A NEW WAY OF PCA: INTEGRATED-SQUARED-ERROR AND EM ALGORITHMS

Jong-Hoon Ahn[†], *Seungjin Choi*[§], *Jong-Hoon Oh*[†]

[†]Department of Physics, POSTECH, Korea
[§]Department of Computer Science, POSTECH, Korea
{*jonghun, seungjin, jhoh*}*@postech.ac.kr*

ABSTRACT

Minimization of reconstruction error (squared-error) leads to principal subspace analysis (PSA) which estimates scaled and rotated principal axes of a set of observed data. In this paper we introduce a new alternative error, so called, *integrated-squared-error*, the minimization of which determines the exact principal axes (without rotational ambiguity) of a set of observed data. We consider the properties of the integrated-squared-error in the framework of coupled generative model, giving efficient EM algorithms for integrated-squared-error minimization. We revisit the generalized Hebbian algorithm (GHA) and show that it emerges from integrated-squared-error minimization in a single-layer linear feedforward neural network.

1. INTRODUCTION

Principal component analysis (PCA) is a widely-used linear dimensionality reduction technique. Its common derivation is in terms of a linear orthogonal projection that minimizes the squared reconstruction error. Principal axes of a set of observed variables can also be determined through maximum likelihood estimation of parameters in a latent variable model. Along this line, probabilistic PCA (PPCA) [7] and EM-PCA [5] were proposed. However, these methods find the scaled and *rotated* principal axes (principal subspace analysis rather than PCA), hence some postprocessing is required to compute exact principal directions (which corresponds to the orthogonal eigenvectors of the data covariance matrix)

In this paper we introduce a new error measure which integrates squared errors in a hierarchical fashion, hence called an *integrated-squared-error*. We consider the integrated-squared-error as a limiting case of the coupled generative model [2]. We show that exact principal axes of a set of observed variables emerge from the minimization of the integrated-squared-error and give simple EM algorithms for estimating exact principal directions iteratively. In addition, we revisit the generalized Hebbian algorithm (GHA) [6] and show that the minimization of the integrated-squared-error using the gradient descent method leads to the GHA.

2. PROBABILISTIC PCA

The probabilistic PCA (PPCA) [7] considers a linear generative model which assumes that the observed data $\boldsymbol{x} \in \mathbb{R}^d$ is generated by

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{\mu} + \boldsymbol{v}, \tag{1}$$

where the parameter matrix $\boldsymbol{A} \in \mathbb{R}^{d\times q}$ contains the factor loadings, the latent variables $\boldsymbol{s} \in \mathbb{R}^q$ have a unit isotropic Gaussian distribution, and $\boldsymbol{\mu}$ is a constant corresponding to the mean of the data ($d > q$). The noise $\boldsymbol{v}$ is also isotropic Gaussian, $\boldsymbol{v} \sim \mathcal{N}(0, \sigma^2\boldsymbol{I})$. It was shown in [7] that the maximum likelihood estimator $\boldsymbol{A}_{ML}$ is the matrix whose columns are the *scaled and rotated principal eigenvectors* of the sample covariance matrix of the data, even when the covariance model is approximate. The maximum likelihood estimator $\boldsymbol{A}_{ML}$ is given by $\boldsymbol{A}_{ML} = \boldsymbol{U}_q(\boldsymbol{\Lambda}_q - \sigma^2\boldsymbol{I})^{1/2}\boldsymbol{R}$, where $\boldsymbol{U}_q \in \mathbb{R}^{d\times q}$ contains q eigenvectors of the sample covariance matrix of the observed data with corresponding eigenvalues in the diagonal matrix $\boldsymbol{\Lambda}_q \in \mathbb{R}^{q\times q}$, $\boldsymbol{R} \in \mathbb{R}^{q\times q}$ is an arbitrary orthogonal rotation matrix, and $\boldsymbol{I}$ is the $d \times d$ identity matrix. The true principal axes can be recovered when the columns of $\boldsymbol{R}^T$ are equal to the eigenvectors of the matrix $\boldsymbol{A}^T\boldsymbol{A}$ matrix.

PCA can be viewed as a limiting case of the linear Gaussian model (1) as the noise variance σ^2 becomes infinitesimally small. Along this line, the EM-PCA algorithm was derived by taking zero noise limit into account [5]. In the case of zero noise limit, the linear generative model can be rewritten as $\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S}$ where the centered data matrix $\boldsymbol{X} \in \mathbb{R}^{d\times N}$ is defined by

$$\boldsymbol{X} = \left[\boldsymbol{x}_{(1)} - \boldsymbol{\mu} \ \cdots \ \boldsymbol{x}_{(N)} - \boldsymbol{\mu}\right] = \left[\widetilde{\boldsymbol{x}}_{(1)} \ \cdots \ \widetilde{\boldsymbol{x}}_{(N)}\right].$$

and $\boldsymbol{S} \in \mathbb{R}^{q\times N}$ is the latent variable matrix.

Algorithm Outline: EM-PCA [5]

E-step

$$\boldsymbol{S} = \left(\boldsymbol{A}^T\boldsymbol{A}\right)^{-1}\boldsymbol{A}^T\boldsymbol{X} \tag{2}$$

M-step

$$\widehat{\boldsymbol{A}} = \boldsymbol{X}\boldsymbol{S}^T\left(\boldsymbol{S}\boldsymbol{S}^T\right)^{-1} \tag{3}$$

As pointed out in [5], in the zero noise limit, the likelihood of a data point $\boldsymbol{x}$ is dominated solely by the squared distance between it and its reconstruction $\boldsymbol{A}\boldsymbol{s}$. In such a case, ML estimation of both $\boldsymbol{A}$ and $\boldsymbol{s}$ becomes a separable LS minimization problem [3]. The LS estimates, $\boldsymbol{A}$ and $\boldsymbol{S}$ are computed by

$$\widehat{\boldsymbol{A}}, \widehat{\boldsymbol{S}} = \min_{\boldsymbol{A},\boldsymbol{S}} \|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{S}\|^2. \tag{4}$$

The separable LS minimization is carried out in two steps. Minimizing (4) with respect to $\boldsymbol{A}$ with $\boldsymbol{S}$ being fixed, leads to the M-step updating (3). The estimate $\widehat{\boldsymbol{A}}$ is substituted back into (4), then we obtain a new criterion which is a function of $\boldsymbol{S}$ only

$$\min_{\boldsymbol{S}} \left\| \boldsymbol{X} \boldsymbol{P}_S^{\perp} \right\|_F^2, \tag{5}$$

where $\boldsymbol{P}_S^{\perp}$ is the orthogonal projection matrix given by

$$\boldsymbol{P}_S^{\perp} = \boldsymbol{I} - \boldsymbol{S}^T \left(\boldsymbol{S}\boldsymbol{S}^T \right)^{-1} \boldsymbol{S}. \tag{6}$$

The minimization of (5) leads to the E-step updating (2).

3. INTEGRATED-SQUARED-ERROR

In this section we introduce the integrated-squared-error and consider its properties, giving EM algorithms for estimating exact principal directions iteratively.

Definition 1 (Integrated-Squared-Error) *Given matrices $\boldsymbol{X} \in \mathbb{R}^{d \times N}$, $\boldsymbol{S} \in \mathbb{R}^{q \times N}$, and $\boldsymbol{A} \in \mathbb{R}^{d \times q}$, the integrated-squared-error between $\boldsymbol{X}$ and $\boldsymbol{A}\boldsymbol{S}$ is defined by a linear sum (with positive coefficients, $c_i > 0,\ i = 1, \ldots, q$) of squared errors $\mathcal{J}_i = \|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{I}_i\boldsymbol{S}\|^2$, i.e.,*

$$\mathcal{J}_{ISE}\left(\boldsymbol{A}, \boldsymbol{S}\right) = \sum_{i=1}^{q} c_i \mathcal{J}_i = \sum_{i=1}^{q} c_i \left\| \boldsymbol{X} - \boldsymbol{A}\boldsymbol{I}_i\boldsymbol{S} \right\|^2, \tag{7}$$

where $\boldsymbol{I}_i \in \mathbb{R}^{q \times q}$ is a diagonal matrix, so called, factor selection matrix *with $\boldsymbol{I}_i(j,j) = 1$ for $j = 1, \ldots, i$ and $\boldsymbol{I}_i(j,j) = 0$ for $j = i+1, \ldots, q$.*

Theorem 1 (Main Theorem) *$\boldsymbol{A}$ and $\boldsymbol{S}$ are globally minimal points of $\mathcal{J}_{ISE}\left(\boldsymbol{A}, \boldsymbol{S}\right)$ if and only if $\frac{\boldsymbol{a}_i}{\|\boldsymbol{a}_i\|} = \boldsymbol{\varphi}_i$ and $\frac{\vec{\boldsymbol{s}}_i}{\|\vec{\boldsymbol{s}}_i\|} = \boldsymbol{\xi}_i$ for $i = 1, \ldots, q$ where $\{\boldsymbol{\varphi}_i\}$ are the normalized eigenvectors of $\boldsymbol{X}\boldsymbol{X}^T$ and $\{\boldsymbol{\xi}_i\}$ are the normalized eigenvectors of $\boldsymbol{X}^T\boldsymbol{X}$ with associated eigenvalues of $\boldsymbol{X}\boldsymbol{X}^T$ (or $\boldsymbol{X}^T\boldsymbol{X}$), $\lambda_1 \geq \cdots \geq \lambda_q$.*

Proof: The proof is left out due to the space limitation. The detailed proof can be found in [1].

Remarks

- The integrated-squared-error (7) is minimized if and only if individual squared errors $\{\mathcal{J}_i\}$ are minimized since the integrated-squared-error is a linear sum of squared errors with positive coefficients $\{c_i\}$. Hence the $\boldsymbol{A}$ and $\boldsymbol{S}$ that minimize the integrated-squared-error, also minimize $\mathcal{J}_q = ||\boldsymbol{X} - \boldsymbol{A}\boldsymbol{S}|\,|^2$. However, the $\boldsymbol{A}$ and $\boldsymbol{S}$ which minimize $\mathcal{J}_q$, does not necessarily minimize the integrated-squared-error.
- The reconstruction error which is just squared error, $\|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{S}\|^2$ is invariant an orthogonal transform $\boldsymbol{R} \in \mathbb{R}^{q \times q}$ because $\boldsymbol{A}\boldsymbol{R}^{-1}$ and $\boldsymbol{R}\boldsymbol{S}$ contributes the same reconstruction error as $\boldsymbol{A}$ and $\boldsymbol{S}$. In contrast, the integrated-squared-error is not invariant under an orthogonal transformation because $\boldsymbol{R}^{-1}\boldsymbol{I}_i\boldsymbol{R} \neq \boldsymbol{I}_i, \forall i \neq q$.

The integrated-squared-error (7) is iteratively minimized by a simple EM algorithm, so called, *EM-ePCA* (exact principal directions are emphasized by a letter "e") which is summarized below:

Algorithm Outline: EM-ePCA

E-step

$$\boldsymbol{S} \;=\; \left[\mathsf{L}\left(\boldsymbol{A}^T\boldsymbol{A} \right) \right]^{-1} \boldsymbol{A}^T\boldsymbol{X}, \tag{8}$$

M-step

$$\widehat{\boldsymbol{A}} \;=\; \boldsymbol{X}\boldsymbol{S}^T \left[\mathsf{U}\left(\boldsymbol{S}\boldsymbol{S}^T \right) \right]^{-1}, \tag{9}$$

where the operator L is defined by

$$\mathsf{L}\left(y_{ij}\right) = \begin{cases} y_{ij} & \text{for } i \geq j \\ y_{ij} \frac{\sum_{k=j}^{q} c_k}{\sum_{k=i}^{q} c_k} & \text{for } i < j \end{cases}, \tag{10}$$

for an arbitrary square matrix $\boldsymbol{Y} = [y_{ij}]$ and $\mathsf{U}(\boldsymbol{Y}) = \mathsf{L}(\boldsymbol{Y}^T)^T$. It is shown in next section that these rules are derived from a coupled generative model.

We consider two limiting cases:

- In the limit of $\frac{c_{i+1}}{c_i} \to 0,\ i = 1, \ldots, q-1$, the operators L and U become usual lower/upper triangularization operators L_T and U_T where

  $$\mathsf{L}_T\left(y_{ij}\right) = \begin{cases} y_{ij} & \text{for } i \geq j \\ 0 & \text{for } i < j \end{cases}. \tag{11}$$

  The EM-updates (8) and (9) are further simplified as (**EM-ePCA (limiting case)**)

  $$\boldsymbol{S} \;=\; \left[\mathsf{L}_T\left(\boldsymbol{A}^T\boldsymbol{A} \right) \right]^{-1} \boldsymbol{A}^T\boldsymbol{X}, \tag{12}$$

  $$\widehat{\boldsymbol{A}} \;=\; \boldsymbol{X}\boldsymbol{S}^T \left[\mathsf{U}_T\left(\boldsymbol{S}\boldsymbol{S}^T \right) \right]^{-1}. \tag{13}$$

  Note that EM-ePCA (limiting case) algorithm is involved with the triangular matrix inversion, hence, computational complexity is greatly reduced, especially for the case of high-dimensional data.
- The EM-PCA algorithm [5] is a special limiting case of our model as $c_i \to 0,\ i = 1, \ldots, q-1$. Under this limit, the inference in (8) reduces to simple least squares projection. The M-step update (9) becomes Wiener filtering.

4. COUPLED GENERATIVE MODEL

A main motivation of the integrated-squared-error (7) came from the coupled linear generative model [2] where a set of linear Gaussian model shares the same latent variables $\boldsymbol{s} \in \mathbb{R}^q$ and parameters $\boldsymbol{A} \in \mathbb{R}^{d \times q}$ with different factor selection matrices $\{\boldsymbol{I}_i\}$. The q-coupled generative model is described by

$$\begin{cases} \boldsymbol{x}_1 &= \boldsymbol{A}\boldsymbol{I}_1\boldsymbol{s} + \boldsymbol{\mu}_1 + \boldsymbol{v}_1, \\ \boldsymbol{x}_2 &= \boldsymbol{A}\boldsymbol{I}_2\boldsymbol{s} + \boldsymbol{\mu}_2 + \boldsymbol{v}_2, \\ \quad\vdots & \quad\vdots \\ \boldsymbol{x}_q &= \boldsymbol{A}\boldsymbol{I}_q\boldsymbol{s} + \boldsymbol{\mu}_q + \boldsymbol{v}_q, \end{cases} \tag{14}$$

where $\boldsymbol{I}_i \in \mathbb{R}^{q \times q}$ is a diagonal matrix with $\boldsymbol{I}_i(j,j) = 1$ for $j = 1, \ldots, i$ and $\boldsymbol{I}_i(j,j) = 0$ for $j = i+1, \ldots, q$.

The coupled linear generative model shares the same latent variables $\boldsymbol{s}$ and factor loading matrix $\boldsymbol{A}$, but takes different isotropic Gaussian noise models $\{\boldsymbol{v}_i \sim \mathcal{N}(0, \sigma_i^2\boldsymbol{I})\}$ and factor selection matrices $\{\boldsymbol{I}_i\}$. The factor selection matrix $\boldsymbol{I}_i$ is designed in such a way that first i principal directions are selected when each model observes the same data, i.e., $\boldsymbol{x}_1 = \cdots = \boldsymbol{x}_q = \boldsymbol{x}$.

For mutually independent isotropic Gaussian noise models, the joint probability distribution $p(\boldsymbol{x}_1 = \boldsymbol{x}, \cdots, \boldsymbol{x}_q = \boldsymbol{x}|\boldsymbol{s})$ over

q-coupled $\boldsymbol{x}$ spaces, conditioned on latent variables $\boldsymbol{s}$ is factorized as

$$\begin{aligned} & \prod_{i=1}^{q} p(\boldsymbol{x}_i = \boldsymbol{x}|\boldsymbol{s}; i) \\ = \; & \prod_{i=1}^{q} (2\pi\sigma_i^2)^{-d/2} \exp\left\{-\frac{1}{2\sigma_i^2} \parallel \boldsymbol{x} - \boldsymbol{A}\boldsymbol{I}_i\boldsymbol{s} - \boldsymbol{\mu} \parallel^2\right\}, \end{aligned}$$

where $p(\boldsymbol{x}_i = \boldsymbol{x}|\boldsymbol{s}; i)$ is the conditional density for the ith generative model and $\int p(\boldsymbol{x}, \cdots, \boldsymbol{x}|\boldsymbol{s})d\boldsymbol{x} \neq 1$.

The expected complete-data log-likelihood $\langle \mathcal{L}_C \rangle$ is given by (see [2] for more details)

$$\begin{aligned} \langle \mathcal{L}_C \rangle &= -\sum_{n=1}^{N}\sum_{i=1}^{q}\left\{\frac{d}{2}\log\sigma_i^2 + \frac{1}{2q}\mathrm{tr}\left(\left\langle \boldsymbol{s}_{(n)}\boldsymbol{s}_{(n)}^T\right\rangle\right)\right. \\ &\quad + \frac{1}{2\sigma_i^2} \parallel \widetilde{\boldsymbol{x}}_{(n)} \parallel^2 - \frac{1}{\sigma_i^2}\left\langle \boldsymbol{s}_{(n)}\right\rangle^T \boldsymbol{I}_i^T\boldsymbol{A}^T\widetilde{\boldsymbol{x}}_{(n)} \\ &\quad \left. + \frac{1}{2\sigma_i^2}\mathrm{tr}\left(\boldsymbol{I}_i^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{I}_i\left\langle \boldsymbol{s}_{(n)}\boldsymbol{s}_{(n)}^T\right\rangle\right)\right\}, \end{aligned} \tag{15}$$

where tr$(\cdot)$ denotes the trace operator, and $\langle\cdot\rangle$ denotes the statistical expectation taken with respect to $p\left(\boldsymbol{s}_{(n)}|\boldsymbol{x}_{(n)}, \cdots, \boldsymbol{x}_{(n)}; \boldsymbol{A}, \sigma_i^2\right)$. The terms irrelevant to parameters were left out.

In E-step, sufficient statistics are computed:

$$\begin{aligned} \left\langle \boldsymbol{s}_{(n)} \right\rangle &= \boldsymbol{M}^{-1}\boldsymbol{Q}^T\boldsymbol{A}^T\widetilde{\boldsymbol{x}}_{(n)}, \\ \left\langle \boldsymbol{s}_{(n)}\boldsymbol{s}_{(n)}^T \right\rangle &= \boldsymbol{M}^{-1} + \left\langle \boldsymbol{s}_{(n)} \right\rangle \left\langle \boldsymbol{s}_{(n)} \right\rangle^T. \end{aligned} \tag{16}$$

In M-step, parameters $\left\{\boldsymbol{A}, \sigma_i^2\right\}$ are updated by

$$\begin{aligned} \widehat{\boldsymbol{A}} &= \left[\sum_{n=1}^{N}\widetilde{\boldsymbol{x}}_{(n)}\left\langle \boldsymbol{s}_{(n)}\right\rangle^T\right]\boldsymbol{Q}^T\left[\sum_{i,n=1}^{q,N}\boldsymbol{I}_i\frac{\left\langle \boldsymbol{s}_{(n)}\boldsymbol{s}_{(n)}^T\right\rangle}{\sigma_i^2}\boldsymbol{I}_i^T\right]^{-1} \\ \sigma_i^2 &= \frac{1}{Nd}\sum_{n=1}^{N}\left\{\parallel \widetilde{\boldsymbol{x}}_{(n)} \parallel^2 - 2\left\langle \boldsymbol{s}_{(n)}\right\rangle^T\boldsymbol{I}_i^T\widehat{\boldsymbol{A}}^T\widetilde{\boldsymbol{x}}_{(n)}\right. \\ &\quad \left. + \mathrm{tr}\left(\left\langle \boldsymbol{s}_{(n)}\boldsymbol{s}_{(n)}^T\right\rangle\boldsymbol{I}_i^T\widehat{\boldsymbol{A}}^T\widehat{\boldsymbol{A}}\boldsymbol{I}_i\right)\right\}. \end{aligned} \tag{17}$$

where the posterior model covariance matrix $\boldsymbol{M} \in \boldsymbol{R}^{q\times q}$ and the matrix $\boldsymbol{Q}$ are defined by $\boldsymbol{M} = \sum_{i=1}^{q}\boldsymbol{I}_i^T\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{I}_i/\sigma_i^2 + \boldsymbol{I}$ and $\boldsymbol{Q} = \sum_{i=1}^{q}\boldsymbol{I}_i/\sigma_i^2$.

Now we consider a limiting case of the coupled linear generative model (14) as

$$\sigma_q^2 \to 0, \;\; \sigma_q^2/\sigma_i^2 = c_i, \quad i = 1, \ldots, q. \tag{18}$$

In this case, maximizing the log-likelihood is practically identical to minimizing the integrated squared error. This is confirmed by computing $\lim_{\{\sigma_i^2\to 0\}}\sigma_q^2\langle\mathcal{L}_C\rangle$ and omitting constants. The EM-updates (16) and (17) reduce to the EM-ePCA algorithm described in (8) and (9).

5. A LINK WITH GENERALIZED HEBBIAN ALGORITHM

The generalized Hebbian algorithm (GHA) [6] is one of well-known PCA neural network algorithms. The convergence behavior of GHA was well studied in [6], however, an optimality criterion for GHA is not clear yet. Here we show that the minimal integrated-squared-error in a single layer linear feedforward net leads to the GHA by equalizing the weights in the recognition model to the weights in the generative model. Under this, hidden variables $\boldsymbol{s}$ are estimated by $\boldsymbol{s} = \boldsymbol{A}^T\boldsymbol{x}$. The gradient descent method (for integrated-squared-error minimization) gives the updating rule for $\boldsymbol{A}^T$ which has the form

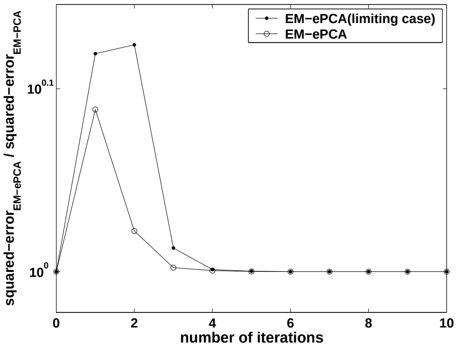

Fig. 1. Our proposed EM algorithms, EM-ePCA and its limiting case, show a slightly different convergence behavior in terms of only squared error $\mathcal{J}_q$, compared to the EM-PCA algorithm. It seems that our EM algorithms are slightly slower than the EM-PCA algorithm in a first few iterations, since our EM algorithms tries to minimize the integrated-squared-error rather than just single squared error $\mathcal{J}_q$. However, it takes almost same number of iterations for all the algorithms to achieve the final convergence. In this simulation, $c_i = 0.8^i$ were used.

$$\boldsymbol{A}^T \leftarrow \boldsymbol{A}^T + \eta\left(\sum_{i=1}^{q}2c_i\boldsymbol{I}_i\right)\left\{\boldsymbol{S}\boldsymbol{X}^T - \mathsf{L}\left(\boldsymbol{S}\boldsymbol{S}^T\right)\boldsymbol{A}^T\right\}. \tag{19}$$

In order for each row vector of $\boldsymbol{A}^T$ to be updated with identical learning rate, we take a learning rate η as

$$\eta = \eta_0\left(\sum_{i=1}^{q}2c_i\boldsymbol{I}_i\right)^{-1}, \tag{20}$$

to obtain an updating rule for $\boldsymbol{A}^T$:

$$\boldsymbol{A}^T \leftarrow \boldsymbol{A}^T + \eta_0\left\{\boldsymbol{S}\boldsymbol{X}^T - \mathsf{L}\left(\boldsymbol{S}\boldsymbol{S}^T\right)\boldsymbol{A}^T\right\}. \tag{21}$$

In the limit $c_{i+1}/c_i \to 0$ for $i = 1, \ldots, q-1$, the operator L becomes L_T. Hence the algorithm (21) reduces to the GHA [6].

The converged weights $\boldsymbol{A}^T$ minimizes the integrated-squared-error under the constraints $\boldsymbol{S} = \boldsymbol{A}^T\boldsymbol{X}$:

$$\mathcal{J} = \sum_{i=1}^{q}c_i \parallel \boldsymbol{X} - \boldsymbol{A}\boldsymbol{I}_i\boldsymbol{A}^T\boldsymbol{X} \parallel^2 \tag{22}$$

Reversely, the weights $\boldsymbol{A}^T$ that minimizes the integrated-squared-error satisfy $\boldsymbol{A}^T\boldsymbol{A} = \boldsymbol{I}$ and $\boldsymbol{A}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{A} = \mathsf{U}(\boldsymbol{A}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{A})$. The error function really gives the normalized principal axes of $\boldsymbol{X}\boldsymbol{X}^T$.

The derivation of the GHA has been already treated in [4] where they proposed a criterion to be maximized in the generalization of variance maximization. It uses the recognition model and the weights are constrained to be orthogonal via the Lagrange multipliers. In our method, however, the orthogonality emerges from the minimal integrated-squared-error without orthogonality constraint and we use the alternating model of recognition and generation with the same weights.

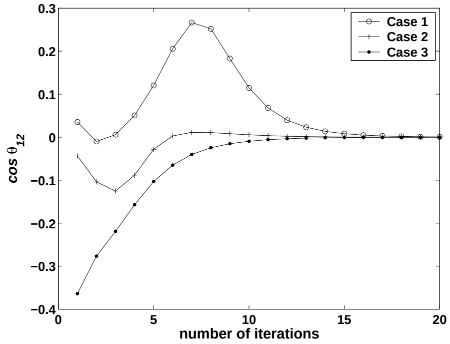

Fig. 2. Our EM-ePCA algorithm estimates the orthogonal eigenvectors of the data covariance matrix. Angles between the first and second principal directions are plotted with respect the number of iterations. The convergence is not monotonic, however, after some iterations our EM-ePCA algorithm find exact two principal directions which are orthogonal each other. Here three different realizations of data were considered.

6. NUMERICAL EXAMPLES

We investigate the convergence behavior and the performance of our EM algorithms: (1) EM-ePCA in Eqs. (8) and (9); (2) EM-ePCA (limiting case) in Eqs. (12) and (13), compared to the EM-PCA algorithm in Eqs. (2) and (3). These three algorithms were tested using USPS data whose dimension is 256. Fig. 1 shows the convergence behavior of all these three algorithms. In terms of only squared error $\mathcal{J}_q$ (for the case of $q = 20$), it takes almost same number of iterations for all three algorithms to reach the final convergence. However, note that our EM algorithms find exact principal directions (without rotation ambiguity), whereas the EM-PCA finds the principal subspace. Fig. 2 shows the time evolution of angle between first two directions estimated by our EM-ePCA algorithm. The convergence is not always monotonic, but, the orthogonality is always guaranteed. We also applied our EM-ePCA algorithm to a non-Gaussian data (see Fig. 3) in order to show that our algorithm does not get stuck in a local minimum even for the non-Gaussian data.

7. CONCLUSION

We have introduce a new error measure, the integrated-squared-error, as an alterative to the conventional reconstruction error for PCA. We have shown that exact principal directions of a set of observed data emerged through the integrated-squared-error minimization and have presented simple but efficient EM algorithms. In fact our EM-ePCA algorithm and its limiting case become more efficient when the extraction of a few principal components from very high-dimensional data is required. We have also revisited GHA, showing that it could be derived using the gradient descent method by minimizing the integrated-squared-error.

8. ACKNOWLEDGMENT

This work was supported by Korea Ministry of Science and Technology under Brain Science and Engineering Research Program, KOSEF 2000-2-20500-009-5, and Brain Korea 21 in POSTECH.

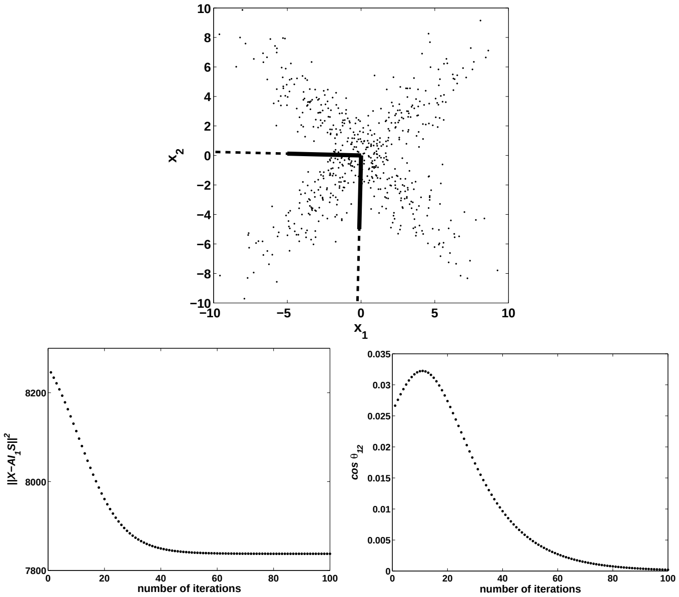

Fig. 3. A scatter plot of some exemplary non-Gaussian data is shown in the upper panel. The dashed lines indicate the directions of the two leading eigenvectors of the sample covariance matrix whose diagonal components are very close to each other. In the lower pannel, the convergence in terms of $\mathcal{J}_1$ (left pannel) and the angle between first two principal directions (right pannel) is shown. Notice that the difficult learning does not get stuck in a local minimum, although it does take more than 100 iterations to converge, which is unusual for Gaussian data [5].

9. REFERENCES

[1] J. H. Ahn, S. Choi, and J. H. Oh, “A new way of PCA: Integrated-squared-error minimization,” *IEEE Trans. Neural Networks*, 2003, submitted.

[2] J. H. Ahn and J. H. Oh, “A constrained EM algorithm for principal component analysis,” *Neural Computation*, vol. 15, no. 1, pp. 57–65, 2003.

[3] S. Choi, “Sequential EM learning for subspace analysis,” in *Proc. ITC-CSCC*, Phuket, Thailand, 2002.

[4] J. Karhunen and J. Joutsensalo, “Generalization of principal component analysis, optimization problems, and neural networks,” *Neural Networks*, vol. 8, no. 4, pp. 549–562, 1995.

[5] S. Roweis, “EM algorithms for PCA and SPCA,” in *Advances in Neural Information Processing Systems*, vol. 10. MIT press, 1998, pp. 626–632.

[6] T. D. Sanger, “Optimal unsupervised learning in a single-layer linear feedforward neural network,” *Neural Networks*, vol. 2, no. 6, pp. 459–473, 1989.

[7] M. E. Tipping and C. M. Bishop, “Mixtures of probabilistic principal component analyzers,” *Neural Computation*, vol. 11, no. 2, pp. 443–482, 1999.