main objective of ordinary projection pursuit is the discovery of non-normal structures in a dataset. Non-normality is evaluated using the degree of difference between the distribution of the projected dataset and the normal distribution.

There are times in which it is desired that special structures be discovered using criterion other than non-normal criterion. For example, if the purpose of analysis is to investigate a feature of a subset of the entire dataset, the projected direction should be searched so that the projected distribution of the subset is far from the distribution of the entire dataset. In sliced inverse regression (please refer to the final subsection of this chapter), the dataset is divided into several subsets based on the values of the response variable, and the effective dimension-reduction direction is searched for using projection pursuit. In this application of projection pursuit, projections for which the distributions of the projected subsets are far from those of the entire dataset are required. Mizuta (2002) proposed the adoption of relative projection pursuit for these purposes. Relative projection pursuit finds *interesting* low-dimensional space that differs from the reference dataset predefined by the user.

3 Nonlinear Reduction of High-Dimensional Data

In the previous section, we discussed linear methods *i.e.* methods for dimension reduction through the use of linear projections. We will now move on to nonlinear methods for dimension reduction. First, we will describe a generalized principal component analysis (GPCA) method that is a nonlinear

extension of PCA. Algebraic curve fitting methods will then be mentioned for a further extension of GPCA. Finally, we will introduce principal curves *i.e.* the parametric curves that pass through the middle of the data.

3.1 Generalized Principal Component Analysis

As long as data have a near-linear structure, the singularities of the data can be pointed out using PCA. On the contrary, if data have a nonlinear structure, GPCA will not be adequate for drawing conclusions regarding the nature of the data. To overcome this difficulty, GPCA has been proposed by Gnanadesikan and Wilk (1969), whereby fitting functions to the data points can be discovered.

Suppose that we have observations of p variables $\boldsymbol{x} = (x_1, x_2, \cdots, x_p)$ on each of n individuals. Let $f_i(\boldsymbol{x})(i = 1, 2, \cdots, k)$ be k real-valued functions of the original variables.

The aim of GPCA is to discover a new set of variables (or functions of $\boldsymbol{x}$), as denoted by $z_1, z_2, \cdots, z_k$, which are mutually uncorrelated and whose variances decrease, from first to last. Each $z_j(j = 1, 2, \cdots, k)$ is considered to be a linear combination of $f_i(\boldsymbol{x})(i = 1, 2, \cdots, k)$, so that

$$z_j = \sum_{i=1}^{k} l_{ij} f_i(\boldsymbol{x}) = \boldsymbol{l}_j^\top \boldsymbol{f}(\boldsymbol{x}),$$

where $\boldsymbol{l}_j = (l_{1j}, l_{2j}, \cdots, l_{kj})^\top$ are k constant vectors such that $\boldsymbol{l}_j^\top \boldsymbol{l}_j = 1$, and $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_k(\boldsymbol{x}))^\top$. The vectors $\boldsymbol{l}_1, \boldsymbol{l}_2, \cdots, \boldsymbol{l}_k$ are the eigenvectors of the covariance matrix of $(f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_k(\boldsymbol{x}))$, as in PCA. The function z_k defined by the "smallest" eigenvalue is considered to be one of the fitting functions to the data.

PCA is a special case of GPCA: real-valued functions $f_i(\boldsymbol{x})$ are reduced to $x_i(i = 1, 2, \cdots, p)$.

Quadratic principal component analysis (QPCA) is specified by the following functions:

$$\begin{cases} f_i(\boldsymbol{x}) = x_i & (i = 1, 2, \cdots, p) \\ f_i(\boldsymbol{x}) = x_j x_m & (i = p+1, \cdots, (p^2+3p)/2), \end{cases}$$

where j, m is uniquely determined by

$$\begin{aligned} i &= \{(2p - j + 3)j/2\} + m - 1, \\ & 1 \leq j \leq m \leq p, \end{aligned}$$

for $i(i = p+1, \cdots, (p^2+3p)/2)$.

QPCA for two dimensional data is defined by

$$
\begin{aligned}
f_1(x, y) &= x \\
f_2(x, y) &= y \\
f_3(x, y) &= x^2 \\
f_4(x, y) &= xy \\
f_5(x, y) &= y^2.
\end{aligned}
$$

Most GPCA methods are not invariant under orthogonal transformations and/or the translations (parallel transformations) of a coordinate system, though PCA is invariant under them. For example, QPCA is not invariant under them. The expression "the method is invariant" in this subsection means that the results of the method are never changed in the original coordinate by coordinate transformation. In the following, the determination of the GPCA methods that are invariant under the orthogonal transformations of a coordinate system will be described in the case of two variables. Translations of a coordinate system are disregarded here because the data can be standardized to have a zero mean vector.

Hereafter, let us assume the following conditions:

A1 $f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_k(\boldsymbol{x})$ are linearly independent as functions of $\boldsymbol{x}$.
A2 For any orthogonal matrix T, there is a matrix W such that $\boldsymbol{f}(T\boldsymbol{x}) \equiv W\boldsymbol{f}(\boldsymbol{x})$.
A3 $f_i(\boldsymbol{x})$ are continuous functions.

Conditions **A1** and **A3** may be proper for GPCA, and condition **A2** is necessary for discussing the influence of orthogonal coordinate transformations. PCA and QPCA clearly satisfy these conditions.

A GPCA method is referred to as "invariant" if its results in the original coordinate system are not changed by the orthogonal transformation of a coordinate system. It can be mathematically described as follows. For any orthogonal coordinate transformation: $\boldsymbol{x}^* = T\boldsymbol{x}$,

$$
\begin{aligned}
z_j^* &= \boldsymbol{l}_j^{*\top} \boldsymbol{f}(\boldsymbol{x}^*) \\
&= \boldsymbol{l}_j^{*\top} \boldsymbol{f}(T\boldsymbol{x}) (j = 1, 2, \cdots, k)
\end{aligned}
$$

denote the results of the method for transformed variables $\boldsymbol{x}^*$, where $\boldsymbol{l}_j^*$ are eigenvectors of $Cov(\boldsymbol{f}(\boldsymbol{x}^*))$. The method is "invariant" if it holds that

$$
\boldsymbol{l}_j^\top \boldsymbol{f}(\boldsymbol{x}) \equiv \pm \boldsymbol{l}_j^{*\top} \boldsymbol{f}(T\boldsymbol{x}) (j = 1, 2, \cdots, k)
$$

as vector-valued functions of $\boldsymbol{x}$ for any orthogonal matrix T. The plus or minus sign is indicated only for the orientations of the eigenvectors.

The GPCA method specified by $\boldsymbol{f}(\boldsymbol{x})$ is invariant under an orthogonal transformation, if and only if the matrix W is an orthogonal matrix for any orthogonal matrix T. The proof will be described below. If the method is invariant, W can be taken as

$$(\boldsymbol{l}_1^*, \boldsymbol{l}_2^*, \cdots, \boldsymbol{l}_k^*)(\boldsymbol{l}_1, \boldsymbol{l}_2, \cdots, \boldsymbol{l}_k)^\top,$$

which is an orthogonal matrix. Conversely, if W is an orthogonal matrix, $W^\top \boldsymbol{l}_j^*$ are eigenvectors of $Cov(\boldsymbol{f}(\boldsymbol{x}))$. Therefore the following is obtained:

$$\boldsymbol{l}_j^\top = \pm \boldsymbol{l}_j^{*\top} W.$$

Mizuta (1983) derived a theorem on invariant GPCA.

Theorem 1. *GPCA methods for two-dimensional data (x, y) under the conditions* **A1**, **A2** *and* **A3** *that are invariant under rotations can be restricted to those specified by the following functions.*
(1) s pairs of functions:

$$\begin{cases} f_{2i-1}(x,y) = g_i(\sqrt{x^2+y^2}) \left(x^{N_i} - \binom{N_i}{2} y^2 x^{N_i-2} + \binom{N_i}{4} y^4 x^{N_i-4} - \cdots \right) \\ \qquad\qquad -h_i(\sqrt{x^2+y^2}) \left(N_i y x^{N_i-1} - \binom{N_i}{3} y^3 x^{N_i-3} + \binom{N_i}{5} y^5 x^{N_i-5} - \cdots \right) \\ f_{2i}(x,y) \quad = g_i(\sqrt{x^2+y^2}) \left(N_i y x^{N_i-1} - \binom{N_i}{3} y^3 x^{N_i-3} + \binom{N_i}{5} y^5 x^{N_i-5} - \cdots \right) \\ \qquad\qquad +h_i(\sqrt{x^2+y^2}) \left(x^{N_i} - \binom{N_i}{2} y^2 x^{N_i-2} + \binom{N_i}{4} y^4 x^{N_i-4} - \cdots \right) \end{cases}$$

$$(i = 1, 2, \cdots, s),$$

where g_i, h_i are arbitrary continuous functions of $\sqrt{x^2+y^2}$ and N_i are arbitrary positive integers.
(2) Continuous functions of $\sqrt{x^2+y^2}$.

The above theorem can be extended for use with GPCA methods for p-dimensional data because invariant GPCA for p-dimensional data methods are invariant under the rotations of any pair of two variables and the reverse is also true.

We will show some set of functions for invariant GPCA here.
(1) 3 dimensional and degree 1:

$$x, y, z.$$

(2) 3 dimensional and degree 2:

$$x^2, y^2, z^2, \sqrt{2}xy, \sqrt{2}yz, \sqrt{2}zx.$$

(3) 3 dimensional and degree 3:

$$x^3, y^3, z^3, \sqrt{3}x^2y, \sqrt{3}y^2z, \sqrt{3}z^2x, \sqrt{3}xy^2, \sqrt{3}yz^2, \sqrt{3}zx^2, \sqrt{6}xyz.$$

(4) 3 dimensional and degree q:

$$\sqrt{\frac{q!}{i!j!k!}} x^i y^j z^k$$
$$(i + j + k = q; 0 \leq i, j, k).$$

(5) p dimensional and degree q:

$$\sqrt{\frac{q!}{\prod_{i=1}^{p} k_t!}} \prod_{t=1}^{p} (x_t)_t^k$$
$$\sum_{t=1}^{p} k_t = q; 0 \leq k_t.$$

3.2 Algebraic Curve and Surface Fitting

Next, we will discuss a method involving algebraic curve and surface fitting to multidimensional data.

The principal component line minimizes the sum of squared deviations in each of the variables. The PCA cannot find non-linear structures in the data. GPCA is used to discover an algebraic curve fitted to data; the function z_k defined by the "smallest" eigenvalue is considered to be one of the fitting functions to the data. However, it is difficult to interpret algebraic curves statistically derived form GPCA.

We will now describe methods for estimating the algebraic curve or surface that minimizes the sum of squares of perpendicular distances from multidimensional data.

Taubin (1991) developed an algorithm for discovering the algebraic curve for which the sum of *approximate* squares distances between data points and the curve is minimized. The approximate squares distance does not always agree with the *exact* squares distance. Mizuta (1995) and Mizuta (1996) presented an algorithm for evaluating the exact distance between the data point and the curve, and have presented a method for algebraic curve fitting with exact distances. In this subsection, we describe the method of algebraic surface fitting with exact distances. The method of the algebraic curve fitting is nearly identical to that of surface fitting, and is therefore omited here.

Algebraic Curve and Surface

A p-dimensional algebraic curve or surface is the set of zeros of k-polynomials $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), \cdots, f_k(\boldsymbol{x}))$ on $\boldsymbol{R}^p$,

$$Z(\boldsymbol{f}) = \{\boldsymbol{x} : \boldsymbol{f}(\boldsymbol{x}) = 0\}.$$

In the case of $p = 2$ and $k = 1$, $Z(f)$ is a curve in the plane. For example, $Z(x^2 + 2y^2 - 1)$ is an ellipse and $Z(y^2 - x^2 + 1)$ is a hyperbola. In the case of $p = 3$ and $k = 2$, $Z(\boldsymbol{f})$ is a curve in the space.

In the case of $p = 3$ and $k = 1$, $Z(f)$ is a surface:

$$Z(f) = \{(x, y, z) : f(x, y, z) = 0\}.$$

Hereafter, we will primarily discuss this case.

Approximate Distance

The distance from a point $\boldsymbol{a}$ to the surface $Z(f)$ is usually defined by

$$\mathrm{dist}(\boldsymbol{a}, Z(f)) = \inf(\| \boldsymbol{a} - \boldsymbol{y} \| : \boldsymbol{y} \in Z(f)).$$

It was said that the distance between a point and the algebraic curve or surface cannot be computed using direct methods. Thus, Taubin proposed *an approximate distance* from $\boldsymbol{a}$ to $Z(f)$ (Taubin, 1991). The point $\hat{\boldsymbol{y}}$ that approximately minimizes the distance $\| \boldsymbol{y} - \boldsymbol{a} \|$, is given by

$$\hat{\boldsymbol{y}} = \boldsymbol{a} - (\nabla f(\boldsymbol{a})^\top)^+ f(\boldsymbol{a}),$$

where $(\nabla f(\boldsymbol{a})^\top)^+$ is the pseudoinverse of $\nabla f(\boldsymbol{a})^\top$. The distance from $\boldsymbol{a}$ to $Z(f)$ is approximated to

$$\mathrm{dist}(\boldsymbol{a}, Z(f))^2 \approx \frac{f(\boldsymbol{a})^2}{\| \nabla f(\boldsymbol{a}) \|^2}.$$

Taubin also presented an algorithm to find the algebraic curve for which the sum of *approximate* squares distances between data points and the curve is minimized.

Exact Distance

In the following, we present a method for calculating the distance between a point $\boldsymbol{a} = (\alpha, \beta, \gamma)$ and an algebraic surface $Z(f)$.

If (x, y, z) is the nearest point to the point $\boldsymbol{a} = (\alpha, \beta, \gamma)$ on $Z(f)$, (x, y, z) satisfies the following simultaneous equations:

$$\begin{cases} \phi_1(x, y, z) = 0 \\ \phi_2(x, y, z) = 0 \\ f(x, y, z) \ = 0, \end{cases} \tag{1}$$

where $\phi_1(x, y, z) = (x - \alpha)\frac{\partial f}{\partial y} - (y - \beta)\frac{\partial f}{\partial x}$, and $\phi_2(x, y, z) = (z - \gamma)\frac{\partial f}{\partial y} - (y - \beta)\frac{\partial f}{\partial z}$.

The equations (1) can be solved using the Newton-Rapson method:

1. Set x_0, y_0 and z_0 (see below).
2. Solve the equations:

$$\begin{cases} h\frac{\partial \phi_1}{\partial x} + k\frac{\partial \phi_1}{\partial y} + l\frac{\partial \phi_1}{\partial z} = -\phi_1(x, y, z) \\ h\frac{\partial \phi_2}{\partial x} + k\frac{\partial \phi_2}{\partial y} + l\frac{\partial \phi_2}{\partial z} = -\phi_2(x, y, z) \\ h\frac{\partial f}{\partial x} + k\frac{\partial f}{\partial y} + l\frac{\partial f}{\partial z} \quad = -f(x, y, z). \end{cases} \tag{2}$$

3. Replace x, y:
$$\begin{cases} x_{i+1} = x_i + h \\ y_{i+1} = y_i + k \\ z_{i+1} = z_i + l. \end{cases}$$
4. Stop if $h^2 + k^2 + l^2$ is below a certain threshold. Otherwise, go to STEP 2.

One of the important points to consider when applying the Newton-Rapson method is to compute an initial point. We have a good initial point: (α, β, γ).

When $x_0 = \alpha, y_0 = \beta, z_0 = \gamma$, the equations (2) are

$$\begin{cases} h\frac{\partial \phi_1}{\partial x} + k\frac{\partial \phi_1}{\partial y} + l\frac{\partial \phi_1}{\partial z} = 0 \\ h\frac{\partial \phi_2}{\partial x} + k\frac{\partial \phi_2}{\partial y} + l\frac{\partial \phi_2}{\partial z} = 0 \\ h\frac{\partial f}{\partial x} + k\frac{\partial f}{\partial y} + l\frac{\partial f}{\partial z} = -f(x, y, z). \end{cases}$$

It is very simple to show that the distance between (x_1, y_1, z_1) and (α, β, γ) agrees with Taubin's approximate distance.

Algebraic Surface Fitting

We have already described the method for calculating the distance between a point and a surface.

The problem of finding a fitting surface that minimizes the sum of the distances from data points can therefore be solved by using an optimization method without derivatives. However, for computing efficiency, the partial derivatives of the sum of squares of distances from data with the coefficients of an algebraic curve are derived.

In general, a polynomial f in a set is denoted by

$$f(b_1, \cdots, b_q; x, y, z),$$

where $b_1, \cdots, b_q$ are the parameters of the set.

Let $\boldsymbol{a}_i = (\alpha_i, \beta_i, \gamma_i)(i = 1, 2, \cdots, n)$ be n data points within the space. The point in $Z(f)$ that minimizes the distance from $(\alpha_i, \beta_i, \gamma_i)$ is denoted by $(x_i, y_i, z_i)(i = 1, 2, \cdots, n)$.

The sum of squares of distances is

$$R = \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{a}_i)^\top (\boldsymbol{x}_i - \boldsymbol{a}_i).$$

R can be minimized with respect to the parameters of polynomial f with the *Levenberg-Marquardt Method*. This method requires partial derivatives of R with respect to b_j:

$$\frac{\partial R}{\partial b_j} = \sum_{i=1}^{n} \frac{\partial R_i}{\partial b_j}, \tag{3}$$

where

$$\frac{\partial R_i}{\partial b_j} = 2\left((x_i - \alpha_i)\frac{\partial x_i}{\partial b_j} + (y_i - \beta_i)\frac{\partial y_i}{\partial b_j} + (z_i - \gamma_i)\frac{\partial z_i}{\partial b_j}\right). \tag{4}$$

The only matter left to discuss is a solution for $\frac{\partial x_i}{\partial b_j}$, $\frac{\partial y_i}{\partial b_j}$ and $\frac{\partial z_i}{\partial b_j}$. Hereafter, the subscript i is omitted. By the derivative of both sides of $f(b_1, \cdots, b_q, x, y, z) = 0$ with respect to b_j $(j = 1, \cdots, q)$, we obtain

$$\frac{\partial f}{\partial x}\frac{\partial x}{\partial b_j} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial b_j} + \frac{\partial f}{\partial z}\frac{\partial z}{\partial b_j} + \frac{df}{db_j} = 0, \tag{5}$$

where $\frac{df}{db_j}$ is the differential of f with b_j when x and y are fixed.

Since $\boldsymbol{x}_i$ is on the normal line from $\boldsymbol{a}_i$,

$$\left(\left.\frac{\partial f}{\partial x}\right|_{\boldsymbol{x}_i}, \left.\frac{\partial f}{\partial y}\right|_{\boldsymbol{x}_i}, \left.\frac{\partial f}{\partial z}\right|_{\boldsymbol{x}_i}\right)^\top (\boldsymbol{x}_i - \boldsymbol{a}_i) = 0.$$

By the derivative of

$$\begin{aligned}
(y - \beta)(z - \gamma)\left.\frac{\partial f}{\partial x}\right|_{\boldsymbol{x}} &= t \\
(x - \alpha)(z - \gamma)\left.\frac{\partial f}{\partial y}\right|_{\boldsymbol{x}} &= t \\
(x - \alpha)(y - \beta)\left.\frac{\partial f}{\partial z}\right|_{\boldsymbol{x}} &= t
\end{aligned}$$

with respect to b_j, we obtain the linear combinations of $\frac{\partial x}{\partial b_j}$, $\frac{\partial y}{\partial b_j}$ and $\frac{\partial z}{\partial b_j}$:

$$c_{1m}\frac{\partial x}{\partial b_j} + c_{2m}\frac{\partial y}{\partial b_j} + c_{3m}\frac{\partial z}{\partial b_j} + c_{4m} = \frac{\partial t}{\partial b_j}, \tag{6}$$

where $c_{1m}, \ldots, c_{4m}$ are constants $(m = 1, \cdots, 3)$.

Equations (5) and (6) are simultaneous linear equations in four variables $\frac{\partial x}{\partial b_j}$, $\frac{\partial y}{\partial b_j}$, $\frac{\partial z}{\partial b_j}$ and $\frac{\partial t}{\partial b_j}$. We then obtain $\frac{\partial x}{\partial b_j}$, $\frac{\partial y}{\partial b_j}$ and $\frac{\partial z}{\partial b_j}$ at (x_i, y_i, z_i). By equation (4), we have the partial differentiation of R_i with respect to b_j.

Therefore, we can obtain the algebraic curve that minimizes the sum of squares of distances from data points with the Levenberg-Marquardt method.

Bounded and Stably Bounded Algebraic Curve and Surface

Although algebraic curves can fit the data very well, they usually contain points far remote from the given data set. In 1994, Keren et al. (1994) and Taubin et al. (1994) independently developed algorithms for a *bounded* (closed) algebraic curve with approximate squares distance. We will now introduce the definition and properties of a bounded algebraic curve.

$Z(f)$ is referred to as *bounded* if there exists a constant r such that $Z(f) \subset \{\boldsymbol{x} :\| \boldsymbol{x} \|< r\}$. For example, it is clear that $Z(x^2+y^2-1)$ is bounded, but $Z(x^2-y^2)$ is not bounded.

Keren et al. (1994) defined $Z(f)$ to be *stably bounded* if a small perturbation of the coefficients of the polynomial leaves its zero set bounded. An algebraic curve $Z((x-y)^4+x^2+y^2-1)$ is bounded but not stably bounded because $Z((x-y)^4+x^2+y^2-1+\varepsilon x^3)$ is not bounded for any $\varepsilon \neq 0$.

Let $f_k(x,y)$ be the form of degree k of a polynomial $f(x,y)$: $f(x,y) = \sum_{k=0}^{d} f_k(x,y)$. The leading form of a polynomial $f(x,y)$ of degree d is defined by $f_d(x,y)$. For example, the leading form of $f(x,y) = x^2+2xy-y^2+5x-y+3$ is $f_2(x,y) = x^2+2xy-y^2$.

Lemma 1. *For an even positive integer d, any leading form $f_d(x,y)$ can be represented by $\boldsymbol{x}^\top A\boldsymbol{x}$. Where A is a symmetric matrix and $\boldsymbol{x} = (x^{\frac{d}{2}}, x^{\frac{d}{2}-1}y, \cdots, xy^{\frac{d}{2}-1}, y^{\frac{d}{2}})^\top$.*

Theorem 2. *(Keren et al., 1994): The $Z(f)$ is stably bounded if and only if d is even and there exists a symmetric positive definite matrix A such that*

$$f_d(x,y) = \boldsymbol{x}^\top A\boldsymbol{x},$$

where $\boldsymbol{x} = (x^{\frac{d}{2}}, x^{\frac{d}{2}-1}y, \cdots, xy^{\frac{d}{2}-1}, y^{\frac{d}{2}})^\top$.

These definitions and theorem for algebraic curves are valid for algebraic surfaces. Hereafter, we will restrict our discussion to algebraic *surfaces*.

Parameterization

We parameterize the set of all polynomials of degree k and the set of polynomials that induce (stably) bounded algebraic surfaces. In general, a polynomial f of degree p with q parameters can be denoted by $f(b_1, \cdots, b_q; x, y)$, where $b_1, \cdots, b_q$ are the parameters of the polynomial.

For example, *all of the polynomials* of degree 2 can be represented by

$$f(b_1, b_2, \cdots, b_{10}; x, y, z) = B^\top X,$$

where $X = (1, x, y, z, x^2, y^2, z^2, xy, yz, zx)^\top, B = (b_1, b_2, \cdots, b_{10})^\top$.

For *stably bounded algebraic curves* of degree 4,

$$\begin{aligned} & f(b_1, \cdots, b_{41}; x, y, z) \\ = & (x^2, y^2, z^2, xy, yz, zx) A^2 (x^2, y^2, z^2, xy, yz, zx)^\top \\ + & (b_{22}, \cdots, b_{41})(1, x, y, z, \cdots, z^3)^\top, \end{aligned}$$

where

$$A = \begin{pmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 \\ b_2 & b_7 & b_8 & b_9 & b_{10} & b_{11} \\ b_3 & b_8 & b_{12} & b_{13} & b_{14} & b_{15} \\ b_4 & b_9 & b_{13} & b_{16} & b_{17} & b_{18} \\ b_5 & b_{10} & b_{14} & b_{17} & b_{19} & b_{20} \\ b_6 & b_{11} & b_{15} & b_{18} & b_{20} & b_{21} \end{pmatrix}.$$

Examples

Here we will show a numerical example of the algebraic surface and bounded algebraic surface fitting methods.

The data in this example is three-dimensional data of size 210. The 210 points nearly lie on a closed cylinder (Fig. 1). The result of GPCA is set for an initial surface and the method is used to search for a fitting algebraic surface of degree 4 (Figs. 2, 3 and 4). The value of R is 0.924.

Fig. 5 presents the result of a *bounded* algebraic surface fitting the same data. The value of R is 1.239, and is greater than that of unbounded fitting. The bounded surface, however, directly reveals the outline of the data.

In this subsection, we have discussed algebraic surface fitting to multidimensional data. Two sets of algebraic surfaces were described: an unbounded algebraic surface and a bounded algebraic surface. This method can be extended for use with any other family of algebraic surfaces.

Taubin (1994) proposed the approximate distance of order k and presented algorithms for rasterizing algebraic curves. The proposed algorithm for exact distance can also be used for rasterizing algebraic curves and surfaces. Mizuta (1997) has successfully developed a program for rasterizing them with exact distances.

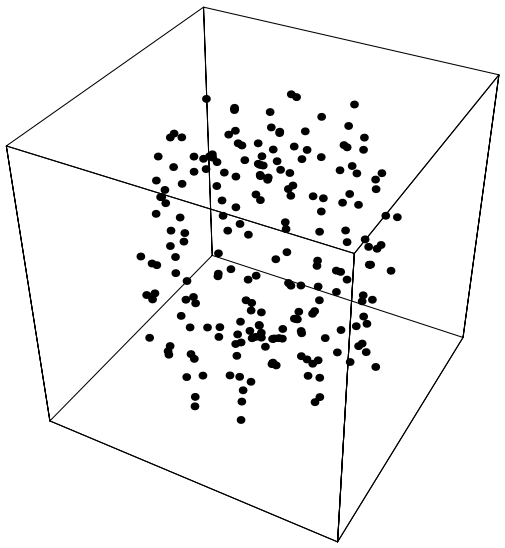

Fig. 1. Surface fitting for disturbed cylinder data (Original Data Points)

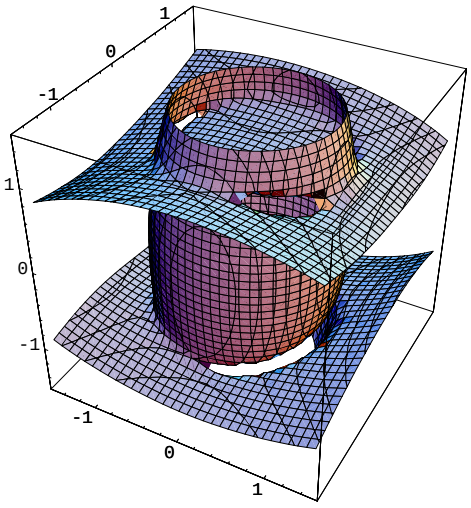

Fig. 2. Surface fitting for disturbed cylinder data (Unbounded Fitting Surface)

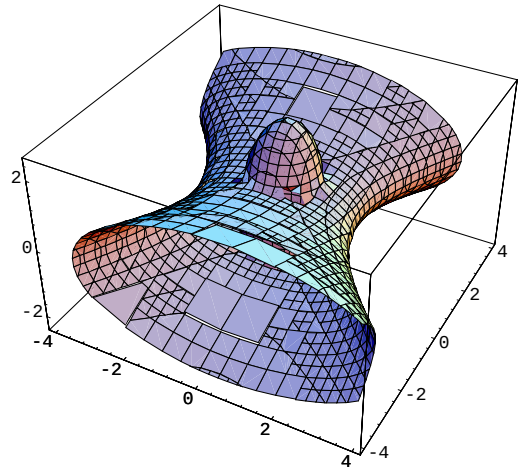

Fig. 3. Surface fitting for disturbed cylinder data (Global View of 2)

3.3 Principal Curves

Curve fitting to data is an important method for data analysis. When we obtain a fitting curve for data, the dimension of the data is nonlinearly reduced to one dimension. Hastie and Stuetzle (1989) proposed the concept of a prin-

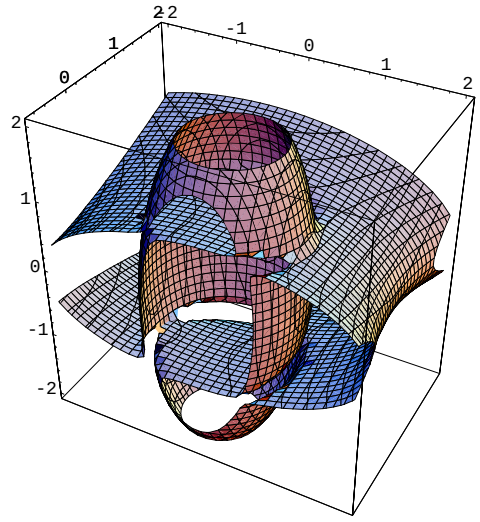

Fig. 4. Surface fitting for disturbed cylinder data (Cutting View of 2)

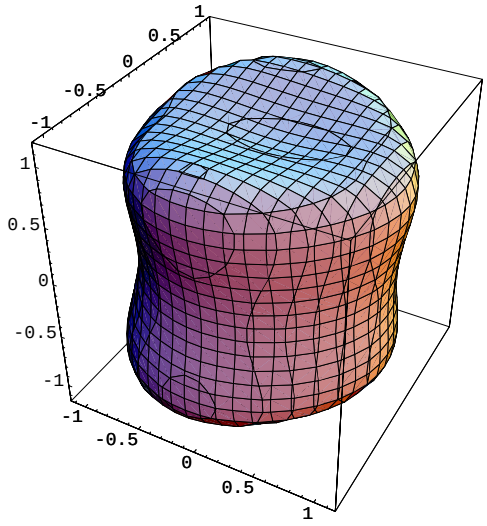

Fig. 5. Surface fitting for disturbed cylinder data (Bounded Fitting Surface)

cipal curve and developed a concrete algorithm to find the principal curve, which is represented by a parametric curve. We can therefore obtain a new nonlinear coordinate for the data using the principal curve.

Definition of Principal Curve

First, we will define principal curves for a p-dimensional distribution function $h(\boldsymbol{x})(\boldsymbol{x} \in R^p)$, rather than a dataset.

The expectation of X with density function h in R^p is denoted by $E_h(X)$. The parametric curve within the p-dimensional space is represented by $\boldsymbol{f}(\lambda)$, where λ is the parameter.

For each point $\boldsymbol{x}$ in R^p, the parameter λ of the nearest point on the curve $\boldsymbol{f}(\lambda)$ is denoted by $\lambda_{\boldsymbol{f}}(\boldsymbol{x})$, which is referred to as the *projection index*. The projection index, which is different from projection index in projection pursuit, is defined as follows:

$$\lambda_{\boldsymbol{f}}(\boldsymbol{x}) = \sup_{\lambda}\{\lambda | \parallel \boldsymbol{x} - \boldsymbol{f}(\lambda) \parallel = \inf_{\mu} \parallel \boldsymbol{x} - \boldsymbol{f}(\mu) \parallel\}.$$

The curve $\boldsymbol{f}(\lambda)$ is referred to as the principal curve of density function h, if

$$E_h(\boldsymbol{x} \parallel \lambda_f(\boldsymbol{x}) = \lambda) = \boldsymbol{f}(\lambda) \quad (\text{for a.e.}\lambda)$$

is satisfied. After all, for any point $\boldsymbol{f}(\lambda)$ on the curve, the average of the conditional distribution of $\boldsymbol{x}$ given $\lambda_f(\boldsymbol{x}) = \lambda$ is consistent with $\boldsymbol{f}(\lambda)$ with the exception of a set of measure 0.

The principal curves of a given distribution are not always unique. For example, two principal components of the two-dimensional normal distribution are principal curves.

The algorithm for finding the principal curves of a distribution is:

1. Initialization Put

$$\boldsymbol{f}^{(0)}(\lambda) = \bar{\boldsymbol{x}} + \boldsymbol{a}\lambda,$$

where $\boldsymbol{a}$ is the first principal component of the distribution defined by the density function h and $\bar{\boldsymbol{x}}$ is the average of $\boldsymbol{x}$.

2. Expectation Step (update of $\boldsymbol{f}(\lambda)$)

$$\boldsymbol{f}^{(j)}(\lambda) = E(\boldsymbol{x} | \lambda_{\boldsymbol{f}^{(j-1)}}(\boldsymbol{x}) = \lambda) \quad \forall\lambda$$

3. Projection Step (update of λ)

$$\lambda^{(j)}(\boldsymbol{x}) = \lambda_{\boldsymbol{f}^{(j)}}(\boldsymbol{x}) \quad \forall \boldsymbol{x} \in R^p$$

And transform the $\lambda^{(j)}$ to be arc length.

4. Evaluation Calculate

$$D^2(h, \boldsymbol{f}^{(j)}) = E_{\lambda^{(j)}} E\{\parallel \boldsymbol{x} - \boldsymbol{f}(\lambda^{(j)}(\boldsymbol{x})) \parallel^2 | \lambda^{(j)}(\boldsymbol{x})\}.$$

If the value

$$\frac{|D^2(h, \boldsymbol{f}^{(j-1)}) - D^2(h, \boldsymbol{f}^{(j)})|}{D^2(h, \boldsymbol{f}^{(j-1)})}$$

is smaller than ε, then stop, otherwise $j = j + 1$ and go to Step 1.

In the Expectation Step, calculate the expectation with respect to the distribution h of the set of $\boldsymbol{x}$ satisfying $\lambda_{\boldsymbol{f}^{(j-1)}}(\boldsymbol{x}) = \lambda$ and substitute $\boldsymbol{f}^{(j)}(\lambda)$ for it. In the Projection Step, project data points in R^p to the curve $\boldsymbol{f}^{(j)}(\lambda)$ and assign $\lambda^{(j)}(\boldsymbol{x})$.

For actual data analysis, only a set of data points is given and the distribution is unknown. Hastie and Stuetzle (1989) also proposed an algorithm with which to derive the principal curve for given p-dimensional data of size n: $x_{ik}(i = 1, 2, \cdots, N; k = 1, 2, \cdots, p)$. In this case, the principal curves are represented by lines determined by N points $(\lambda_i, \boldsymbol{f}_i)$.

1. Initialization

$$\boldsymbol{f}^{(0)}(\lambda) = \bar{\boldsymbol{x}} + \boldsymbol{u}\lambda,$$

where $\boldsymbol{u}$ is the first principal component of the data and $\bar{\boldsymbol{x}}$ is the average of $\boldsymbol{x}$.

2. Expectation Step Smooth x_{ik} $(i = 1, 2, \cdots, N)$ with respect to λ for each k independently and calculate $\boldsymbol{f}^{(j)}(\lambda)$.

3. Projection Step Search for the nearest point on the curve (line curve) of each data point and assign it to their value of λ.

4. Evaluation If a terminal condition is satisfied, the algorithm is stopped. If not, $j = j + 1$ and go to Step 2.

4 Linear Reduction of Explanatory Variables

Thus far, we have described dimension reduction methods for multidimensional data, where there are no distinctions among variables. However, there are times when we must analyze multidimensional data in which a variable is a response variable and others are explanatory variables. Regression analysis is usually used for the data. Dimension reduction methods of explanatory variables are introduced below.

Sliced Inverse Regression

Regression analysis is one of the fundamental methods used for data analysis. A response variable y is estimated by a function of explanatory variables $\boldsymbol{x}$, a p-dimensional vector. An immediate goal of ordinary regression analysis is to find the function of $\boldsymbol{x}$. When there are many explanatory variables in the data set, it is difficult to stably calculate the regression coefficients. An approach to reducing the number of explanatory variables is explanatory variable selection, and there are many studies on variable selection. Another approach is to project the explanatory variables on a lower dimensional space that nearly estimates the response variable.

Sliced Inverse Regression (SIR), which was proposed by Li (1991), is a method that can be employed to reduce explanatory variables with linear

projection. SIR finds linear combinations of explanatory variables that are a reduction for non-linear regression. The original SIR algorithm, however, cannot derive suitable results for some artificial data with trivial structures. Li also developed another algorithm, SIR2, which uses the conditional estimation $E[cov(\boldsymbol{x}|y)]$. However, SIR2 is also incapable of finding trivial structures for another type of data.

We hope that projection pursuit can be used for finding linear combinations of explanatory variables. A new SIR method with projection pursuit (SIRpp) is described here. We also present a numerical example of the proposed method.

Sliced Inverse Regression Model

SIR is based on the model (SIR model):

$$y = f(\boldsymbol{\beta}_1^\top \boldsymbol{x}, \boldsymbol{\beta}_2^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_K^\top \boldsymbol{x}) + \varepsilon, \tag{7}$$

where $\boldsymbol{x}$ is the vector of p explanatory variables, $\boldsymbol{\beta}_k$ are unknown vectors, ε is independent of $\boldsymbol{x}$, and f is an arbitrary unknown function on $\boldsymbol{R}^K$.

The purpose of SIR is to estimate the vectors $\boldsymbol{\beta}_k$ for which this model holds. If we obtain $\boldsymbol{\beta}_k$, we can reduce the dimension of $\boldsymbol{x}$ to K. Hereafter, we shall refer to any linear combination of $\boldsymbol{\beta}_k$ as the effective dimensional reduction (e.d.r.) direction.

Li (1991) proposed an algorithm for finding e.d.r. directions, and it was named SIR. However, we refer to the algorithm as SIR1 to distinguish it from the SIR model.

The main idea of SIR1 is to use $E[\boldsymbol{x}|y]$. $E[\boldsymbol{x}|y]$ is contained in the space spanned by e.d.r. directions, but there is no guarantee that $E[\boldsymbol{x}|y]$ will span the space. For example, in Li, if $(X_1, X_2) \sim N(0, I_2), Y = X_1^2$ then $E[X_1|y] = E[X_2|y] = 0$.

SIR Model and Non-Normality

Hereafter, it is assumed that the distribution of $\boldsymbol{x}$ is standard normal distribution: $\boldsymbol{x} \sim N(0, I_p)$. If not, standardize $\boldsymbol{x}$ by affine transformation. In addition, $\boldsymbol{\beta}_i^\top \boldsymbol{\beta}_j = \delta_{ij}, (i, j = 1, 2, \cdots, K)$ is presumed without loss of generality. We can choose $\boldsymbol{\beta}_i (i = K+1, \cdots, p)$ such that $\{\boldsymbol{\beta}_i\}$ $(i = 1, \cdots, p)$ is a basis for $\boldsymbol{R}^p$.

Since the distribution of $\boldsymbol{x}$ is $N(0, I_p)$, the distribution of $(\boldsymbol{\beta}_1^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_p^\top \boldsymbol{x})$ is also $N(0, I_p)$. The density function of $(\boldsymbol{\beta}_1^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_p^\top \boldsymbol{x}, y)$ is

$$h(\boldsymbol{\beta}_1^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_p^\top \boldsymbol{x}, y) = \phi(\boldsymbol{\beta}_1^\top \boldsymbol{x}) \cdots \phi(\boldsymbol{\beta}_p^\top \boldsymbol{x}) \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y - f(\boldsymbol{\beta}_1^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_K^\top \boldsymbol{x}))^2}{2\sigma^2}\right),$$

where $\phi(\boldsymbol{x}) = 1/\sqrt{2\pi} \exp(-x^2/2)$ and we assume $\varepsilon \sim N(0, \sigma^2)$.

The conditional density function is

$$h(\boldsymbol{\beta}_1^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_p^\top \boldsymbol{x} \mid y) = \phi(\boldsymbol{\beta}_{K+1}^\top \boldsymbol{x}) \cdots \phi(\boldsymbol{\beta}_p^\top \boldsymbol{x}) g(\boldsymbol{\beta}_1^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_K^\top \boldsymbol{x}),$$

where $g()$ is a function of $\boldsymbol{\beta}_1^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_K^\top \boldsymbol{x}$, which is not generally the normal density function.

Thus, $h(\boldsymbol{\beta}_1^\top \boldsymbol{x}, \cdots, \boldsymbol{\beta}_p^\top \boldsymbol{x} \mid y)$ is separated into the normal distribution part $\phi(\boldsymbol{\beta}_{K+1}^\top \boldsymbol{x}) \cdots \phi(\boldsymbol{\beta}_p^\top \boldsymbol{x})$ and the non-normal distribution part $g()$.

Projection Pursuit is an excellent method for finding non-normal parts, so we adopt it for SIR.

SIRpp Algorithm

Here we show the algorithm for the SIR model with projection pursuit (SIRpp). The algorithm for the data $(y_i, \boldsymbol{x}_i)$ $(i = 1, 2, \cdots, n)$ is as follows:

1. Standardize $\boldsymbol{x}$: $\tilde{\boldsymbol{x}}_i = \hat{\Sigma}_{\boldsymbol{xx}}^{-\frac{1}{2}}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})(i = 1, 2, \cdots, n)$, where $\hat{\Sigma}_{\boldsymbol{xx}}$ is the sample covariance matrix and $\bar{\boldsymbol{x}}$ is the sample mean of $\boldsymbol{x}$.
2. Divide the range of y into H slices, $I_1, \cdots, I_H$.
3. Conduct a projection pursuit in K dimensional space for each slice. The following H projections are obtained: $(\boldsymbol{\alpha}_1^{(h)}, \cdots, \boldsymbol{\alpha}_K^{(h)})$, $(h = 1, \cdots, H)$.
4. Let the K largest eigenvectors of $\hat{V}$ be $\hat{\boldsymbol{\eta}}_k (k = 1, \cdots, K)$. Output $\hat{\boldsymbol{\beta}}_k = \hat{\boldsymbol{\eta}}_k \Sigma_{\boldsymbol{xx}}^{-\frac{1}{2}} (k = 1, 2, \cdots, K)$ for the estimation of e.d.r. directions, where $\hat{V} = \sum_{h=1}^{H} w(h) \sum_{k=1}^{K} \boldsymbol{\alpha}_k^{(h)^\top} \boldsymbol{\alpha}_k^{(h)}$.

Numerical Examples

Two models of the multicomponent are used:

$$y = x_1(x_1 + x_2 + 1) + \sigma \cdot \varepsilon, \tag{8}$$

$$y = \sin(x_1) + \cos(x_2) + \sigma \cdot \varepsilon \tag{9}$$

to generate $n = 400$ data, where $\sigma = 0.5$. We first generate x_1, x_2, ε with N(0,1) and calculate response variable y using (8) or (9). Eight variables $x_3, \cdots, x_{10}$ generated by N(0,1) are added to the explanatory variables. The ideal e.d.r. directions are contained within the space spanned by two vectors $(1, 0, \cdots, 0)$ and $(0, 1, \cdots, 0)$.

The squared multiple correlation coefficient between the projected variable $\boldsymbol{b}^\top \boldsymbol{x}$ and the space B spanned by ideal e.d.r. directions:

$$R^2(\boldsymbol{b}) = \max_{\boldsymbol{\beta} \in B} \frac{(\boldsymbol{b}^\top \sum_{\boldsymbol{xx}} \boldsymbol{\beta})^2}{\boldsymbol{b}^\top \sum_{\boldsymbol{xx}} \boldsymbol{b} \cdot \boldsymbol{\beta}^\top \sum_{\boldsymbol{xx}} \boldsymbol{\beta}} \tag{10}$$

is adopted as the criterion for evaluating the effectiveness of estimated e.d.r. directions.

Table 1 shows the mean and the standard deviation (in parentheses) of $R^2(\hat{\boldsymbol{\beta}}_1)$ and $R^2(\hat{\boldsymbol{\beta}}_2)$ of four SIR algorithms for $H = 5, 10$, and 20, after 100 replicates. SIR2 cannot reduce the explanatory variables from the first example. The result of the second example is very interesting. SIR1 finds the

asymmetric e.d.r. direction, but, does not find the symmetric e.d.r. direction. Conversely, SIR2 finds only the symmetric e.d.r. direction. SIRpp can detect both of the e.d.r. directions.

The SIRpp algorithm performs well in finding the e.d.r. directions; however, the algorithm requires more computing power. This is one part of projection pursuit for which the algorithm is time consuming.

Table 1. Results for SIR1, SIR2, and SIRpp (Example 1)

	SIR1		SIR2		SIRpp	
H	$R^2(\hat{\beta}_1)$	$R^2(\hat{\beta}_2)$	$R^2(\hat{\beta}_1)$	$R^2(\hat{\beta}_2)$	$R^2(\hat{\beta}_1)$	$R^2(\hat{\beta}_2)$
5	.92	.77	.96	.20	.97	.78
	(.04)	(.11)	(.03)	(.21)	(.02)	(.15)
10	.93	.81	.92	.10	.95	.79
	(.03)	(.09)	(.09)	(.12)	(.04)	(.13)
20	.92	.76	.83	.11	.95	.75
	(.04)	(.18)	(.19)	(.13)	(.07)	(.18)

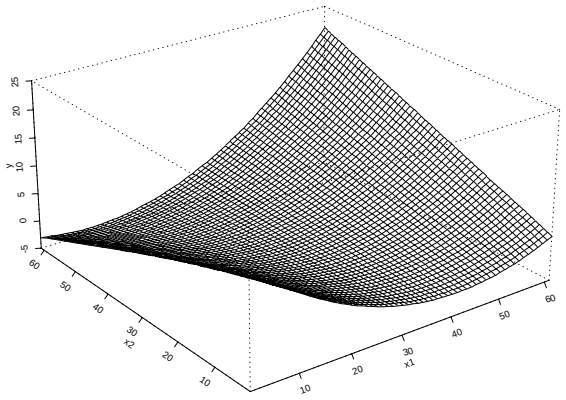

Fig. 6. Function of the example 1. Asymmetric function $y = x_1(x_1 + x_2 + 1) + \sigma \cdot \varepsilon$

5 Concluding Remarks

In this chapter, we discussed dimension reduction methods for data analysis. First, PCA methods were explained for the linear method. Then, pro-

Table 2. Results of SIR1, SIR2, and SIRpp (Example 2)

	SIR1		SIR2		SIRpp	
H	$R^2(\hat{\beta}_1)$	$R^2(\hat{\beta}_2)$	$R^2(\hat{\beta}_1)$	$R^2(\hat{\beta}_2)$	$R^2(\hat{\beta}_1)$	$R^2(\hat{\beta}_2)$
5	.97	.12	.92	.01	.92	.88
	(.02)	(.14)	(.04)	(.10)	(.05)	(.11)
10	.97	.12	.90	.05	.88	.84
	(.02)	(.15)	(.06)	(.07)	(.08)	(.13)
20	.97	.12	.85	.05	.84	.73
	(.02)	(.14)	(.09)	(.06)	(.10)	(.22)

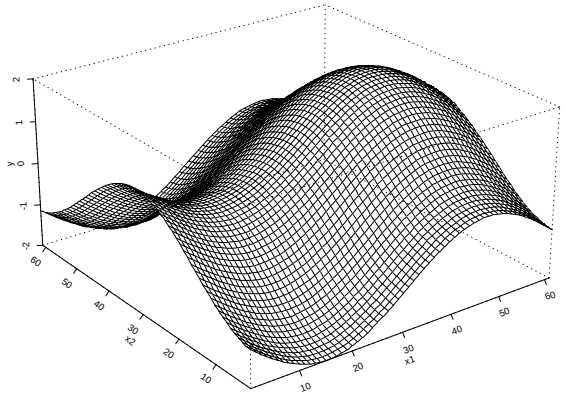

Fig. 7. Function of the example 2. Function of asymmetric with respect to the x_1 axis and symmetric with respect to x_2 axis. $y = \sin(x_1) + \cos(x_2) + \sigma \cdot \varepsilon$

jection pursuit methods were described. For nonlinear methods, GPCA algebraic curve fitting methods and principal curves were introduced. Finally, we explained sliced inverse regression for the reduction of the dimension of explanatory variable space.

These methods are not only useful for data analysis, but also effective for preprocessing when carrying out another data analysis. In particular, they are indispensable for the analysis of enormous amounts of and complex data, e.g. microarray data, log data on the Internet, etc. Research in this field will continue to evolve in the future.

References

Friedman, J. (1987). Exploratory projection pursuit. *Journal of the American Statistical Association*, 82:249–266.

Friedman, J. and Tukey, J. (1974). A projection pursuit algorithm for exploratory data analysis. *IEEE Transaction on Computer*, c-23(9):881–890.

Gnanadesikan, R. and Wilk, M. (1969). Data analytic methods. In Krishnaiah, P., editor, *Multivariate Analysis II*, pages 593–638. Academic Press.

Hall, P. (1989). On polynomial-based projection indices for exploratory projection pursuit. *Annals of Statistics*, 17:589–605.

Hastie, T. and Stuetzle, W. (1989). Principal curves. *Journal of the American Statistical Association*, 84:502–516.

Huber, P. (1985). Projection pursuit (with discussion). *Annals of Statistics*, 13:435– 475.

Iwasaki, M. (1991). Projection pursuit: the idea and practice (in japanese). *Bulletin of the Computational Statistics of Japan*, 4(2):41–56.

Jones, M. C. and Sibson, R. (1987). What is projection pursuit? (with discussion). *Journal of the Royal Statistical Society, Series A*, 150:1–36.

Keren, D., Cooper, D., and Subrahmonia, J. (1994). Describing complicated objects by implicit polynomials. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 16(1):38–53.

Koyama, K., Morita, A., Mizuta, M., and Sato, Y. (1998). Projection pursuit into three dimensional space (in japanese). *The Japanese Journal of Behaviormetrics*, 25(1):1–9.

Li, K. (1991). Sliced inverse regression for dimension reduction. *Journal of the American Statistical Association*, 86:316–342.

Mizuta, M. (1983). Generalized principal components analysis invariant under rotations of a coordinate system. *Journal of the Japan Statistical Society*, 14:1–9.

Mizuta, M. (1995). A derivation of the algebraic curve for two-dimensional data using the least-squares distance. In Escoufier, Y., Hayashi, C., Fichet, B., Ohsumi, N., Diday, E., Baba, Y., and Lebart, L., editors, *Data Science and Its Application*, pages 167–176. Academic Press, Tokyo.

Mizuta, M. (1996). Algebraic curve fitting for multidimensional data with exact squares distance. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pages 516–521.

Mizuta, M. (1997). Rasterizing algebraic curves and surfaces in the space with exact distances. In *Progress in Connectionist-Based Information Systems –Proceedings of the 1997 International Conference on Neural Information Processing and Intelligent Information Systems*, pages 551–554.

Mizuta, M. (2002). Relative projection pursuit. In Sokolowski, A. and Jajuga, K., editors, *Data Analysis, Classification, and Related Methods*, page 131. Cracow University of Economics.

Nason, G. (1995). Three-dimensional projection pursuit (with discussion). *Applied Statistics*, 44(4):411–430.

Taubin, G. (1991). Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 13(11):1115–1138.

Taubin, G. (1994). Distance approximations for rasterizing implicit curves. *ACM Transaction on Graphics*, 13:3–42.

Taubin, G., Cukierman, F., Sullivan, S., Ponce, J., and Kriegman, D. (1994). Parameterized families of polynomials for bounded algebraic curve and surface fitting. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 16(3):287–303.

Index