A Study of Nonlinear Principal Component Analysis Using Neural Networks

ニューラルネットワークを用いた非線形主成分分析に関する研究

March 2005

Major in Pure and Applied Physics
Graduate School of Science and Engineering
Waseda University

Ryo Saegusa

Contents

Chapter 1

Introduction

1.1 Background and objectives

In the field of data analysis, it is important to reduce the dimensionality of data, because it will help to extract new knowledge from the data and to decrease the computational cost. As a method of dimensionality reduction, Principal Component Analysis (PCA)[8] has been applied in various areas, such as data compression and pattern recognition.

PCA is an orthogonal transformation of a coordinate system in which the data are described. The basis of the objective coordinate system efficiently represents data distributed on a linear hyper plane as a coordinate value that is called a principal component. However, when n-dimensional data are distributed on a m $(< n)$-dimensional nonlinear manifold in a n-dimensional Euclidean space, more than the m dimensionality is required to represent the data in PCA which makes the dimensionality reduction inefficient.

In order to solve these problems, some methods of Non-Linear Principal Component Analysis (NLPCA) have been developed [3]. These methods can be roughly classified into three categories.

The methods in the first category are based on nonlinear transformation. Kernel PCA proposed by Schölkopf [21] is a method that performs linear PCA for the image of the input data mapped by a nonlinear mapping function. Consequently, the method constructs the ordered principal components as PCA does. However, an adequate way to determine the nonlinear mapping function for a given data set is not well established.

The methods in the second category are based on a piecewise linear approximation. Principal Curve proposed by Hastie et al. [7] summarizes the high-dimensional data with polygonal lines. In the method, however, it is difficult to construct more than two principal components. Local PCA proposed by Kambhalta et al. [11] divides an input space into local domains and performs PCA for each domain. However, the eigenbases of adjacent local domains are discontinuous, which prevents a global understanding of the distribution of the data.

The methods in the third category are based on neural networks such as the Sandglass-type Multi-Layered Perceptron (SMLP) proposed by Kramer [13] and Irie and Kawato [9]. The method can obtain the global and explicit nonlinear mapping functions to reduce the dimensionality. However, these methods do not provide a way for deciding the number of principal components aside from inefficient trial and error. Even if an adequate number of principal components are given after the trial and error, the method does not explicitly provide a parameter to measure the contribution of each principal component for data-representation, such as an eigenvalue in a conventional PCA. Moreover, the number of principal components has to be specified in advance. These drawbacks will limit the use of the method for practical purposes.

In this study, a novel method of nonlinear principal component analysis is proposed. The method preserves the order of principal components based on their contribution to the data-representation. Moreover, the method does not need to determine in advance the number of principal components to be used. In the proposed method, hierarchically arrayed neural networks are trained to build a set of nonlinear functions that map an input vector to its corresponding vector in the principal component space. The proposed model also has the ability to reconstruct high-dimensional data from its low dimensional representation in the principal component space. These functions are automatically adjusted during a training process. The property of these functions is discussed and the experimental results analyzed.

1.2 Organization of the dissertation

This dissertation is organized as follows.

In Chapter 2, conventional methods of PCA and NLPCA are reviewed. In the first section, PCA and NLPCA on multi-dimensional data are overviewed. In the next section, PCA and NLPCA are mathematically formulated. In the final section, the major methods of NLPCA are classified into three categories, and the properties and problems of each are pointed out.

In Chapter 3, a novel method of NLPCA is proposed. In the first section, the order of the principal components in PCA and nonlinear extension of PCA are discussed. In the next section, the implementation of the method using neural networks is described. The implementation makes possible the construction of nonlinear mapping functions and their iterative adjustment.

In Chapter 4 to 7, several experiments are demonstrated and the results are discussed.

In Chapter 4, two experimental results are presented. In the first experiment, three-dimensional artificial data sampled from a simple curved surface are utilized. The distribution of the data set is easy to understand. We examine the proposed method's construction of the curvilinear axes in the order of significance. In the second experiment, high dimensional data sampled from an artificial waveform are utilized. We examine the proposed method's extraction of the feature of the waveform with a small number of principal components.

In Chapter 5, the representational ability of the mapping functions is discussed. In the proposed method, each mapping function is implemented by a three-layered perceptron. In the first section, the representational ability of the mapping functions is described based on the degree of freedom of the network. In the next section, experiments are demonstrated to examine the representational ability. In the experiments, the representational ability is shown with variations in the degree of freedom, utilizing the three-dimensional artificial data used in the preceding chapter.

In Chapter 6, the distortion of the distribution is discussed with a comparison between PCA and the proposed method. In the first section, an experiment on data-reconstruction is demonstrated utilizing a set of iris data sampled from an open database. In the second section, we measure the distance of the input vectors and the distance of the reconstructed vectors. Then, we examine the distortion of these distances.

In Chapter 7, the effectiveness of the proposed method for practical problems is examined with several open databases. Two experiments on dimensionality reduction are demonstrated. One is a data compression of facial images. The other is a feature extraction for the classification of handwritten numerals.

In Chapter 8, we discuss NLPCA in regard to complexity and redundancy. These properties are originated from its nonlinear extension, which frees the PCA from constraints such as orthogonality. The comparison of the proposed method to other methods is described.

In Chapter 9, conclusion of the study and perspective are described.

A list of the references cited in this study and a list of the author's publications are attached at the end of the dissertation.

Chapter 2

A Review of PCA and NLPCA

In Chapter 2, conventional methods of PCA and NLPCA are reviewed. In the first section, PCA and NLPCA on multi-dimensional data are overviewed. In the next section, PCA and NLPCA are mathematically formulated. In the final section, the major methods of NLPCA are classified into three categories, and the properties and problems of each are pointed out.

2.1 An overview of PCA and NLPCA

An objective of PCA and NLPCA is a dimensionality reduction of multi-dimensional data which means to map the data in a high-dimensional space onto a low-dimensional space preserving characteristics of the distribution.

Let $\boldsymbol{x} \in R^n$, $\boldsymbol{y} \in R^m$ and $\hat{\boldsymbol{x}} \in R^n$ be coordinates of data in a high-dimensional space, their images (principal components) in a low-dimensional space and the reconstructed data in the original high-dimensional space, respectively. $n, m \in N$ $(n \geq m)$ indicates the dimensionality of the given data and the dimensionality of principal components, respectively. R and N represent a set of real numbers and natural numbers. In the following, we assume $E[\boldsymbol{x}] = 0$ for simplicity.

Figure 2.1 shows an example of a dimensionality reduction from R^2 to R^1. Figure 2.1 (a) and (b) show the appearances of the dimensionality reduction by PCA and NLPCA, respectively.

In Figure 2.1(a), PCA linearly maps a vector $\boldsymbol{x}^p \in R^2$ onto $y^p \in R^1$, and also linearly maps $y^p \in R^1$ onto $\hat{\boldsymbol{x}}^p \in R^2$. All of $\hat{\boldsymbol{x}}^p$ are mapped on a straight line because they are the images by linear mappings.

On the other hand, in Figure 2.1 (b), NLPCA nonlinearly maps a vector $\boldsymbol{x}^p \in R^2$ onto $y^p \in R^1$, and also nonlinearly maps $y^p \in R^1$ onto $\hat{\boldsymbol{x}}^p \in R^2$. If the nonlinear mapping functions are monotonous, all of $\hat{\boldsymbol{x}}^p$ are mapped on a curved line in R^2.

The performance of PCA and NLPCA is measured with the Mean Square Error (MSE) between original data and the data reconstructed from the principal components in a low-dimensional space into the original high-dimensional space. The smaller MSE indicates the higher fidelity. The criterion of MSE is

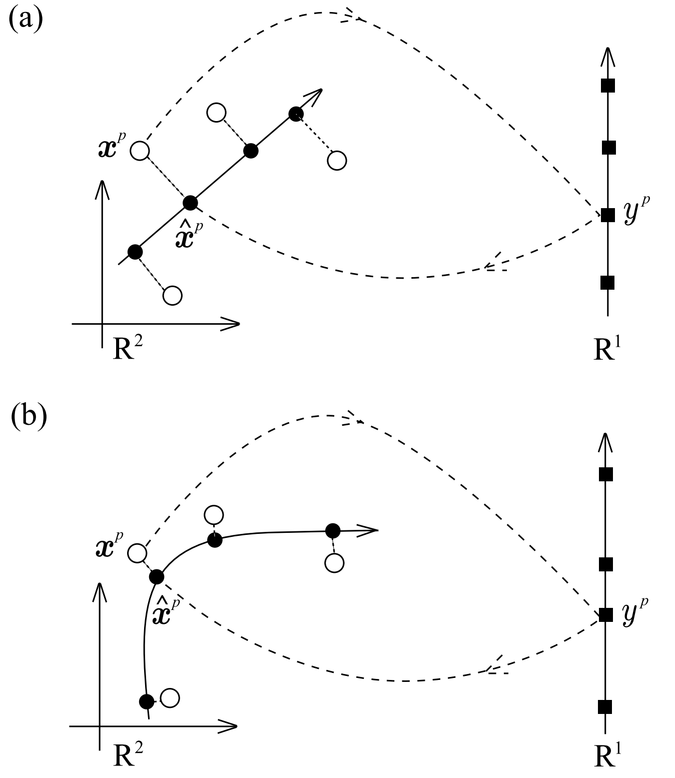

Figure 2.1. An concept of the dimensionality reduction of two-dimensional vectors onto one-dimensional vectors. The figure (a) and (b) show the appearances of dimensionality reduction by PCA and NLPCA, respectively. In the figure, a white circle, a black square, and a black circle represent the input vector $\boldsymbol{x}^p$, the principal component y^p, and the reconstructed vector $\hat{\boldsymbol{x}}^p$, respectively.

described as

$$\text{MSE} \quad = \quad E[||\boldsymbol{x} - \hat{\boldsymbol{x}}(\boldsymbol{y})||^2], \tag{2.1}$$

where $E[\cdot]$ represents the expectation and $||\cdot||$ represents L_2 norm.

In Figure 2.1(a), the MSE of PCA corresponds to the average of the distance between $\boldsymbol{x}^p$ and $\hat{\boldsymbol{x}}^p$ which is the projection onto the line. In Figure 2.1 (b), the MSE of NLPCA corresponds to the average of the distance between $\boldsymbol{x}^p$ and $\hat{\boldsymbol{x}}^p$ which is the mapped vector onto the curved line. In comparison with Figure 2.1 (a) and Figure 2.1 (b), the distances between $\boldsymbol{x}^p$ and $\hat{\boldsymbol{x}}^p$ in (b) are smaller than ones in (a), so that the MSE of (b) is also smaller. In the case of Figure 2.1, NLPCA represents the distribution of the data more accurately than PCA does.

Both PCA and NLPCA reduce the dimensionality of data with a linear and nonlinear mapping function constructed dependently on the data, respectively. Although PCA and NLPCA are considered to be equivalent in the meaning of the functional optimization, NLPCA may decrease the MSE more than PCA as shown in Figure 2.1 (a) and (b), because the mapping functions of NLPCA have a higher degree of freedom rather than the linear mapping functions of PCA have.

2.2 The formulation of PCA and NLPCA

In PCA, when we reduce the dimensionality of data in R^n to obtain their principal components in R^m ($n \geq m$), a linear mapping function $L^T : R^n \mapsto R^m$ is defined as

$$\begin{aligned} \boldsymbol{y} &= L^T \boldsymbol{x} && (2.2) \\ &= (\boldsymbol{e}_1^T \boldsymbol{x}, \boldsymbol{e}_2^T \boldsymbol{x}, \cdots, \boldsymbol{e}_m^T \boldsymbol{x})^T, && (2.3) \end{aligned}$$

while a linear mapping function $L : R^m \mapsto R^n$ is defined as

$$\begin{aligned} \hat{\boldsymbol{x}} &= L\boldsymbol{y} && (2.4) \\ &= \sum_{i=1}^{m} \boldsymbol{e}_i (\boldsymbol{e}_i^T \boldsymbol{x}), && (2.5) \end{aligned}$$

where L is a $n \times m$ matrix, $\boldsymbol{e}_i$ is the i-th column vector of the matrix L, and T represents a transposing operation. $\{\boldsymbol{e}_i\}_{i=1,\cdots,m}$ are orthonormal bases that satisfy the condition of

$$\boldsymbol{e}_i^T \boldsymbol{e}_j = \delta_{ij} \quad i, j = 1, \cdots, m, \tag{2.6}$$

where δ is the Kronecker delta so that when $i \neq j$, $\delta_{ij} = 0$, and when $i = j$, $\delta_{ii} = 1$.

In NLPCA, when we reduce the dimensionality of data in R^n to obtain their principal components in R^m, a nonlinear mapping function $\boldsymbol{\phi} : R^n \mapsto R^m$ is defined as

$$\boldsymbol{y} = \boldsymbol{\phi}\ (\boldsymbol{x}), \tag{2.7}$$

while a nonlinear mapping function $\boldsymbol{\psi} : R^m \mapsto R^n$ is defined as

$$\hat{\boldsymbol{x}} = \boldsymbol{\psi}\ (\boldsymbol{y}). \tag{2.8}$$

These mapping functions associate data with their principal components nonlinearly.

2.3 Conventional methods of NLPCA

The conventional methods of NLPCA are considered to be classified into three categories: transformation based models, piecewise linear models and neural-network based models. In this section, we review major methods of the categories.

2.3.1 Transformation based models of NLPCA

Transformation based models of NLPCA perform PCA for the transformed data by a mapping function.

Generalized PCA [5] proposed by Gnanadesikan is a nonlinear extension of PCA. As shown in the preceding section, the transformation of a coordinate system in PCA is described as

$$\begin{aligned} y_i &= \boldsymbol{e}_i \boldsymbol{x} && (2.9) \\ &= \sum_{j=1}^{n} e_{ij} x_j. && (2.10) \end{aligned}$$

When we replace x_j with $\psi_j(\boldsymbol{x})$ in this equation, the equation is represented as

$$y_i = \sum_{j=1}^{k} e_{ij} \psi_j(\boldsymbol{x}), \tag{2.11}$$

where k is the dimensionality of $\boldsymbol{\psi}(\boldsymbol{x})$. The dimensionality is not necessarily equal to the dimensionality of $\boldsymbol{x}$.

Generalized PCA performs conventional linear PCA for $\boldsymbol{\psi}(\boldsymbol{x})$ as a substitute of $\boldsymbol{x}$, so that the method performs NLPCA for $\boldsymbol{x}$, when $\boldsymbol{\psi}$ is a nonlinear mapping function of $\boldsymbol{x}$.

Kernel PCA (KPCA) [21] proposed by Schölkopf is also a transformation based model of NLPCA. In similar fashion, Kernel PCA maps a vector $\boldsymbol{x}$ onto a Hilbert space with a mapping function ψ. The method performs PCA for $\psi(\boldsymbol{x})$ in the mapped Hilbert space with a kernel method.

In the kernel method, the integral (a dot product between two functions) of $\psi(\boldsymbol{x}_1) \cdot \psi(\boldsymbol{x}_2)$ is replaced by a kernel function $K(\boldsymbol{x}_1, \boldsymbol{x}_2)$, therefore, we can prevent from the computation of integrals using the kernel function. When we use a kernel function, in most case, ψ is not given explicitly. However, the existence of the mapping function for a given kernel function is guaranteed by Mercer's theorem and we do not necessarily know the mapping function, if the operation in the Hilbert space is described with only the combination of the kernel functions such as PCA, discriminant analysis, and support vector machine.

These transformation based model seems to naturally extend PCA with nonlinear mapping functions, however, the adequate way to determine the function for the given data set is not well known. The parameters of the functions are searched by a large amount of trials and error. In some case, the dimensionality of principal components by KPCA can be larger than the dimensionality of original data, which results in that the KPCA is not applicable in dimensionality reduction.

2.3.2 Piecewise linear models of NLPCA

Piecewise linear models of NLPCA perform a linear PCA in a local domain of data space, and totally perform nonlinear PCA in the data space.

Principal Curve proposed by Hastie [7] summarizes the distribution of data in multi-dimensional space with a one-dimensional polygonal curve, which is called a principal curve.

In the method, $\boldsymbol{x}$ is projected onto a nearest point $\boldsymbol{f}(\lambda)$ on the curve, where λ corresponds the arc length of the curve and is measured from a starting point of the curve. λ is determined by the following equation,

$$\lambda_{\boldsymbol{f}}(\boldsymbol{x}) = \sup_{\lambda}\{\lambda : ||\boldsymbol{x} - \boldsymbol{f}(\lambda)|| = \inf_{\mu} ||\boldsymbol{x} - \boldsymbol{f}(\mu)||\}. \tag{2.12}$$

$\boldsymbol{f}(\lambda)$ is subject to the condition:

$$E(\boldsymbol{x}|\lambda_{\boldsymbol{f}}(\boldsymbol{x}) = \lambda) = \boldsymbol{f}(\lambda), \tag{2.13}$$

for any λ. The condition means that $\boldsymbol{f}(\lambda)$ is the average of all points which are projected onto the $\boldsymbol{f}(\lambda)$.

The principal curve is obtained by the following algorithm:

Initialization: Initialize the principal curve with the line determined by the first eigenvector of PCA.

1. Expectation: Reset $\boldsymbol{f}(\lambda)$ to be a average point.
2. Projection: Reset $\lambda(\boldsymbol{x})$ to be a arc length.
3. Evaluaton: Calculate the MSE of the representation.

Until: the change of MSE falls below some threshold.

In this method, we can avoid constructing a projection function of a vector onto the curve, because the algorithm gives a rule of the projection (of a vector onto the nearest point on the principal curve). However, the projection is not continuous at an ambiguity point such as a point on a symmetry line of the principal curve [14].

Local PCA proposed by Kambhalta [11] is another method of a piecewise linear PCA. The method performs the clustering of the data and the PCA for each cluster. The clustering and the PCA are processed simultaneously and iteratively to minimize the MSE on the reconstruction.

In Local PCA, a reference vector of a cluster is defined to be an average of the vectors which belongs to the cluster, and PCA is performed for the belonging vectors with the reference vector to be an origin. The iteration is continued until the average MSE of all clusters falls below some threshold.

In Local PCA, it takes more time to compress a vector than a neural network model. However, it takes less time to reconstruct a vector, because the cluster to which the vector belongs is determined after the compression.

A problem in Local PCA is that the principal component axes (eigenvectors) of the adjacent local domains are not continuous. The discontinuity is considered to be caused by no overlapping of the adjacent local domains to construct each set of axes. Therefore, it is difficult to represent a global location of the data with the constructed axes. Moreover, the iterative process of clustering and PCA for each local domain requires a greater computational cost when the number of data samples increases.

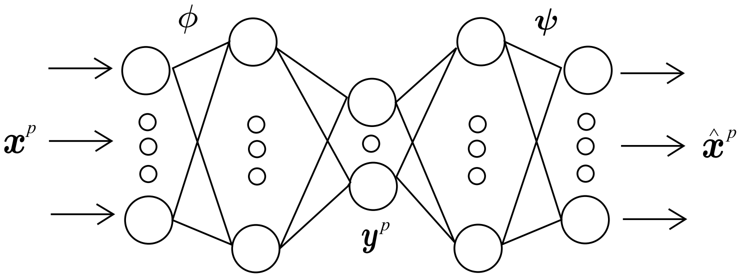

Figure 2.2. The autoassociator model by Kramer and Irie et al.

2.3.3 Neural Network based models of NLPCA

Neural network based models realize NLPCA using the nonlinearity of the units and the bottleneck network architecture. In the neural networks, the nonlinear mapping functions are originated from a nonlinear monotone activation function such as a sigmoid function and a radial basis function. A neural network works a global approximator of mapping functions. Theoretically, it is proved that multilayered-type neural networks realize any mapping functions in any precision, if the network has an enough number of units in hidden layers.

Autoassociator model proposed by Kramer [13] perform NLPCA uisng its bottleneck architecture which enables the network to compress and decompress the dimensionality of data. In the study [13], the performance of the model and a conventional PCA is compared with some criterion such as MSE, FPE and AIC. The architecture of the autoassociator model is shown in Figure 2.2.

In the autoassociator model, principal component scores can be obtained in the bottleneck layer. For practical use, the autoassociator model poses a problem that their principal components are not ordered in the contribution to represent the data, while the components of PCA are ordered. Kramer proposed a serial model as shown in Figure 2.3, where each autoassociator is corresponding to one principal component and the autoassociator are serially connected to make the components ordered. However, the serial model is not discussed in detail in the study [13].

In the serial model, the residual error vector of a higher autoassociator is inputted into a lower autoassociator. However, the nonlinearity of the distribution extracted by the higher autoassociator is not used to advantage in the lower autoassociator, since the residual error vector is calculated with a linear operation, and the linear operation does not reduce the dimensionality of nonlinearly distributed data. As shown in Figure 2.4, in a linear case, the residual error vectors in R^2 space are laid on a straight line in the space, but in a nonlinear case, the residual error vectors are not laid on a straight line, because the direction of the projections of the different points are different in the nonlinear case. The second autoassociator has to retry the dimensionality reduction from R^2 into R^1. The extracted nonlinearity by the higher autoassociator is not fed to the lower autoassociator.

Input optimization model proposed by Tan [24] performs NLPCA with training of a three-layer network and training of its inputs to minimize the MSE on data-reconstruction. The input of the network is a principal component vector.

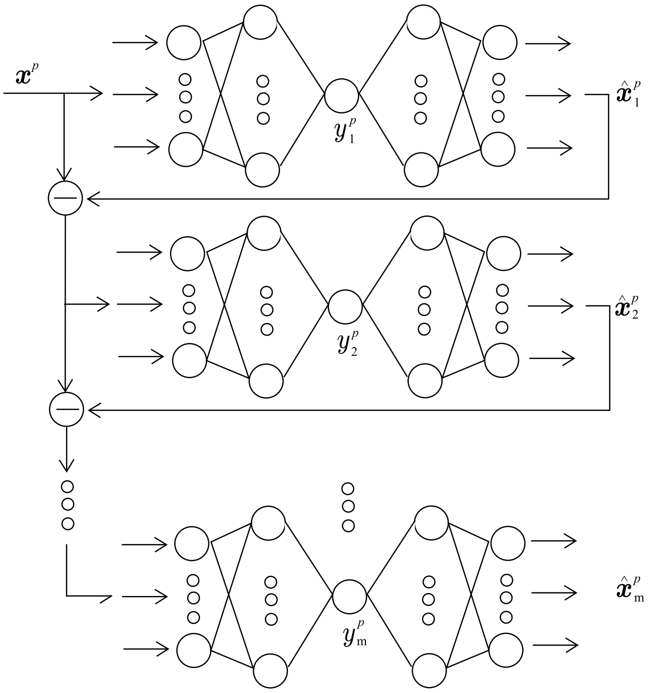
$\boldsymbol{x}^p$
y_1^p
$\hat{\boldsymbol{x}}_1^p$
y_2^p
$\hat{\boldsymbol{x}}_2^p$
y_m^p
$\hat{\boldsymbol{x}}_m^p$

Figure 2.3. The serial model of autoassociators by Kramer.

The architecture of the model is shown in Figure 2.5. This network corresponds to the decompression part of the autoassociator by Kramer. The model does not construct the mapping functions to compress the data into the principal components, while the components are obtained by training.

In test process, the training of the input is required to obtain a principal component vector of a test sample. In this process, the parameters of the neural network are fixed. Therefore, the computation cost for test samples is large.

The superposed training method proposed by Takahashi, et al. [23] enables an SMLP to extract the ordered principal components. The authors of the paper mathematically proved that, in a case where the training method is applied to a three layered linear perceptron, the outputs of the units in the bottleneck layer converge to the ordered principal components obtained by PCA.

This superposed training is considered to be effective and interesting, while the network architecture of the perceptron is not hierarchical. Therefore, when we add a new principal component, the re-training of the entire networks is required.

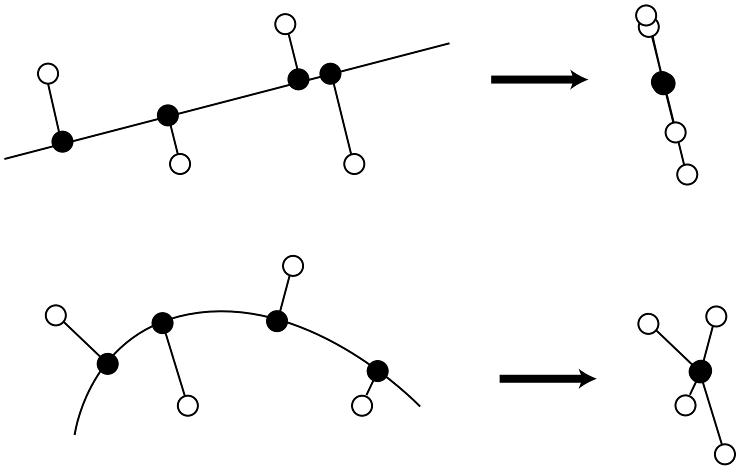

Figure 2.4. The residual error in a linear case (above) and the residual error in a nonlinear case (below).

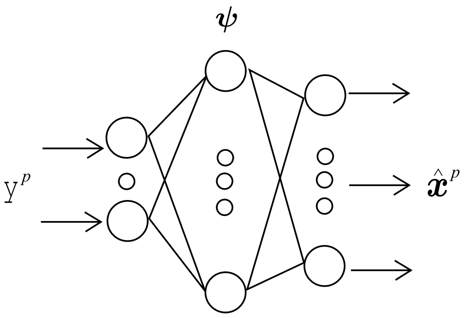

Figure 2.5. The input training model by Tan.

Chapter 3

A Proposed Method

In Chapter 3, a novel method of NLPCA is proposed. In the first section, the order of the principal components in PCA and nonlinear extension of PCA are discussed. In the next section, the implementation of the method using neural networks is described. The implementation makes possible the construction of nonlinear mapping functions and their iterative adjustment.

3.1 Nonlinear extension of PCA

In PCA, a reconstruction function is composed of a linear combination of the bases. The bases $\{\boldsymbol{e}_i\}_{i=1,\cdots,m}$ are obtained from the minimization of MSE on the data-reconstruction under the constraint that the bases are normalized $||e_i|| = 1$ for any i.

After the bases have been obtained, we can generate the extraction function from the bases as follows,

$$
\begin{aligned}
y_1 &= \boldsymbol{e}_1^T \cdot \boldsymbol{x}, && (3.1)\\
y_2 &= \boldsymbol{e}_2^T \cdot \boldsymbol{x}, && (3.2)\\
&\vdots \\
y_i &= \boldsymbol{e}_i^T \cdot \boldsymbol{x}, && (3.3)\\
&\vdots \\
y_m &= \boldsymbol{e}_m^T \cdot \boldsymbol{x}, && (3.4)
\end{aligned}
$$

and the reconstruction function as follows,

$$
\begin{aligned}
\hat{\boldsymbol{x}}_1 &= \boldsymbol{e}_1 y_1, && (3.5)\\
\hat{\boldsymbol{x}}_2 &= \boldsymbol{e}_1 y_1 + \boldsymbol{e}_2 y_2, && (3.6)\\
&\vdots \\
\hat{\boldsymbol{x}}_i &= \boldsymbol{e}_1 y_1 + \boldsymbol{e}_2 y_2 + \cdots + \boldsymbol{e}_i y_i, && (3.7)\\
&\vdots \\
\hat{\boldsymbol{x}}_m &= \boldsymbol{e}_1 y_1 + \boldsymbol{e}_2 y_2 + \cdots + \boldsymbol{e}_m y_m, && (3.8)
\end{aligned}
$$

where y_i represents the i-th principal component of the input vector, and $\hat{\boldsymbol{x}}_i$ is the n-dimensional vector reconstructed from $y_1, \cdots, y_i$.

The amount of the eigenvalue of a basis indicates the contribution of the basis for the data-representation. Consequently, if the bases are combined in the order of the contribution, the representation efficiency is maximized for the number of principal components employed. We can design the valance of tradeoff between the representational ability and the dimensionality of principal components, for a specific purpose such as the high fidelity of the data-reconstruction (high-dimensionality), and the low cost of memory (low-dimensionality), adjusting the number of the ordered bases to be combined.

However, in the neural network based models of NLPCA such as Kramer' model [13] and Tan's model [24], the principal components are not constructed in order of the contribution as the principal components of PCA.

In order to overcome this problem, let us introduce nonlinear extraction functions that independently map a vector in an input space onto a principal component space as follows,

$$\begin{aligned} y_1 &= \phi_1(\boldsymbol{x}), && (3.9)\\ y_2 &= \phi_2(\boldsymbol{x}), && (3.10)\\ &\vdots \\ y_i &= \phi_i(\boldsymbol{x}), && (3.11)\\ &\vdots \\ y_m &= \phi_m(\boldsymbol{x}), && (3.12) \end{aligned}$$

and *hierarchically-arranged* reconstruction functions that hierarchically map a vector in the principal component space onto the original input space as follows,

$$\begin{aligned} \hat{\boldsymbol{x}}_1 &= \boldsymbol{\psi}_1(y_1), && (3.13)\\ \hat{\boldsymbol{x}}_2 &= \boldsymbol{\psi}_2(y_1, y_2), && (3.14)\\ &\vdots \\ \hat{\boldsymbol{x}}_i &= \boldsymbol{\psi}_i(y_1, \cdots, y_i), && (3.15)\\ &\vdots \\ \hat{\boldsymbol{x}}_m &= \boldsymbol{\psi}_m(y_1, \cdots, y_m). && (3.16) \end{aligned}$$

The above definition is obtained by replacing the linear operator of a dot product: $\boldsymbol{e}_i^T \cdot ()$ with a nonlinear mapping function: $\phi_i()$ and replacing the linear operator of a vectorial sum: $\boldsymbol{e}_1() + \boldsymbol{e}_2() + \cdots + \boldsymbol{e}_i()$ with a nonlinear mapping function: $\boldsymbol{\psi}_i()$.

In the proposed method, the i-th extraction function ϕ_i and the i-th reconstruction function $\boldsymbol{\psi}_i$ are paired in the following manner:

$$\begin{aligned} \hat{\boldsymbol{x}}_1 &= \boldsymbol{\psi}_1(\phi_1(\boldsymbol{x})), && (3.17)\\ \hat{\boldsymbol{x}}_2 &= \boldsymbol{\psi}_2(y_1, \phi_2(\boldsymbol{x})), && (3.18)\\ &\vdots \\ \hat{\boldsymbol{x}}_i &= \boldsymbol{\psi}_i(y_1, \cdots, y_{i-1}, \phi_i(\boldsymbol{x})), && (3.19) \end{aligned}$$

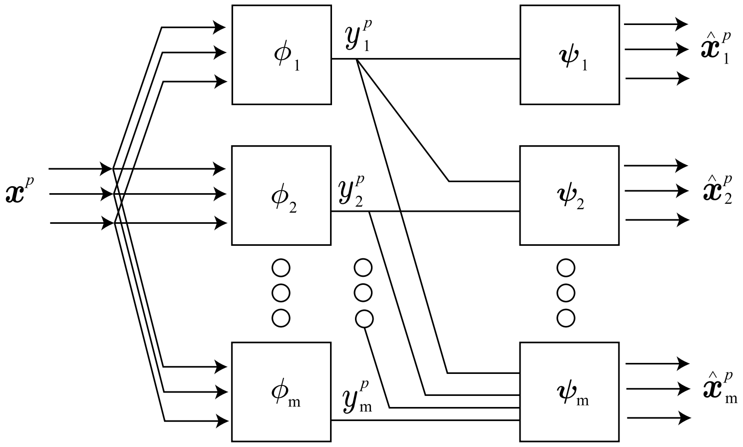

Figure 3.1. A diagram of the hierarchically-arranged mapping functions

$$\vdots$$

$$\hat{\boldsymbol{x}}_m = \boldsymbol{\psi}_m(y_1, \cdots, y_{m-1}, \phi_m(\boldsymbol{x})). \tag{3.20}$$

The hierarchically paired functions are shown in the diagram of Figure 3.1.

Each pair of the functions: $(\phi_i, \boldsymbol{\psi}_i)$ is adjusted in the increasing order of i to minimize the MSE. Corresponding to the number of i, ϕ_i and $\boldsymbol{\psi}_i$ are optimized with the i-th MSE criterion:

$$\text{MSE}_{\ i} = E[||\boldsymbol{x} - \hat{\boldsymbol{x}}_i(\boldsymbol{y}_i)||^2] \tag{3.21}$$

$$= E[||\boldsymbol{x} - \boldsymbol{\psi}_i(y_1, \cdots, y_{i-1}, \phi_i(\boldsymbol{x})||^2], \tag{3.22}$$

under the condition that $y_1, \cdots, y_{i-1}$ are given. Let us call the principal component of small i the higher component, and the principal component of large i the lower component. In the proposed method, y_i is regarded as the i-th significant nonlinear principal component for the following reason.

In the proposed method, the order of the principal components is defined as the order of them to be combined. The proposed method adjusts the parameters of the i-th nonlinear extraction function ϕ_i to perform the best mapping combined with the upper nonlinear extraction functions $\phi_1, \phi_2, \cdots, \phi_{i-1}$. Consequently, when we choose a nonlinear extraction function from $\phi_i, \phi_{i+1}, \cdots, \phi_m$ for the function combining with $\phi_1, \phi_2, \cdots, \phi_{i-1}$, the representational ability of the i-th function ϕ_i is the best, while the lower functions $\phi_{i+1}, \phi_{i+2}, \cdots, \phi_m$ are equal to or less significant than ϕ_i. The order of $\phi_1, \phi_2, \cdots, \phi_i, \cdots, \phi_m$ corresponds to the best order to combine the principal components for data-representation. Therefore, y_i can be regarded as the i-th significant nonlinear principal component.

3.2 The implementation using neural networks

We can implement the extraction functions $\{\phi_i\}_{i=1,\cdots,m}$ and reconstruction functions $\{\boldsymbol{\psi}_i\}_{i=1,\cdots,m}$ with several ways. In the proposed method, the author focused on the neural network based model to obtain the mapping functions.

The neural network model has some advantages for an implementation of a mapping function. The first advantage is the representational ability. We can approximate any mapping function in any precision with adjusting the number of hidden units in the network. The other advantage is the learning ability. We can adjust the mapping functions iteratively with a learning algorithm for the neural networks such as a back propagation algorithm.

The author proposes a hierarchical neural network to perform NLPCA. The proposed network is composed of a number of independent sub-networks that can extract the ordered nonlinear principal components. The number of sub-networks equals to the number of principal components extracted. The number of layers in each sub-network is five or larger than five, and the number of the input and the output units are equally set to the dimensionality of the input vector, while the number of units in the middle layer (extraction layer) corresponds to the incremental dimensionality of the principal components employed for the reconstruction in the sub-network. The structure of the model is shown in Figure 3.2.

The activation function of the units in the input, output, and extraction layers is a linear function,

$$f(u) = u, \tag{3.23}$$

while the activation function of the units in the other layer is a differentiable nonlinear function such as a hyperbolic tangent

$$f(u) = \tanh(\frac{u}{T}), \tag{3.24}$$

where T is a constant value on the nonlinearity and u is a weighted sum of the inputs to the units. We can apply the other functions to an activation function such as a sigmoid function,

$$f(u) = \frac{1}{1 + \exp(-\frac{u}{T})}, \tag{3.25}$$

and a radial basis function,

$$f(u) = \exp\left\{-\left(\frac{u}{T}\right)^2\right\}. \tag{3.26}$$

A threshold is employed in the nonlinear units of the second and the fourth layer in the proposed network.

Each sub-network is trained to reconstruct an input vector $\boldsymbol{x}$ in its output layer by producing $\hat{\boldsymbol{x}}$. In order to extract the first principal component in the first sub-network, we set one unit in its extraction layer. The output of this unit represents the first principal component of the input vector. The extraction layer of the second sub-network receives the first principal component calculated in the first sub-network and is independently trained to generate a function that maps the input vector into the second principal component. This action is hierarchically performed in all of the sub-networks. It is obvious that the proposed model performs the hierarchically nonlinear mapping described in the previous section.

When n-dimensional data $\boldsymbol{x}^p$ (numbered as the p-th sample) is given to the first layer of the i-th sub-network, the i-th unit in the extraction layer outputs the i-th principal component:

$$y_i^p = \phi_i(\boldsymbol{x}^p) \in R^1, \tag{3.27}$$

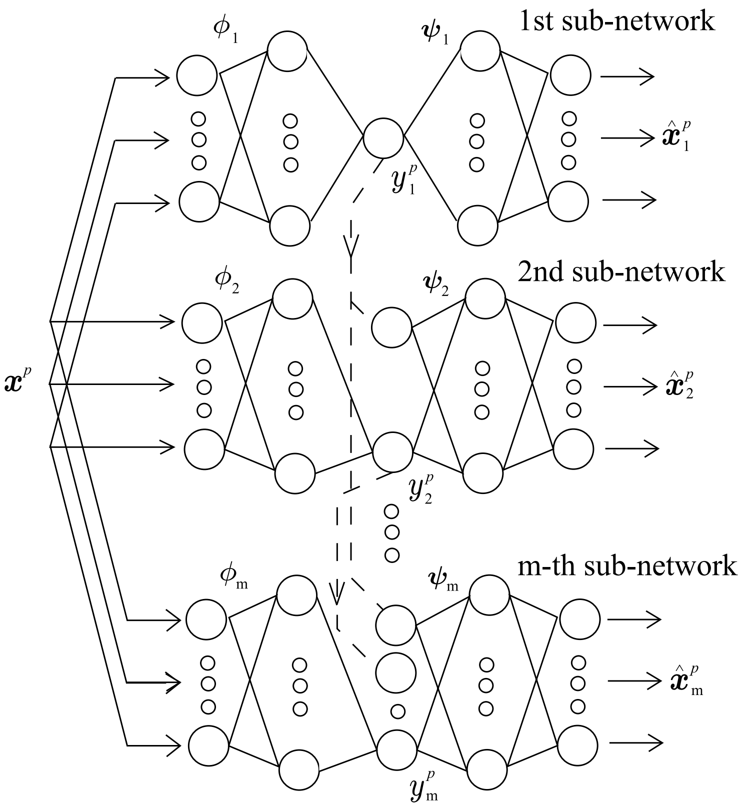

Figure 3.2. The proposed neural network. The network is composed of the number of m sub-networks. In the i-th sub-network, the left three layers from the middle layer, and the right three layers play the role of ϕ_i and $\boldsymbol{\psi}_i$, respectively. The i-th sub-network is given the values of principal components $y_1, \cdots, y_{i-1}$ from the higher $1, \cdots, (i-1)$-th sub-networks. The sub-networks are adjusted in the increasing order of i with a back propagation algorithm.

where the mapping function ϕ_i is the i-th extraction function from $\boldsymbol{x}^p$ onto a principal component y_i^p. In the i-th sub-network, ϕ_i corresponds to the extraction part of the network connected from all units in the input layer to the i-th unit of the extraction layer, while the principal components $y_1^p, y_2^p, \cdots, y_{i-1}^p$ are fed to the other units of this layer from all of the higher sub-networks.

The mapping function $\boldsymbol{\psi}_i$ is the i-th reconstruction function from the principal components $y_1^p, y_2^p, \cdots, y_i^p$ onto the reconstructed data $\hat{\boldsymbol{x}}_i^p$. $\boldsymbol{\psi}_i$ corresponds to the reconstruction part of the network connected from all units of the extraction layer to all units of the output layer.

The calculation of the outputs is carried out in the increasing order of the sub-network number i. Each connection weight of the i-th sub-network is then adjusted for each input vector: $\boldsymbol{x}^p$ in order to obtain an identical mapping with the criterion of the i-th MSE:

$$\text{MSE}\ {}_i^p = ||\boldsymbol{x}^p - \hat{\boldsymbol{x}}_i^p||^2. \tag{3.28}$$

In regard with the $\boldsymbol{x}^p$, the additional correction: Δw_i^p of a connection weight:

w_i^p is defined as

$$\Delta w_i^p = -\eta \frac{\partial E_i^p}{\partial w_i^p}, \tag{3.29}$$

where η is a constant training parameter. w_i^p of the sub-networks is iteratively corrected by a back propagation algorithm.

The w_i^p of the i-th sub-network is adjusted only inside the i-th sub-network. Error signals of the i-th sub-network are not propagated into the extraction part of all higher sub-networks. As a result, the adjustment of the connection weights in the i-th sub-network does not depend on the lower sub-networks but depends on the higher sub-networks. We can expect that this unidirectional dependency forces the extraction function to be different from any of the higher function and to cover the data-reconstruction of the higher function.

Chapter 4

Curvilinear Axis

In Chapter 4, two experimental results are presented. In the first experiment, three-dimensional artificial data sampled from a simple curved surface are utilized. The distribution of the data set is easy to understand. We examine the proposed method's construction of the curvilinear axes in the order of significance. In the second experiment, high dimensional data sampled from an artificial waveform are utilized. We examine the proposed method's extraction of the feature of the waveform with a small number of principal components.

4.1 Experiment with three-dimensional samples

A sample vector of a training set and a test set is the coordinate: (x_1, x_2, x_3) of a point on a elliptic paraboloid in R^3 given as follows:

$$x_3 = \frac{x_1^2}{a_1^2} + \frac{x_2^2}{a_2^2}, \tag{4.1}$$

where $a_1 = 1.0$, $a_2 = 3.0$ in this experiment.

The neural network used in this experiment has two sub-networks. Each sub-network has five layers, where the number of the units in the first and fifth layers of each sub-network is three corresponding to the dimensionality of the coordinate. The number in the second and fourth layers is ten. The parameters for the network are set to $\eta = 0.05$ and $T = 0.1$. The number of training samples is 20,000, which are randomly chosen from the elliptic paraboloid. The number of the test samples is 400, which randomly chosen from lattice points on the surface as shown in Figure 4.1.

Figure 4.2 shows the first principal component axis which is generated in the first sub-network. The plot of the first principal component axis in the figure is obtained from the sample reconstruction with varying a value of the first principal component in the first sub-network. Figure 4.3 shows the second principal component axis which is generated in the second sub-network. The plot of the second axis is obtained from the sample reconstruction with varying a value of the second principal component in the second sub-network. The input into the unit of the first principal component in the second sub-network is fixed to zero in this reconstruction.

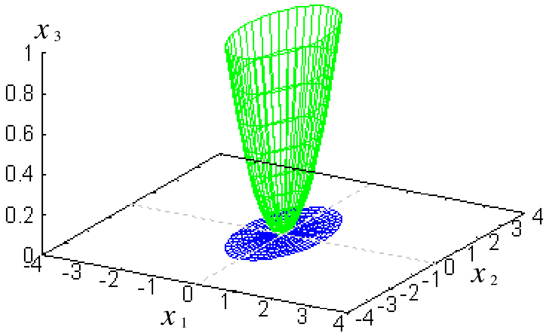

Figure 4.1. Test samples. The samples are the lattice points on the elliptic paraboloid

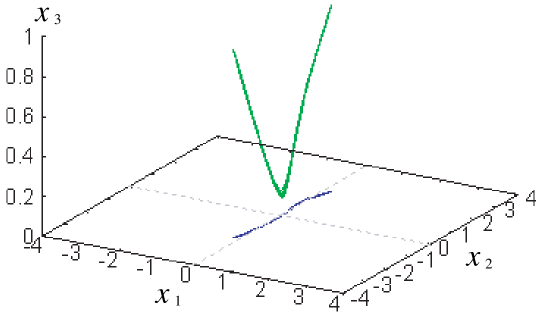

Figure 4.2. The plot of the first principal component axis.

From Equation (4.1), when $a_1 < a_2$, the distribution of the samples has the greatest variance along

$$x_3 = \frac{x_2^2}{a_2^2}, \tag{4.2}$$

which corresponds to Figure 4.2 and the second greatest variance is along

$$x_3 = \frac{x_2^2}{a_1^2}, \tag{4.3}$$

which is finely reflected in Figure 4.3.

Figure 4.4 shows the samples reconstructed by the second sub-network with the first and second principal components. It is obvious from this experiment that the proposed model is able to extract the nonlinear principal components axis from the nonlinearly distributed samples.

Figure 4.5 shows the principal component scores of the inputs. The horizontal axis indicates the first principal component score, while the vertical axis indicates the second principal component score.

Next, the author demonstrated the experiment to construct three principal components with three sub-networks in the proposed network. The condition of the experiment is the same as the above experiment except for the number

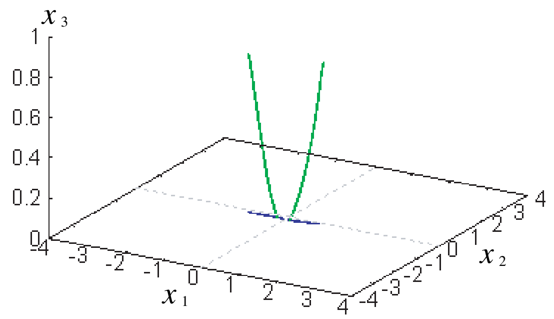

Figure 4.3. The plot of the second principal component axis.

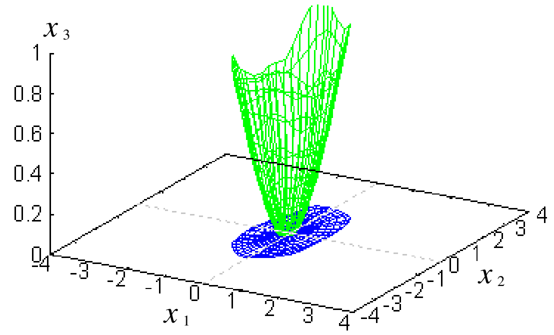

Figure 4.4. The samples reconstructed with the first and second principal components.

of sub-networks. Table 4.1 shows the MSE of each sub-network for 1,000 test samples.

As shown in Table 4.1, the MSE of the first, second, and third sub-network were 0.18319, 0.00473, and 0.00257, respectively. It is considered that the reconstruction error is smaller when the number of the available principal components is larger.

When the number of a principal component increased from one to two, the MSE decreased by 0.17846, but when the number of principal components increased from two to three, the MSE decreased by the small value of 0.00216. Since the MSE is not much improved by the addition of the third principal component, we can consider that it is sufficient to represent a sample in this experiment with two principal components, and the higher principal components have a higher ability to represent the sample than the lower components.

Additionally, the author demonstrated the experiments with the samples of the above-described elliptic paraboloid embedded in a four, five, and six-dimensional space. In these cases of the embedded samples in a high-dimensional space, the proposed network could almost reconstruct the input sample with using two principal components as well as the case of the samples in a three-dimensional space.

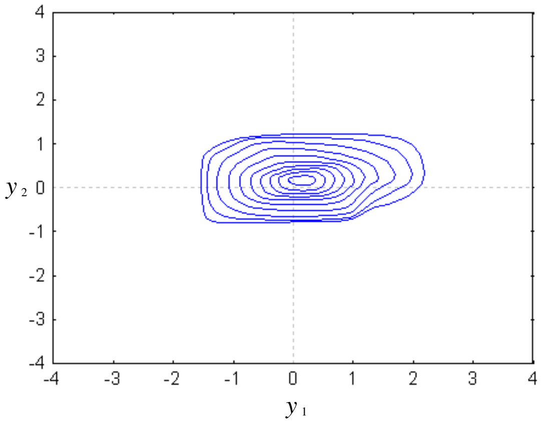

Figure 4.5. A distribution of the first and second principal component scores.

Table 4.1. The MSE of each sub-network

No. of sub-network	MSE	The decrease of MSE
1st	0.18319	-
2nd	0.00473	0.17846
3rd	0.00257	0.00216

4.2 Experiment with the samples of a waveform

In this experiment, the component of an input vector is sampled at constant intervals from a function which is a superposition of three sinusoidal functions. The input vector is formulated as follows:

$$(x_1, x_2, \ldots, x_n) = (f(\tau + \theta), f(2\tau + \theta), \ldots, f(n\tau + \theta)), \tag{4.4}$$

and the function employed is following,

$$f(\alpha; \boldsymbol{a}, \boldsymbol{\omega}, K) = \sum_{k=1}^{K} a_k \cdot \sin(\omega_k \alpha), \tag{4.5}$$

where, in this experiment, the dimensionality $n = 100$, the number of the superposition $K = 3$, the angular frequencies: $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3) = (1.0, 2.0, 3.0)$, the amplitudes: $\boldsymbol{a} = (a_1, a_2, a_3) = (0.5, 0.3, 0.2)$, and $\tau = \frac{2\pi}{n}$. The initial phases θ of the training samples are chosen at random.

The number of training samples is 50,000. The number of test samples is 100. The number of sub-networks is five and the number of layers in each sub-network is seven through out this experiment. The representational ability of the network is enhanced by assigning two layers in the hidden layer of the extraction part and the reconstruction part in each sub-network. The number of units in the first and seventh layers is 100, and the number of units in the second, third, fifth and sixth layers is 200. The parameters are set to $\eta = 0.001$ and $T = 0.1$.

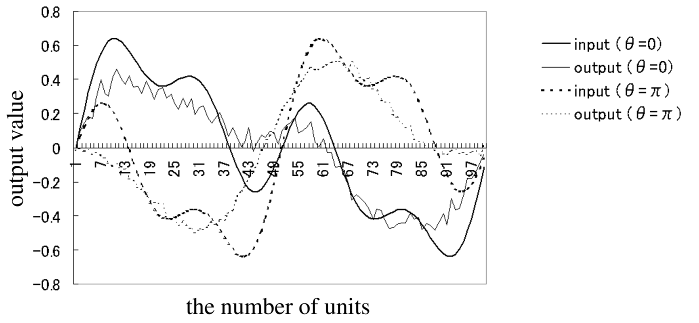

Figure 4.6. Reconstructed waveforms by the first sub-network.

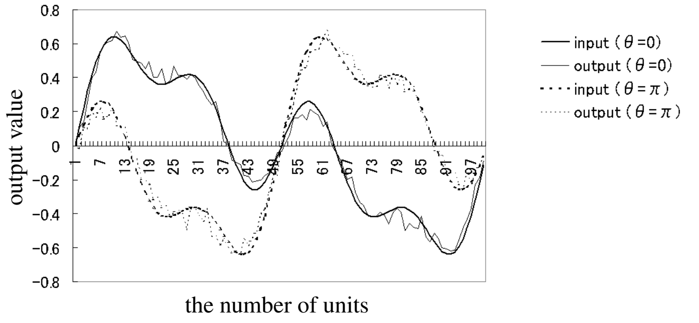

Figure 4.7. Reconstructed waveforms by the fifth sub-network.

Figure 4.6 and Figure 4.7 shows the reconstructed waveforms with the initial phases of $\theta = 0$ and π. The waveforms were reconstructed from the test samples by the trained network.

It is clear from Figure 4.6 and Figure 4.7 that the waveform reconstructed by the fifth sub-network which uses five principal components is better than the waveform reconstructed by the first sub-network which uses only one principal component.

In Figure 4.6, the reconstructed waveform of the first sub-network is similar to a sinusoidal wave. The first extraction function seems to obtain the lowest-frequency sinusoidal function in the superposed function: f. The reason will be that the lowest-frequency sinusoidal function has the largest amplitude which most contributes to the superposition. The first extraction function works most efficiently to reconstruct the objective waveform if the first extraction function obtains the lowest-frequency sinusoidal function.

Figure 4.8 shows a variation of the principal components in regard with the initial phases. The variation indicates that the principal components represent the phase of the wave. In the figure, the first principal component works as a low-frequency component, and the frequency gradually grows larger in the subsequent principal components.

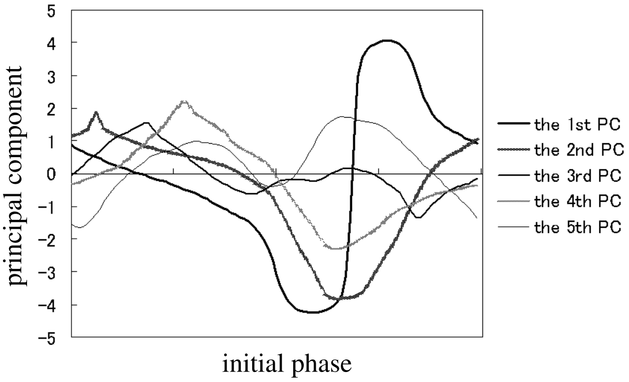

Figure 4.8. The variation of the principal components in regard with the initial phases.

This result is comparable to the Fourier series expansion. Since the proposed model can construct new basis functions for the given waveforms, the proposed method can be expected to construct a more efficient representation for the signals that cannot be expressed efficiently by the Fourier series expansion, i.e., a signal with the discontinuity.

4.3 Summary

In order to examine the nonlinearity of the proposed method and its efficiency, two experiments are demonstrated.

In the first experiment we examined the proposed method's construction of curvilinear axes in the order of the significance, utilizing three-dimensional artificial samples which distribute on a surface. Since the axes were fitted nonlinearly to the distribution of the samples, the distribution was represented with fewer principal components than PCA.

Moreover, we examined the proposed method's extraction of the features of a waveform with a small number of principal components. The experimental results suggest that the proposed method will provide more effective basis functions than the Fourier series expansion by adjusting the mapping functions for an objective data set.

Chapter 5

Representational Ability

In Chapter 5, the representational ability of the mapping functions is discussed. In the proposed method, each mapping function is implemented by a three-layered perceptron. In the first section, the representational ability of the mapping functions is described based on the degree of freedom of the network. In the next section, experiments are demonstrated to examine the representational ability. In the experiments, the representational ability is shown with variations in the degree of freedom, utilizing the three-dimensional artificial data used in the preceding chapter.

5.1 A mapping function of a three-layered perceptron

The proposed method provides the framework to construct any hierarchical mapping functions to extract the principal components and to reconstruct the data vectors. The set of the feasible mapping functions by neural networks is determined by the structure of the networks.

A mapping function $\boldsymbol{F}$ by the three layered perceptron is mathematically formulated as follows,

$$\begin{aligned} \boldsymbol{F} &= (F_1, \cdots, F_l, \cdots, F_L), & (5.1) \\ F_l\Big(\boldsymbol{x}; \overline{W}^{L(M+1)}, \underline{W}^{M(N+1)}\Big) &= \sum_{k=1}^{M} \overline{w}_{lk} f\Big(\sum_{j=1}^{N} \underline{w}_{kj} x_j + \underline{w}_{k0}\Big) + \overline{w}_{l0} & (5.2) \end{aligned}$$

where the perceptron has a nonlinear activation function in the hidden layer and a linear function in the output layer. F_l is the l-th component of $\boldsymbol{F}$. $\overline{W}^{L(M+1)}$ and $\underline{W}^{M(N+1)}$ indicates a $L \times (M+1)$ matrix and a $M \times (N+1)$ matrix. They are corresponding to the weights of connections which connect the hidden layer and the output layer, and the input layer and the hidden layer, respectively. L, M, N are the number of units in the output layer, the hidden layer and the input layer, respectively. f is a differentiable function. In this study, f is a hyperbolic tangent as is referred to Equation (3.24). Note that M is the number of the superposition of the nonlinear functions $\boldsymbol{f}$.

In order to focus on the size of the matrices, here, let $\boldsymbol{F}$ rename

$$F_l(\boldsymbol{x}; L, M, N) = F_l(\boldsymbol{x}; \overline{W}^{L(M+1)}, \underline{W}^{M(N+1)}). \tag{5.3}$$

We can represent the mapping functions $(\phi_i, \boldsymbol{\psi}_i)_{i=1,\cdots,m}$ of the proposed method as

$$\phi_i(\boldsymbol{x}) = F(\boldsymbol{x}; 1, \alpha, n), \tag{5.4}$$

and

$$\boldsymbol{\psi}_i(\boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{x}; n, \beta, i), \tag{5.5}$$

where α and β are the number of each hidden layer, and n, m are the dimensionality of the input vector and the principal components defined in the preceding chapter.

Since the α and β are the number of the superposition of the nonlinear function f, the parameters are considered to indicate the representational ability. When the values of α and β increase, the mapping functions can represent the more complex functions, namely, the representational ability of the functions increases.

5.2 Experiment on the representational ability

The author demonstrated the experiment on the representational ability of the mapping functions. In the experiment, the same data as the three-dimensional artificial data in the preceding chapter were utilized, and the compression and the decompression of the data were performed with two principal components.

Figure 5.1 shows the MSE of the proposed method in regard with a different number of units in the second and the fourth layer. The numbers of the units in the second and the fourth layer are corresponding to α and β, respectively. In this experiment, $\alpha = \beta$. The figure shows that sub-networks with a large number of units in the second and the fourth layers have better representational ability.

Moreover, the author examined the performance of the proposed method by setting different number of the units for the second or the fourth layer. In Figure 5.2, the number of units in the fourth layer is fixed at $\beta = 15$, while the number of the units in the second layer α is varied. On the other hand, in Figure 5.3, the number of the units in the second layer is fixed at $\alpha = 15$, while the number of units in the fourth layer β is varied. As shown in Figure 5.1, 15 units in the second and the fourth layer is enough to represent the sample vector.

From Figure 5.2 and Figure 5.3, we can draw the conclusion that if one of the representational ability of the extraction function or the reconstruction function is low, the total representation is not better.

As shown in Figure 5.4, in the input data space, the extraction function gives the counter line on which points are projected onto a same score of principal components, and the reconstruction function gives a manifold to summarize the distribution of the data. The total ability to represent the distribution of the data will depend on the relation of the extraction and the reconstruction function.

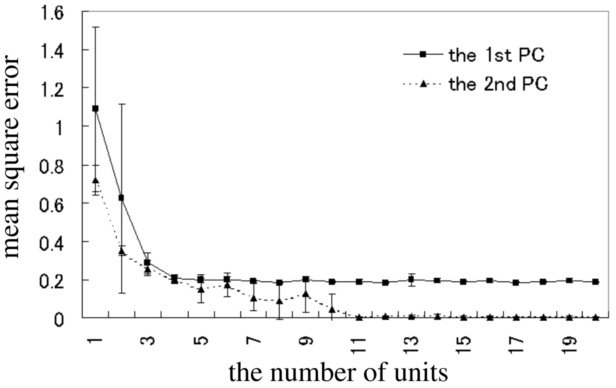

Figure 5.1. The mean square error with various numbers of the units in the second (α) and fourth layers (β). The numbers are set to $\alpha = \beta$.

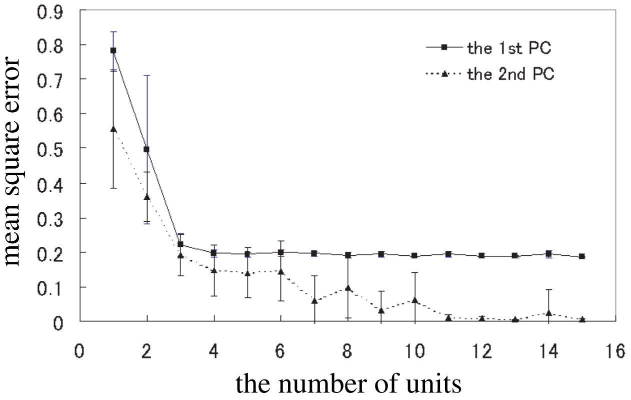

Figure 5.2. The mean square error with various number of the units in the second layer (α) with fixing the number of the units in the fourth layer at $\beta = 15$.

Although the proposed method does not have an explicit parameter that corresponds to the contribution rate as in the conventional PCA, the MSE of each sub-network can be loosely applicable to measure the representational ability of the sub-network.

5.3 Summary

In this chapter, the representational ability of the mapping function was examined. In the beginning, the mapping functions implemented by the neural networks were mathematically formulated. The representational ability depends on the number of units in the hidden layer. In the next section, the data were reconstructed with variations in the number of units. The experimental results showed that even if only one of the representational ability of the extraction function or the representational ability of the reconstruction function is low,

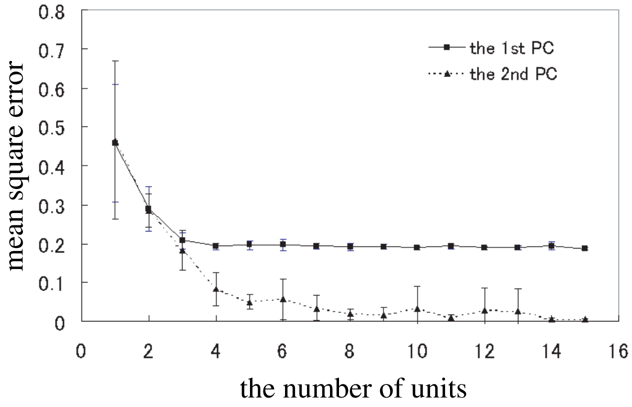

Figure 5.3. The mean square error with various numbers of the units in the fourth layer (β) with fixing the number in the second layer at $\alpha = 15$.

the total representation is not better. The combination of the extraction and the reconstruction function determines the total performance.

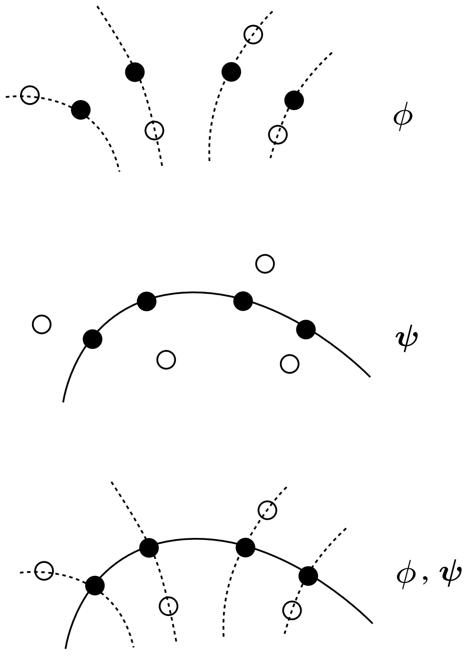

Figure 5.4. The counter line by an extraction function: ϕ and the manifold by a reconstruction function: $\boldsymbol{\psi}$. The white and black circles indicate the input vectors and the reconstructed vectors, respectively.

Chapter 6

The Distortion of the Distance

In Chapter 6, the distortion of the distribution is discussed with a comparison between PCA and the proposed method. In the first section, an experiment on data-reconstruction is demonstrated utilizing a set of iris data sampled from an open database. In the second section, we measure the distance of the input vectors and the distance of the reconstructed vectors. Then, we examine the distortion of these distances.

6.1 Experiments on data-reconstruction

In this section, data-reconstruction based on MSE is discussed, utilizing a set of Fisher's iris data which is sampled from the open database [1].

Table 6.1 shows the parameters of iris data used in this experiment. The data set has four attributes, which are corresponding to the length and the breadth sizes of a sepal and a petal. PCA and the proposed method are performed with the data set. The parameters of the proposed method used in the experiment are shown in Table 6.2.

Figure 6.1 shows the MSE by PCA and the proposed method. In this experiment, the number of principal components was varied from one to four. The principal components are applied to reconstruct the data $\boldsymbol{x} \in R^4$ in order from the first to the fourth component.

As shown in Figure 6.1, the MSE of the proposed method are smaller than the MSE of PCA in the region from one to three of the horizontal axis. In the proposed method, the data are roughly reconstructed from two principal components, while PCA cannot give the same performance even if three principal components are used. With respect to the MSE with four principal components, however, the MSE of PCA is almost zero, while the MSE of the proposed method has a little residual error. The MSE of the proposed method may decrease after sufficient training, but it is not guaranteed that the MSE of the proposed method converges to zero.

In PCA, data vectors are completely reconstructed from m principal components, if m equals to the dimensionality of original data, because PCA constructs a coordinate system of the principal components by rotating and shifting

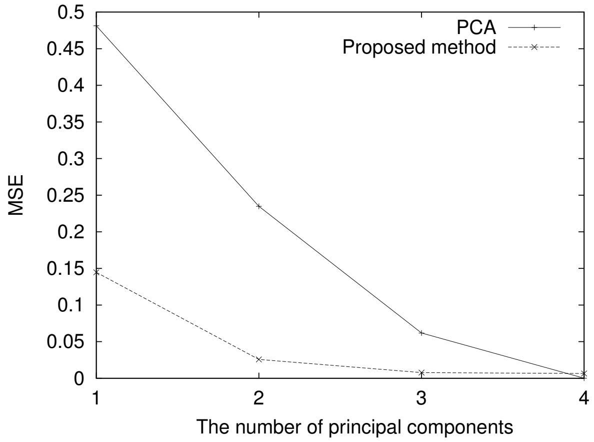

Figure 6.1. The MSE by PCA and the proposed method. A horizontal axis indicates the number of principal components. Principal components are applied to reconstruct data in order from the first to the fourth component.

Table 6.1. The parameters of an objective data set.

Parameter	Value
No. of samples	50
No. of attributes	4
Category of iris	Iris Setosa

an original coordinate system, essentially. On the other hand, in the proposed method, the perfect reconstruction is not guaranteed.

This point may be a drawback of the proposed method. However, when we practically use the proposed method for the purpose of the dimensionality reduction, the imperfection on the data-reconstruction will be not a critical problem, if the number of the principal components is set to be smaller than the dimensionality of the original data for the dimensionality reduction.

6.2 Experiments on the distortion of the distance

In this section, we discuss the distortion of the input vectors and the reconstructed vectors by measuring each distortion.

Let us define a distance between the p-th input vector $\boldsymbol{x}^p$ and the q-th input vector $\boldsymbol{x}^q$ as

$$d(p, q) = ||\boldsymbol{x}^p - \boldsymbol{x}^q||, \tag{6.1}$$

and the corresponding distance of the reconstructed vectors as

$$\hat{d}(p, q) = ||\hat{\boldsymbol{x}}^p - \hat{\boldsymbol{x}}^q||, \tag{6.2}$$

Table 6.2. The parameters of the proposed method.

Parameter	Value
no. of training	6000
no. of hidden units	15
Learning rate	0.005
Parameter T	0.5
Initial w	random number over [-0.03, 0.03]

where $\hat{\boldsymbol{x}}^p$ and $\hat{\boldsymbol{x}}^q$ are reconstructed from the principal components of $\boldsymbol{x}^p$ and $\boldsymbol{x}^q$, respectively. $||\cdot||$ represents L_2 norm as follows,

$$||\boldsymbol{u}-\boldsymbol{v}|| = \sqrt{\sum_{i=0}^{n}(u_i - v_i)^2}\ , \tag{6.3}$$

where $\boldsymbol{u}$ and $\boldsymbol{v}$ are the n-dimensional vectors.

When $\hat{d}(p,q)$ equals to $d(p,q)$ for any pair of p and q, the nonlinear mapping functions $\{\phi_i\}_{i=1,\cdots,m}$ and $\{\boldsymbol{\psi}_i\}_{i=1,\cdots,m}$ are considered to reconstruct the set of data, preserving the distribution of the set. If not equals, the distribution of the set is distorted by the mapping functions.

In order to examine the distortion of the distribution, a pair of vectors: ($\boldsymbol{x}^p$, $\boldsymbol{x}^q$) was picked up from the iris data set, and a corresponding pair of vectors: ($\hat{\boldsymbol{x}}^p$, $\hat{\boldsymbol{x}}^q$) reconstructed in the previous experiment was also picked up. Then, the distance $d(p,q)$ as well as the distance $\hat{d}(p,q)$ was calculated. This calculation was demonstrated for every combination of p and q.

Figure 6.2 shows the plots of the distance $(d,\hat{d})$. d and $\hat{d}$ are respectively the distance of input vectors and the distance of the reconstructed vectors calculated from the first principal component by PCA. In a similar fashion, Figure 6.3, Figure 6.4 and Figure 6.5 show the plots of the distance where $\hat{d}$ is calculated from the first and the second principal component by PCA, from the first principal component by the proposed method, and from the first and the second principal component by the proposed method, respectively. In these figures, the horizontal axis indicates d, while the vertical axis indicates $\hat{d}$. Additionally, an identical line $\hat{d} = d$ is also described in these figures.

Figure 6.2 and Figure 6.3 show that all plots $(d,\hat{d})$ by PCA distribute in the region of

$$\hat{d} \leqq d. \tag{6.4}$$

Since, PCA projects any reconstructed vector orthogonally onto a straight line or a hyper-plane as shown in Figure 6.6, Inequality (6.4) holds true.

Comparing with Figure 6.2 and Figure 6.3, we can see that the plots in Figure 6.3 are distributed more closely to the identical line $\hat{d} = d$ than in Figure 6.2. When we use more number of principal components, the reconstructed vectors preserve the distribution more accurately.

On the other hand, we notice that the proposed method has no constrain such as Inequality (6.4). Figure 6.4 and Figure 6.5 show that the plots are

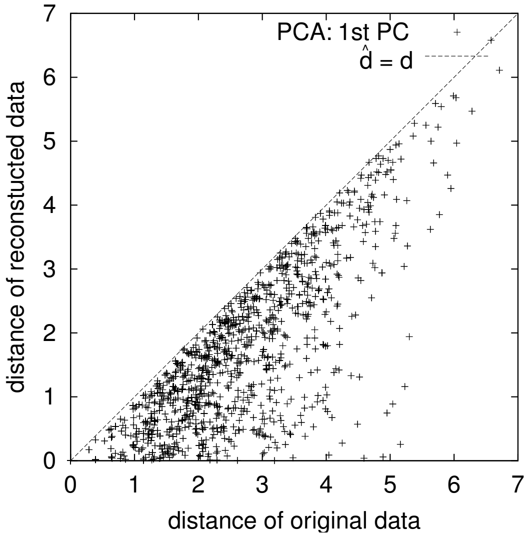

Figure 6.2. The plot of the distance by PCA using the first principal component. The horizontal axis indicates the distance of a pair of the input vectors. The vertical axis indicates the distance of the corresponding pair of the reconstructed vectors from the first principal component.

distributed in above and below of the identical line $\hat{d} = d$, and the variances from the line are small.

Comparing with Figure 6.2 and Figure 6.4, we can see that when we approximate vectors in R^4 with one principal component, the approximation of the vectors is more accurately than that of PCA. This suggests that the most of vectors may not distribute along a straight line, but distribute along a curved line.

We can consider that when the input vectors are nonlinearly mapped onto a curved subspace that represents the distribution of the vectors, the mapping function gives a new coordinate system to absorb a nonlinearity of the distribution of the input vectors.

6.3 Summary

In this chapter, the experiments with Fisher's iris data were demonstrated to reconstruct the data from their principal components of which dimensionality was reduced. A performance of the proposed method was better than that of PCA. We can consider that the high fidelity of the proposed method is caused by high degree of freedom in mapping functions.

PCA can give a perfect reconstruction with the same dimensionality of the principal components as the input vector, because PCA works as an orthogonal transformation of the coordinate system. The proposed method does not guarantee the perfect reconstruction. However, all principal components are not need in the dimensionality reduction. We do not have problems when the input vectors are represented with few principal components.

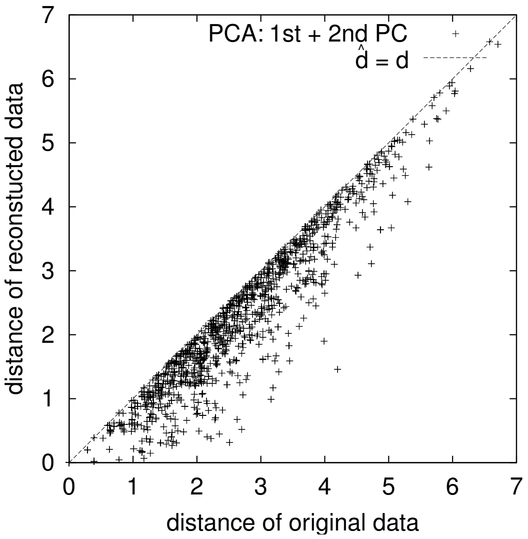

Figure 6.3. The plot of the distance by PCA using the first and second principal component. The horizontal axis indicates the distance of a pair of the input vectors. The vertical axis indicates the distance of the corresponding pair of the reconstructed vectors from the first and the second principal components.

Moreover, the distortion of the reconstruction was examined comparing the distance between the original data and the distance between the corresponding reconstructed data. The experiment was demonstrated with PCA and the proposed method. Firstly, a pair of data vectors was chosen from the original data set, and the corresponding pair of reconstructed vectors was calculated. Next, the distance between the vectors in each pair was calculated. When these distances were compared, it was found that the data reconstructed by the proposed method preserved the structure of the distance more accurately than the data by PCA did. The result is considered to suggest that objective data may distribute nonlinearly.

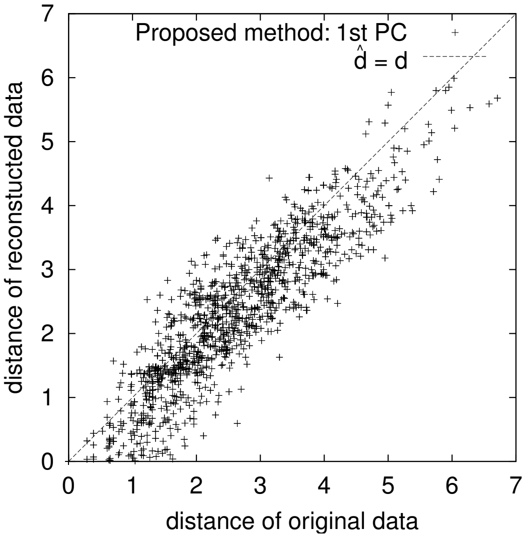

Figure 6.4. The plot of the distance by the proposed method using the first principal component. The horizontal axis indicates the distance of a pair of the input vectors. The vertical axis indicates the distance of the corresponding pair of the reconstructed vectors from the first principal component.

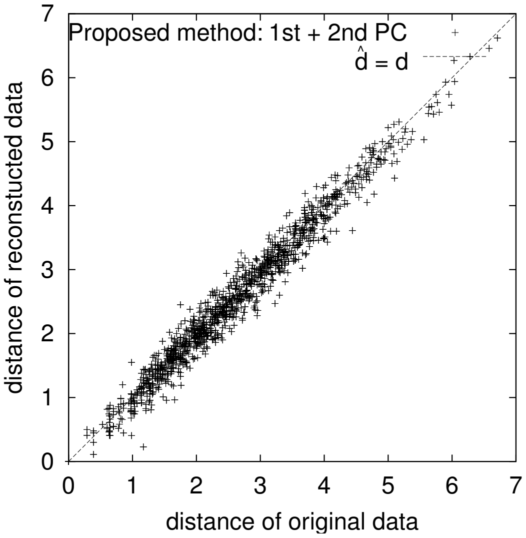

Figure 6.5. The plot of the distance by the proposed method using the first principal component. The horizontal axis indicates the distance of a pair of the input vectors. The vertical axis indicates the distance of the corresponding pair of the reconstructed vectors from the first and the second principal components.

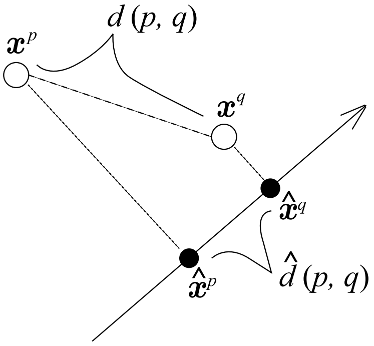

Figure 6.6. Orthogonal projections of data by PCA. In PCA, the inequality $\hat{d} \leqq d$ holds true.

Chapter 7

Applications for Practical Problems

In Chapter 7, the effectiveness of the proposed method for practical problems is examined with several open databases. Two experiments on dimensionality reduction are demonstrated. One is a data compression of facial images. The other is a feature extraction for the classification of handwritten numerals.

7.1 Dimensionality reduction of facial images

PCA has been one of the conventional methods to extract features of image data. Eigenface by Turk extracts features of facial images for recognition [26]. The subspace method extracts a subspace of images for each category and classifies an input image into the category to which subspace has the greatest similarity.

However, it is reported that nonlinear characteristics exist in some data sets of images, such as facial images with emotional expressions and images of an object with variable orientation [17]. When we reduce the dimensionality of such data, a nonlinear method is considered to perform more effectively than a linear method such as PCA.

We carried out a dimensionality reduction experiment with facial images sampled from the UMIST Face Database [6].

In this experiment, 750 training images and 250 test images of 20 peoples were utilized. The resolution of the image was 16 × 16 pixels in 256 gray scales.

The parameters of the PCA and the proposed method were adjusted with training samples. Next, the MSE of these methods with training and test samples was calculated, respectively. In the experiment, the dimensionality was reduced from 256 into 10, and the values shown in Table 7.1 were employed for the parameters of the proposed method.

Figure 7.1 shows the MSE with training and test samples by PCA and the proposed method. The horizontal axis indicates the number of principal components in the reconstruction, and the vertical axis indicates the MSE per pixel. In the experiment, the principal components were applied to reconstruct the image in order from the higher to the lower components.

As shown in Figure 7.1, when the number of principal components increases, the MSE of the proposed method on training samples decreases more rapidly

Table 7.1. Parameters of the proposed method.

no. of principal components	10
learning rate	0.001
parameter T	1.0
no. of hidden units	200
no. of training iterations	30,000
initial w	random over [-0.03, 0.03]

than that of the PCA. The MSE of the proposed method on test samples is also superior to that of the PCA, which shows the proposed method does not over-train. This result represents the effectiveness of the proposed method in dimensionality reduction.

Figure 7.2 shows the images which are extracted and reconstructed from the training samples and the test samples by PCA and the proposed method. The blocks of the images from top to bottom correspond to the results on the training samples by PCA, the results on the training samples by the proposed method, the results on the test samples by PCA, and the results on the test samples by the proposed method. The images in row (No) are the results of the (No)-th target image. The images in column (T) are the target images. The images in column (1), (3), $\cdots$, and (9) are reconstructed with one, three, $\cdots$, and nine principal components, respectively.

As shown in Figure 7.2, the images reconstructed by the proposed method are high in fidelity with a small number of principal components. The advantage of the proposed method over PCA is considered to come from the high representation ability of nonlinear mapping functions.

The set of facial images in this experiment are sampled from peoples whose sex and race are different. In this case, it is reported that the distribution of the data has nonlinearity [20]. In this experiment, nonlinearity of the proposed method is considered to contribute to the effective extraction and reconstruction of facial images.

7.2 Feature extraction of hand-written numerals

In this section, we discuss the dimensionality reduction as a feature extraction for the recognition. The author demonstrated experiments on dimensionality reduction of hand-written numerals with PCA and the proposed method.

In character recognition, dimensionality reduction is important to reduce the computational cost and to prevent the so-called curse of dimensionality, which means that the number of samples required for the estimation increases in the order of the dimensionality of the feature space.

Conventionally, linear methods such as PCA and Discriminant Analysis (DA) are mainly used in dimensionality reduction of character recognition. However, the proposed method may be more effective for such recognition, because its nonlinear mapping functions can represent the samples in high fidelity. Moreover, since the proposed method preserves the order of principal components,

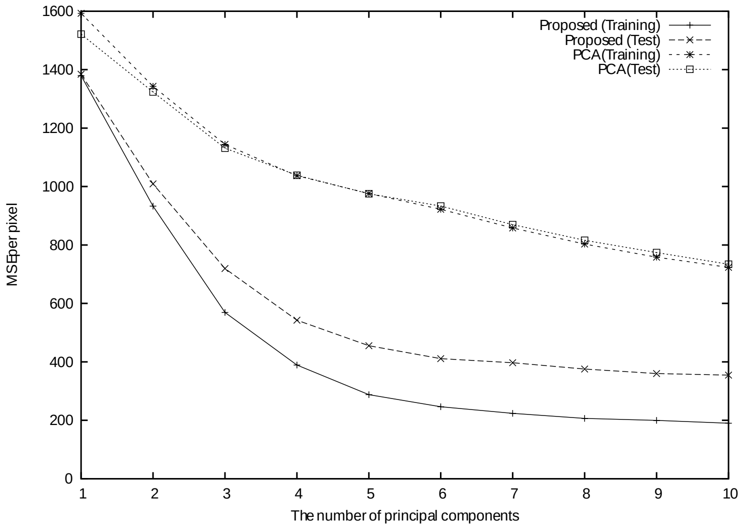

Figure 7.1. The MSE of Principal Component Analysis (PCA) and the proposed method in reconstructing the training and test samples. The horizontal axis indicates the number of principal components employed in the reconstruction, and the vertical axis indicates the MSE per pixel. The principal components were applied to reconstruct the images in the order from the higher to the lower components.

we can easily select the dimensionality number of the feature vectors.

In this experiment, an image database of hand-written numerals, IPTP-CDROM1 is utilized. The database has 10 categories which correspond to the numerals from zero, one, $\cdots$, to nine. Figure 7.3 shows sample images of the category "zero". Any image in the database is represented in the binary scale. 20,000 training samples and 4,950 test samples were randomly chosen for each category, respectively.

In a preprocessing, the original images were transformed into gray-scale images in low resolution. In this reduction, the original image is divided into 8×8 square regions. The author defined the gray-scale value of a region as the number of zero-brightness-value pixels included in the region. A zero-brightness-value pixel corresponds to a pixel whose color is black. After the set of the 64-dimensional gray-scale images was obtained, the images were normalized in order to set the average and the standard deviation to be zero and one, respectively.

After the preprocessing, the parameters of the PCA and the proposed method were adjusted with the training samples of all categories, and the principal components of the training samples were calculated. The author defined the average vector of the principal components of a category as the template vector of the category.

Then, the principal components of the test samples were calculated, and the principal component vectors were classified with the template vectors subject to the Nearest Neighbor Rule (NNR). NNR classifies a vector into a category to which the nearest template belongs.

In the experiment, the parameters shown in Table 7.2 were employed.

Figure 7.4 shows the recognition rate with PCA and with the proposed

Table 7.2. Parameters of the proposed method.

no. of principal components	10
learning rate	0.0005
parameter T	1.0
no. of hidden units	20
no. of training iterations	100,000
initial w	random over [-0.03, 0.03]

method. In Figure 7.4, the horizontal axis indicates the number of the principal components employed for recognition.

The figure shows the effectiveness of the proposed method in pattern recognition. When the number of principal components increases, the recognition rate increases in the both methods.

In this experiment, the recognition is performed for a pattern compressed into low-dimensional space by PCA and by the proposed method. In general, the compression and recognition are based on different assumptions so that the category of a pattern is taken into account in recognition, but is not in compression.

For example, PCA and DA construct different axes for data-compression and classification, as shown in Figure 7.5.

Therefore, the performance of a compressor cannot be evaluated only by the performance of its recognition rate. The recognition rate is affected by the relationship between the compressor and the classifier, which in this experiment is the NNR and templates.

7.3 Summary

In this chapter, the practicality of the propose method was examined. In the experiment, the proposed method was applied for the data compression with facial images and the feature extraction for the classification with hand-written numerals.

The experimental results showed that the proposed method could reconstruct the facial images high in fidelity, and the proposed method performed as a good feature extractor of the hand-written numerals for the recognition.

The proposed method do not only extracts the characteristic of the distribution of the data, but also provides the reconstruction functions for every set of the principal components in the order of significance. Therefore, the number of the principal components employed is not required to determine in advance. We can select the number of principal components when we use the mapping functions. Since, in the recognition, we often require the trial and error to determine the dimensionality of the feature vector, the proposed method can be an efficient feature extractor for the recognition.

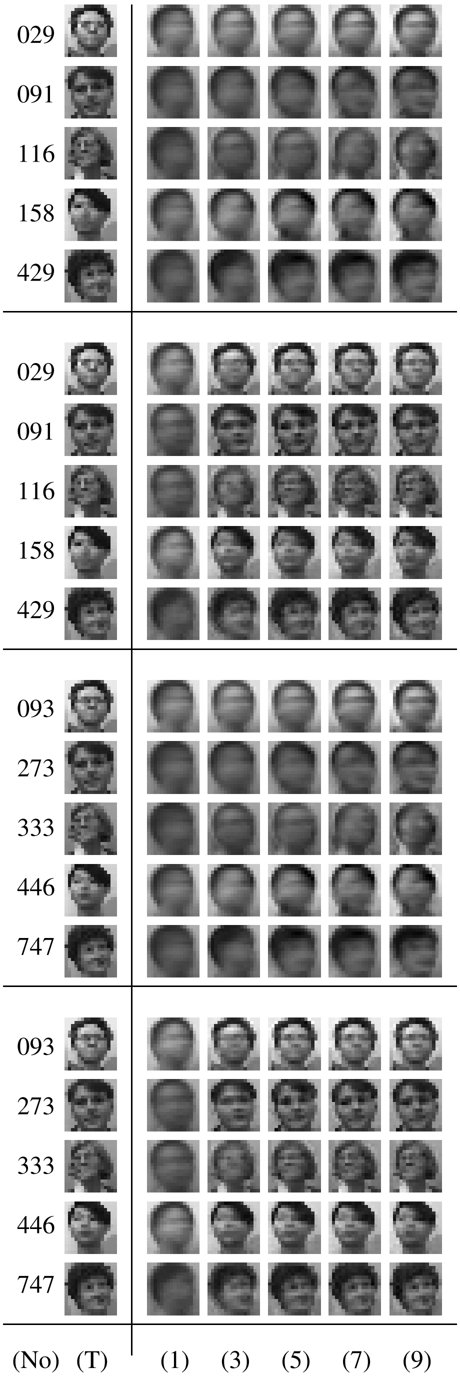

Figure 7.2. The images reconstructed from the training samples and the test samples by PCA and the proposed method. The blocks of the images from top to bottom correspond to the results on the training samples by PCA, the results on the training samples by the proposed method, the results on the test samples by PCA, and the results on the test samples by the proposed method. The images in row (No) are the results of the (No)-th target image. The images in column (T) are the target images. The images in column (1), (3), $\cdots$, and (9) are reconstructed with one, three, $\cdots$, and nine principal components, respectively.

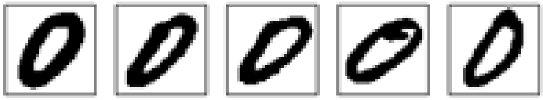

Figure 7.3. Sample images of the category ”zero” in an image database of hand-written numerals IPTP-CDROM1. The database has 10 categories which correspond to the numerals from zero, one, $\cdots$, to nine.

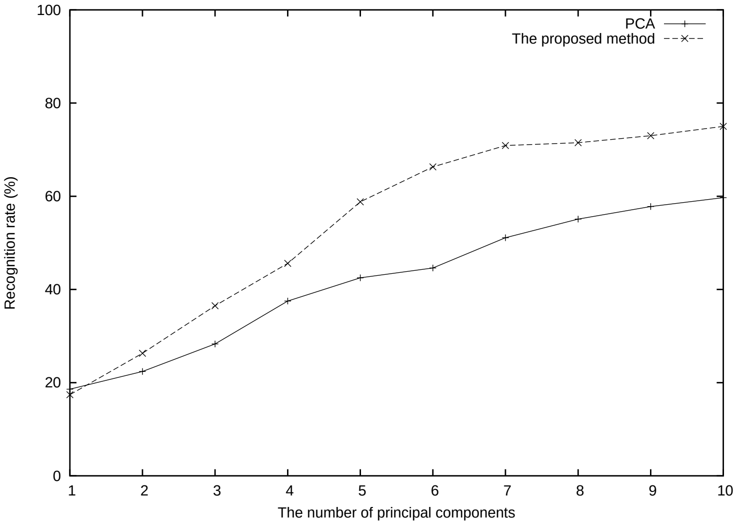

Figure 7.4. Recognition rate with Principal Component Analysis and the proposed method. The horizontal axis indicates the number of principal components used to classify a type of characters. Principal components are applied to the classification in order from the higher to the lower components.

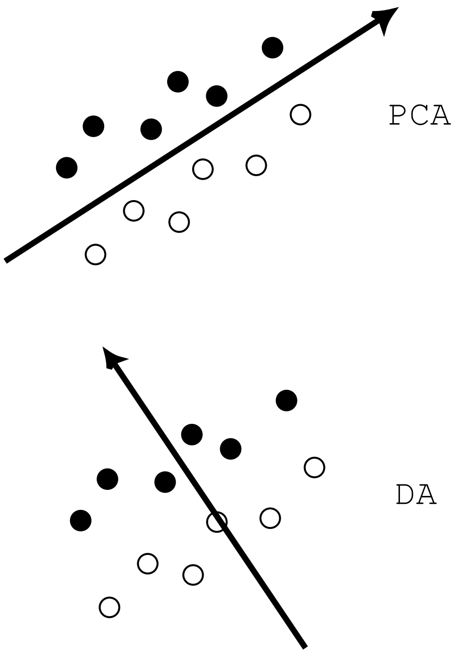

Figure 7.5. The axes constructed by PCA (above) and DA (below) with two categories of data. The axes do not coincide with each other, generally.

Chapter 8

Discussion

In Chapter 8, we discuss NLPCA in respect to complexity and redundancy. These properties are originated from its nonlinear extension, which frees the PCA from constraints such as orthogonality. The comparison of the proposed method to other methods is described.

8.1 Complexity and redundancy

The proposed method extends PCA nonlinearly preserving the order of principal components. This nonlinear extension releases the characteristic and constraints which originally PCA has.

Dimensionality reduction of data is considered as a deconstruction of information. In order to represent the data in high fidelity with a small number of principal components, the information of the data should be deconstructed adequately. The deconstruction of PCA is linear as follows,

$$\hat{\boldsymbol{x}}_i = \boldsymbol{e}_1 y_1 + \boldsymbol{e}_2 y_2 + \cdots + \boldsymbol{e}_i y_i. \tag{8.1}$$

Each principal component represents the coordinate on the line by the corresponding eigenvector. On the other hand, the deconstruction of information in the proposed method is,

$$\hat{\boldsymbol{x}}_i = \boldsymbol{\psi}_i(y_1, \cdots, y_i). \tag{8.2}$$

Each principal component represents coordinate on the curve by the nonlinear mapping function.

The degree of freedom in eigenvector of PCA is $m - 1$ under the constraint of normality where m equals the dimensionality of the data, while the degree of freedom in nonlinear mapping function of the proposed method is higher than the dimensionality. Consequently, the proposed method has high representation ability than PCA.

This dissertation does not describe in detail on how to determine the complexity of the extraction and reconstruction functions, and the balance between them.

The complexity of the mapping functions is considered to depend on the user's policy on how to extract the structure of the data-distribution. For example, we can represent all the samples with one principal component corresponding to a curve which connects all samples, if the mapping functions have enough complexity and their adjustment is successful. In this example, however, we cannot extract the summarized structure of the distribution. There exits a tradeoff between the complexity of the mapping functions and the number of the principal components. An external policy out of the proposed method will be required to determine the balance point.

In PCA, user needs not to consider the tradeoff, because its linearity determines the complexity. However, in the case of NLPCA, user should consider the tradeoff and determine the form of mapping function to be optimized.

In the proposed method, the information is deconstructed with preserving the ordinality of the principal components, while the ordinality is not preserved in other method of NLPCA such as the sandglass-type neural networks which deconstruct information to components equivalently. The policy in the deconstruction of information is different in each method of NLPCA. Therefore, the performance of dimensionality reduction cannot be compared with only the number of the principal components among the methods of NLPCA which includes PCA.

The principal components of the proposed method have redundancy in their scale, since the proposed method has no regularization for the extraction function, such as the normalization of the eigenvectors in PCA.

In PCA, the eigenvectors are calculated under the condition that the eigenvectors are the orthonormal vectors.

$$\boldsymbol{e}_i^T \boldsymbol{e}_j = \delta_{ij} \tag{8.3}$$

where $\delta_{ij} = 1$ if $i = j$, otherwise, $\delta_{ij} = 0$. In PCA, the distance between any pair of reconstructed vectors: $\hat{\boldsymbol{x}}_a$ and $\hat{\boldsymbol{x}}_b$ in the data space, equals to the distance between the pair of the corresponding principal components: $\boldsymbol{y}_a$ and $\boldsymbol{y}_b$, which is proved as follows,

$$\begin{aligned}
||\hat{\boldsymbol{x}}_a - \hat{\boldsymbol{x}}_b||^2 &= \Big(\sum_i y_{ai}\boldsymbol{e}_i - \sum_i y_{bi}\boldsymbol{e}_i\Big)^2 \\
&= \Big\{\sum_i (y_{ai} - y_{bi})\ \boldsymbol{e}_i\Big\}^2 \\
&= \sum_i \sum_j (y_{ai} - y_{bi})(y_{aj} - y_{bj})(\boldsymbol{e}_i \cdot \boldsymbol{e}_j) \\
&= \sum_i \sum_j (y_{ai} - y_{bi})(y_{aj} - y_{bj})\delta_{ij} \\
&= \sum_i (y_{ai} - y_{bi})^2 \\
&= ||\boldsymbol{y}_a - \boldsymbol{y}_b||^2, \qquad (8.4)
\end{aligned}$$

therefore,

$$||\hat{\boldsymbol{x}}_a - \hat{\boldsymbol{x}}_b|| = ||\boldsymbol{y}_a - \boldsymbol{y}_b||. \tag{8.5}$$

In the proposed method, however, the corresponding equation does not hold true as follows,

$$\begin{aligned} ||\hat{\boldsymbol{x}}_a - \hat{\boldsymbol{x}}_b|| &= ||\boldsymbol{\psi}(\boldsymbol{y}_a) - \boldsymbol{\psi}(\boldsymbol{y}_b)|| \\ \neq ||\boldsymbol{\phi}(\boldsymbol{x}_a) - \boldsymbol{\phi}(\boldsymbol{x}_b)|| &= ||\boldsymbol{y}_a - \boldsymbol{y}_b||, \end{aligned} \tag{8.6}$$

for any $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ where $\hat{\boldsymbol{x}} = \boldsymbol{\psi}(\boldsymbol{y})$ and $\boldsymbol{y} = \boldsymbol{\phi}(\boldsymbol{x})$, since the mapping functions do not have the orthogonality such as PCA. Consequently, the scale of principal components is indefinite. The nonlinear mapping functions of the proposed method and the eigenvectors of PCA are both optimized with the criterion of MSE on the data-reconstruction, while the constraint of the orthonomality is free in the proposed method.

How to give the scale of the principal components is application-oriented. A PCA gives a uniform scale, while we can give a variable scale in consideration of the density of the distribution; for example, we can assign a small scale to a high-density region and a large scale to a low-density region. We will try to regularize the scale with some criterion.

8.2 Comparison to other methods

Let us assume the ensemble of the m different SMLPs, and the i-th member of the ensemble has i hidden units in the bottleneck layer. In this case, the ensemble of the SMLPs will demonstrate the same performance as the proposed method with respect to computational cost and reconstruction accuracy.

However, the principal components extracted in each SMLP are not ordered on their contribution of the reconstruction, while the principal components in the proposed method are ordered so that their principal components represent the summarized structure of the data-distribution in the order of significance.

The proposed method has a drawback in that it requires greater computational cost than PCA does, however, the amount of the cost is not so large in an extraction process. Actually, the orders of the estimated computational costs to calculate the principal components are $O(mn)$ in PCA and $O(lmn)$ in the proposed method, where n is the dimensionality of the data, m is the dimensionality of the principal components, and l is the number of the units in the hidden layer per one principal component. The computational costs in SMLP and the Takahashi's method are equal to $O(lm(m+n))$. The l times evaluations of the sigmoid function is required in these neural network methods. Considering the advanced computational ability of the present days, these costs are small enough in a practical application.

As is described in Chapter 2, KPCA has several problems in its practical use with respect to computational costs, dimensionality reduction, and data-reconstruction.

When we perform KPCA for a dataset which contains N samples, we have to solve the eigenvalue problem of the $N \times N$ dot product matrix in the training process. In the test process, we have to evaluate the kernel function N times to extract each principal component. If N is large, both the proposed method and KPCA will be time-consuming in the training process, since the training process depends on N. However, the propose method will be less time-consuming than

KPCA in the test process, since the test process in the proposed method does not depends on N.

In KPCA, the dimensionality of the principal components can be larger than the dimensionality of the input space, since the maximum number of the principal components is equal to N. In this case, the dimensionality is not reduced but increased.

The pre-image of a principal component is not known in KPCA, since the data-mapping is unidirectional from the input space into the feature space. For example, when we apply KPCA to a set of facial images, we can not obtain the facial image corresponding to a principal component vector.

These problems have been overcome with the additional optimization to the framework of KPCA [16], [25], however the simplicity of KPCA has been lost.

The crucial disadvantage of the approaches of the autoassociator and the proposed method is the problems of local minima [21], while KPCA, which perform a linear optimization, does not have these problems. However, trial and error is required to determine the proper values of the kernel parameters in KPCA. In the authors' opinion, another problem of KPCA is that KPCA does not explicitly state the way to obtain the kernel parameters in its framework, while the proposed method explicitly optimizes its parameter by training algorithms.

In comparison with non-PCA based methods such as Huffman coding, JPEG and GIF compression, the proposed method and PCA have a feature to construct mapping functions especially for the objective data to utilize the bias of its distribution. Therefore, it is possible to use the proposed method as a pre-compressor for the non-PCA based methods to achieve higher compression rates.

Chapter 9

Conclusion

In this study, a novel method of Non Linear Principal Component Analysis (NLPCA) was proposed. The proposed method is a hierarchical nonlinear extension of Principal Component Analysis (PCA) which preserves the order of the principal components. In the proposed method, the mapping functions are implemented with a hierarchically arrayed neural network.

The proposed method is considered to have three advantages. The first advantage is the ordinality of the dimensionality. The higher principal components in the proposed method contribute more than lower components to represent data. The second advantage is the selectivity of the dimensionality. The user can determine the number of principal components to use after the adjustment of the mapping functions. The third advantage is the scalability of the dimensionality. The user can add a new principal component with only the adjustment of an additional mapping function.

In order to evaluate the effectiveness of the proposed method, several experiments were demonstrated utilizing artificial data sets and natural data sets sampled from several open databases.

In the first experiment we examined the proposed method's construction of curvilinear axes in the order of the significance, utilizing three-dimensional artificial samples which distribute on a surface. Since the axes were fitted nonlinearly to the distribution of the samples, the distribution was represented with fewer principal components than PCA. Moreover, we examined the proposed method's extraction of the features of a waveform with a small number of principal components. The experimental results suggest that the proposed method will provide more effective basis functions than the Fourier series expansion by adjusting the mapping functions for an objective data set.

In the second experiment, the representational ability of the mapping function was examined. In the beginning, the mapping functions implemented by the neural networks were mathematically formulated. The representational ability depends on the number of units in the hidden layer. In the next section, the data were reconstructed with variations in the number of units. The experimental results showed that even if only one of the representational ability of the extraction function or the representational ability of the reconstruction function is low, the total representation is not better. The combination of the extraction and the reconstruction function determines the total performance.

In the third experiment, the distortion of the reconstruction was examined comparing the distance between the original data and the distance between the corresponding reconstructed data. The experiment was demonstrated with PCA and the proposed method. Firstly, a pair of data vectors was chosen from the original data set, and the corresponding pair of reconstructed vectors was calculated. Next, the distance between the vectors in each pair was calculated. When these distances were compared, it was found that the data reconstructed by the proposed method preserved the structure of the distance more accurately than the data by PCA did. The result is considered to suggest that objective data may distribute nonlinearly.

In the fourth experiment, the practicality of the proposed method was examined. In the experiment, the proposed method was applied to data compression with facial images and feature extraction for classification with hand-written numerals. The experimental results showed that the proposed method could reconstruct facial images very faithfully, and the proposed method performed as a good feature extractor for the recognition of hand-written numerals.

In the discussion, complexity and redundancy of the proposed method were described. The proposed method is a nonlinear extension of PCA. Because of the extension, the constraint of orthogonality is removed, and the scale of the principal components is indefinite. In the dimensionality reduction of data, NLPCA is considered to deconstruct information. In NLPCA, the user can determine the method of this deconstruction by providing the form of the mapping functions. The method of deconstruction depends on the user's policy.

In the near future, the author expects to discuss the NLPCA from the perspective of its theory and applications. Theoretically, the regularization of the nonlinear mapping functions and the relationship between the mapping functions and the geometrical structure of the axes will be discussed. For the applications, the proposed method will be applied to a large-scale database to extract new knowledge. In this application, pattern generation from the principal component space will be tried, comparing the conventional interpolation and extrapolation methods. The final goal of this study is to adduce the general way of measurement and the representation of objective data based on its dimensionality.

Acknowledgements

I would like to express my sincere thanks to Professor Shuji Hashimoto who guided me during bachelor, master and doctoral courses and taught me the interesting world of research.

My great appreciations for Professor Tomiji Hisamura, Professor Hirochika Nakajima, Professor Atsushi Takeuchi and Professor Shigeo Morishima for their kindness of providing me with their time and expertise to review this dissertation and helpful comments.

Many thanks for my senior Dr. Pitoyo Hartono, Mr. Herath Kumarage Upali Samarasinghe, Mr. Hiroki Yoshizawa, and Dr. Kenji Suzuki for their valuable helps during my study and activities in Waseda University. I would like to thank my senior Dr. Hideyuki Sawada, and Dr. Yoshimitsu Aoki who encouraged me. Also thanks for Mr. Hitoshi Sakano who provided me with his time to discuss the study as a co-researcher.

Sincere thanks to my colleague and friend Mr. Mikito Toguro who enjoyed together in a lot of activities and enhanced my programming skills. My special thanks to the Neural Network Group and the colleagues of the doctoral courses and all members of past and present in Hashimoto laboratory for their valuable advices and discussion.

Finally I would like to express my greatest appreciation and gratitude to my parents and sister for their endless supports and warmly watching over the development of my study.

References

[1] C. Blake, C. Merz, UCI Repository of machine learning databases [http://www.ics.uci.edu/ mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science, 1998.

[2] D. DeMers and G. Cottrell, "Nonlinear dimensionality reduction", Advances in Neural Information Processing Systems 5, Morgan Kaufmann, pp.580-587, 1993.

[3] K. I. Diamantaras, S. Y. Kung, Principal Component Neural Networks Theory and Applications, John Wiley & Sons Inc, 1996.

[4] R. Duda, P. Hart, Pattern classification theory and systems, Springer-Verlag, 1988.

[5] R. Gnanadesikan, Methods for statistical data analysis of multivariate observations, John Wiley & Sons Inc, 1977.

[6] D. Graham, N. Allinson, Characterizing virtual eigensignatures for general purpose face recognition, Face Recognition From Theory to Applications, ed. H. Wechsler, et al., Springer Verlag, 1998. The UMIST Face Database, http://images.ee.umist.ac.uk/danny/database.html

[7] T. Hastie, W. Stuetzle, Principal curves, Journal of the Americal Statistical Association, Vol. 84, No.406, 1989.

[8] H. Hotelling, Analysis of complex statistical variables into principal components, Journal of Educational Psychology, 24, pp.417–441, pp.498-520, 1933.

[9] B. Irie, M. Kawato, Acquisition of internal representation by multi-layered perceptrons, Journal of IEICE, Vol.J73-D2 No.8, pp.1173-1178, 1990 (in Japanese).

[10] N. Japkowicz, S. Hanson, M. Gluck, Nonlinear Autoassociation is not Equivalent to PCA, Neural Computation, Vol.12, pp.531-545, 2000.

[11] N. Kambhalta, Dimension reduction by local principal component analysis, Neural Computation, Vol.9, pp.1493-1516, 1997.

[12] J. Karhunen, J. Joutsensalo, Generalization of principal component analysis, optimization problems, and neural network, Neural Networks, Vol.8, No.4, pp.549-562, 1995.

[13] M. Kramer, Nonlinear Principal Component analysis using Autoassociative Neural Networks, AIChE Journal, Vol.37, No.2, 1991.

[14] E. Malthouse, Liminations of Nonlinear PCA as Performed with Generic Neural Networks, IEEE Trans. Neural Networks, Vol.9, No.1, 1998.

[15] H. Masuda, T. Oohori, and K. Watanabe, A three-layered neural network for K-L transformation, Journal of IEICE, vol.J77-D-II, no.2, pp.397-404, 1994.

[16] S. Mika, B. Schölkopf, A. Smola, K.-R. Müller, M. Scholz, and G. Rütsch, Kernel PCA and de-noising in feature spaces, In Advances in Neural Information Processing Systems 11, 1999

[17] H. Murase and S. Nayar, 3D object recognition from appearance -Parametric Eigenspace Method-, Journal of IEICE, Vol.J77-D-II, No.11, pp2179-2187, 1994 (in Japanese).

[18] R. Saegusa, S. Hashimoto, Nonlinear principal component analysis using neural networks, Proc. of the 64th IPSJ general conference, vol.2, pp.205-206, 2002 (in Japanese).

[19] T. Sanger, Optimal unsupervised learning in a single-layer linear feedforward neural network, Neural networks, Vol.2, pp.459-473, 1989.

[20] T. Suenaga, A. Sato and H. Sakano, Cluster discriminant analysis for feature space visualization, Journal of IEICE, J85-D-II, No. 5, pp.785-795, 2002 (in Japanese).

[21] B. Schölkopf, A. Smola, and K. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Computation, Vol.10, No.5 (1998), 1299-1319.

[22] E. Saund, Dimensioanlity-Reduction Using Connectionist Networks, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.11, No.3, 1989.

[23] T. Takahashi, R. Tokunaga, and Y. Hirai, On supervised learning algorithm of three-layer linear perceptron -an extension of Baldi-Hrnik's theorem-, Journal of IEICE, vol.J80-D-II, no.5, pp.1267-1275, 1997.

[24] S. Tan, M. Mavrovouniotis, Reducing data dimsionality through optimizing neural network inputs, AIChE Journal, Vol.41, No.6, 1995.

[25] M. Tipping, Sparse kernel principal component analysis. In Advances in Neural Information Processing Systems 13, 2001.

[26] M.Turk and A.Pentland, Eigenfaces for recognition, Journal of Cognitive Neuroscience, Vol.3, No.1, pp.71-86, 1991.

[27] K. Watanabe, H. Ito, H. Matsuda, and T. Oohori, Multi-tandem perceptron for K-L transformation, Journal of IEICE, vol.J75-D-II, no.11, pp.1925-1932, 1992.

[28] K. Watanabe, T. Oohori, and T. Shimozawa, A theoretical study on the convergibility of unit perceptron, Journal of IEICE, vol.J75-D-II, no.11, pp.1933-1939, 1992.

A List of Publications

[Papers of Journals]

1. Ryo Seagusa, Hitoshi Sakano, Shuji Hashimoto, Nonlinear principal component analysis to preserve the order of principal components, Neurocomputing, no.61, pp.57-70, 2004.

2. 三枝 亮, 坂野 鋭, 橋本 周司, 主成分の順位を保持する非線形主成分分析, 電子情報通信学会論文誌, Vol.J86-D-II, No.7, pp.943-950, 2003.

3. Akihito Sudou, Pitoyo Hartono, Ryo Saegusa, Shuji Hashimoto, Signal reconstruction from sampled data tainted by aliasing phenomena using neural network, Journal of Signal Processing, Vol.7, No.1, pp.5-13, January 2003.

[Papers of International Conferences]

1. Ryo Saegusa, Hitoshi Sakano, Shuji Hashimoto, A nonlinear principal component analysis on image data, Proc. of 2004 IEEE Int. Workshop on Machine Learning for Signal Processing, pp.589-598 Sao Luis, Brazil, Sep. 29 - Oct. 1, 2004.

2. Ryo Saegusa, Shuji Hashimoto, On the evaluation of a nonlinear principal component analysis, Proc. of the 2nd IASTED Int. Conf. on Neural Networks and Computational Intelligence, pp.66-72, Grindelwald, Switzerland, Feb. 23-25, 2004.

3. Ryo Saegusa, Shuji Hashimoto, Nonlinear principal component analysis to preserve the order of principal components, Proc. of the 2nd Int. Conf. on Hybrid Intelligent Systems, pp.54-63, Santiago, Chile, Dec. 1-4, 2002.

4. Akihito Sudou, Pitoyo Hartono, Ryo Saegusa, Shuji Hashimoto, Signal reconstruction from sampled data using neural network, Proc. of 2002 IEEE Int. Workshop on Neural Networks for Signal Processing, pp.707-715, Valais, Switzerland, Sep. 4-6, 2002.

5. Ryo Saegusa, Pitoyo Hartono, and Shuji Hashimoto, Position-based competition learning of neural-networks array Proc. Int. Joint Conf. on Neural Networks, pp.2817-2820, Washington DC, USA, Jul. 15-19, 2001.

[Papers of National Conferences]

1. 三枝 亮, 坂野 鋭, 橋本 周司, 非線形主成分分析による画像データの解析, 画像の認識・理解シンポジウム (MIRU2004), II pp.67-72, 北海道, 2004 年 7 月 23 日-25 日.
2. 橋本 周司, Pitoyo Harutono, 三枝 亮, ニューラルネットワークアンサンブルによる情報の組織化, 第 4 回 SICE システムインテグレーション部門講演会 (SI2003), 計測自動制御学会, 2003 年 12 月 19 日-21 日.
3. 三枝 亮, 橋本 周司, 非線形主成分分析による実世界データの解析, 電子情報通信学会技術研究報告, Vol. 103, No.295, pp.55-60, 2003, 電子情報通信学会パターン認識・メディア理解研究会，北海道，2003 年 9 月 8,9 日.
4. 三枝 亮, 橋本 周司, 主成分の順位を保持する非線形主成分分析法, 電子情報通信学会技術研究報告, Vol.102, No.157, pp.25-30, 2002, 電子情報通信学会ニューロコンピューティング研究会, 沖縄，2002 年 6 月 27,28 日.

[Oral Presentations]

1. 三枝 亮, ニューラルネットワークを用いた非線形主成分分析と情報圧縮, 第 1 回若手研究者による Workshop 講演論文集, pp.5-6, 21 世紀 COE プログラム「超高齢社会における人とロボット技術の共生」, 東京都, 2004 年 4 月 24 日, 5 月 8 日.
2. 三枝 亮, 橋本 周司, 画像データの非線形主成分分析, 2004 年電子情報通信学会総合大会講演論文集, pp.240, 電子情報通信学会 2004 年総合大会, 東京都, 2004 年 3 月 22 日-25 日.
3. 三枝 亮, 橋本 周司, ニューラルネットワークによる非線形主成分分析 ー主成分順位を保持する階層化モジュール多層パーセプトロンー, 第 64 回情報処理学会全国大会論文集 (2), pp.205-206, 2002.
4. 三枝 亮, 橋本 周司, 複数環境に適応するニューラルネットワーク, 第 62 回情報処理学会全国大会論文集, pp. 89-90, 2001.
5. 三枝 亮, ピトヨ ハルトノ, 橋本 周司, 環境変化に高速に適応するニューラルネットワーク, 1999 年電子情報通信学会総合大会講演論文集, D-2-17, 1999, 1999 年電子情報通信学会総合大会, 東京都, 1999 年 3 月 27 日.

[Masters Dissertation]

三枝 亮, 空間的な位置関係を考慮したニューラルネットワークの競合学習, 早稲田大学, 2001.

[Bachelors Dissertation]

三枝 亮, 環境変化に高速に適応するニューラルネットワークの研究, 早稲田大学, 1999.