Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Learning robust principal components from L1-norm maximization*

Ding-cheng FENG[1,2], Feng CHEN‡[1,2], Wen-li XU[1,2]

([1]*Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100084, China*)

([2]*Department of Automation, Tsinghua University, Beijing 100084, China*)

E-mail: fdc08@mails.tsinghua.edu.cn; chenfeng@tsinghua.edu.cn; xuwl@tsinghua.edu.cn

Received June 11, 2012; Revision accepted Nov. 12, 2012; Crosschecked Nov. 12, 2012

Abstract: Principal component analysis (PCA) is fundamental in many pattern recognition applications. Much research has been performed to minimize the reconstruction error in L1-norm based reconstruction error minimization (L1-PCA-REM) since conventional L2-norm based PCA (L2-PCA) is sensitive to outliers. Recently, the variance maximization formulation of PCA with L1-norm (L1-PCA-VM) has been proposed, where new greedy and non-greedy solutions are developed. Armed with the gradient ascent perspective for optimization, we show that the L1-PCA-VM formulation is problematic in learning principal components and that only a greedy solution can achieve robustness motivation, which are verified by experiments on synthetic and real-world datasets.

Key words: Principal component analysis (PCA), Outliers, L1-norm, Greedy algorithms, Non-greedy algorithms
doi:10.1631/jzus.C1200180 **Document code:** A **CLC number:** TP391.4

1 Introduction

Principal component analysis (PCA) is one of the most popular dimensionality reduction methods in statistical pattern recognition (Duda *et al.*, 2001; Hastie *et al.*, 2005; Bishop, 2006). PCA can construct orthogonal transformation which maps high dimensional data to low dimensional representations such that the information loss is minimized. Due to its effectiveness and simplicity, PCA and its variants have been widely applied in many data mining tasks such as exploratory data analysis, data preprocessing, data compression and reconstruction, and data visualization (de la Torre and Black, 2001; Zass and Shashua, 2007; Zhang and Teng, 2010; Nakajima *et al.*, 2011).

The robustness problem of PCA has attracted much attention in the past decade (Wright *et al.*, 2009). The conventional L2-norm based PCA (L2-PCA) is considered to be sensitive to outliers since the effect of outliers with large norms can be exaggerated by the L2-norm (Kwak, 2008), which leads to inaccurate estimation of the transformation matrices (principal bases). Since L1-norm is more robust to outliers, much work has applied L1-norm based reconstruction error minimization (L1-PCA-REM) to alleviate the drawback of L2-PCA (Ke and Kanade, 2003; 2005). Ding *et al.* (2006) proposed a rotational invariant L1-norm PCA (R1-PCA), where the reconstruction error is controlled by a new $L_{2,1}$ norm (Liu *et al.*, 2009; Nie *et al.*, 2010). Observing that these studies are either ambiguous in interpreting the learning process or difficult for optimization, Kwak (2008) combined the variance maximization formulation of L2-norm PCA and L1-norm idea in the reconstruction error minimization formulation of conventional PCA, and proposed a new simpler L1-norm and variance maximization based PCA (L1-PCA-VM). Furthermore, Kwak (2008) proposed an

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 61071131 and 61271388), the Beijing Natural Science Foundation (No. 4122040), the Research Project of Tsinghua University (No. 2012Z01011), and the United Technologies Research Center (UTRC)

efficient greedy algorithm to achieve robust PCA solutions. Recently, Nie *et al.* (2011) proposed a new non-greedy algorithm, L1-norm maximization, for the L1-norm variance maximization PCA, where higher L1-norm objective function values can be achieved.

This paper presents a comparative study of different optimization methods for L1-norm variance maximization PCA (L1-PCA-VM). Specifically, we show that the objective of L1-PCA-VM can be problematic in learning bases as conventional L2-PCA and dealing with outliers as the L1-PCA-REM formulation. We compare the greedy (Kwak, 2008) and non-greedy (Nie *et al.*, 2011) solutions with L1-PCA-VM, and propose a new solution based on gradient ascent with orthonormal projection to facilitate analysis. Experiments on synthetic and real-world datasets show that the non-greedy methods for L1-PCA-VM, which are better than greedy methods (evaluated by the objective function), often return different and poorer principal bases, especially when outliers are incorporated. This contrasts with the motivation of the L1-PCA-VM formulation in learning better principal bases, especially in datasets with outliers. Our results suggest that L1-PCA-VM is reasonable only with the greedy solutions, which can potentially overcome the drawback of conventional L2-PCA.

2 Preliminaries and notations

Denoting the L1- and L2-norm of a matrix $\boldsymbol{M}$ by $\|\boldsymbol{M}\|_1=\sum_{i,j}|m_{i,j}|$ and $\|\boldsymbol{M}\|_2=\sqrt{\sum_{i,j}m_{i,j}^2}$, respectively, the conventional L2-norm based PCA (L2-PCA) can be formulated as solving the variance maximization problem

$$\max_{\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}=\boldsymbol{I}} \mathcal{J}_{l_2}(\boldsymbol{W}) = \left\| \boldsymbol{W}^{\mathrm{T}}\boldsymbol{X} \right\|_2, \tag{1}$$

where $\boldsymbol{X}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_n]\in\mathbb{R}^{d\times n}$ is the given centralized data (d and n are the dimensionality and number of samples, respectively), $\boldsymbol{W}=[\boldsymbol{w}_1,\boldsymbol{w}_2,\ldots,\boldsymbol{w}_m]\in\mathbb{R}^{d\times m}$ ($m<d$) is the projection matrix whose columns are m orthonormal bases in $\mathbb{R}^d$, and $\boldsymbol{I}$ is an identity matrix. Since L2-norm is sensitive to outliers, Kwak (2008) proposed a similar L1-norm based PCA (L1-PCA-VM):

$$\max_{\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}=\boldsymbol{I}} \mathcal{J}_{l_1}(\boldsymbol{W}) = \left\| \boldsymbol{W}^{\mathrm{T}}\boldsymbol{X} \right\|_1. \tag{2}$$

Since it is difficult to solve the nonconvex problem (2) directly, Kwak (2008) first proposed an algorithm (called PCA-L1) to solve

$$\max_{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}=1} \mathcal{J}_{l_1}(\boldsymbol{w}) = \left\| \boldsymbol{w}^{\mathrm{T}}\boldsymbol{X} \right\|_1, \tag{3}$$

where $\boldsymbol{w}\in\mathbb{R}^{d\times 1}$, and then used a greedy strategy to find a local optimum (Algorithm 1).

Algorithm 1 Greedy search strategy

1: Initialize $\boldsymbol{w}_0$
2: **for** $j=1$ to m **do**
3: Update data $\boldsymbol{X}=\boldsymbol{X}-\boldsymbol{w}_{j-1}(\boldsymbol{w}_{j-1}^{\mathrm{T}}\boldsymbol{X}^{j-1})$
4: Find $\boldsymbol{w}_j$ by solving Eq. (3) using PCA-L1
5: **end for**
6: **return** $\boldsymbol{W}=[\boldsymbol{w}_1,\boldsymbol{w}_2,\ldots,\boldsymbol{w}_m]$

Motivated by the fact that the PCA-L1 algorithm is interminable in convergence and ineffective in finding good solutions in some cases, Nie *et al.* (2011) followed Eq. (2), and proposed a non-greedy algorithm to achieve better solutions. The non-greedy method usually obtains much higher objective values than the greedy method.

3 Analysis of L1-norm PCA

We show that Eq. (2) is problematic in learning principal components from data.

In traditional L2-norm PCA, the principal components can denote: (1) the orthonormal bases, called 'principal bases' (The bases returned by the greedy method can be called 'principal axes'. When the data is projected on these bases, it will have maximal orderly variances. Principal axes are special principal bases.) or (2) space spanned by the orthonormal bases, called 'principal space'. Due to the rotational invariance property of L2-norm (Ding *et al.*, 2006), different orthonormal bases lead to the same objective value in Eq. (1): we have $\mathcal{J}_{l_2}(\boldsymbol{W})=\mathcal{J}_{l_2}(\boldsymbol{WR})$, where $\boldsymbol{R}_{m\times m}$ is an arbitrary orthogonal matrix satisfying $\boldsymbol{R}\boldsymbol{R}^{\mathrm{T}}=\boldsymbol{I}$. However, this property does not hold in L1-norm, which leads to problematic solutions when Eq. (2) is used. We begin with the following example:

Example 1 (Samples are in a line) Given data $\boldsymbol{X}=[\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_n]\in\mathbb{R}^{3\times n}$ generated from Gaussian distribution $\boldsymbol{x}_i\sim\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$ where $\boldsymbol{\mu}=[0,0,0]^{\mathrm{T}}$, $\boldsymbol{\Sigma}=\mathrm{diag}\{\sigma,0,0\}$, find the principal space with dimensionality $m=2$ under Eq. (2). Algorithm 1

will return $\boldsymbol{W}_{\mathrm{g}}=[\boldsymbol{w}_1,\boldsymbol{w}_2]$ (principal axes) where $\boldsymbol{w}_1=[1,0,0]^{\mathrm{T}}$, and $\boldsymbol{w}_2\perp\boldsymbol{w}_1$. The expectation of the objective (mean deviation in the projection direction) is $\mathcal{J}_{l_1}(\boldsymbol{W}_{\mathrm{g}})=nE\left\{\left\|\boldsymbol{W}_{\mathrm{g}}^{\mathrm{T}}\boldsymbol{x}\right\|_1\right\}=n(1+0)C=nC$ ($C=\sigma\sqrt{2/\pi}$). Consider the other solution $\boldsymbol{W}_{\mathrm{n}}=[\boldsymbol{w}_1,\boldsymbol{w}_2]$ where $\boldsymbol{w}_1=\left[\sqrt{2}/2,\sqrt{2}/2,0\right]^{\mathrm{T}}$, $\boldsymbol{w}_2=\left[\sqrt{2}/2,-\sqrt{2}/2,0\right]^{\mathrm{T}}$. The expectation of objective is $\mathcal{J}_{l_1}(\boldsymbol{W}_{\mathrm{n}})=nE\left\{\left\|\boldsymbol{W}_{\mathrm{n}}^{\mathrm{T}}\boldsymbol{x}\right\|_1\right\}=n(\sqrt{2}/2+\sqrt{2}/2)C=\sqrt{2}nC>\mathcal{J}_{l_1}(\boldsymbol{W}_{\mathrm{g}})$. Therefore, $\boldsymbol{W}_{\mathrm{n}}$ is a better solution than $\boldsymbol{W}_{\mathrm{g}}$ (evaluated by Eq. (2)). However, this is unreasonable since $\boldsymbol{W}_{\mathrm{g}}$ corresponds to principal axes, which are intuitively better principal components.

We also provide a perspective from gradient based methods for Eq. (2). In unsupervised feature learning, the independent component analysis (ICA) model (Hyvärinen *et al.*, 2009)

$$\min_{\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}=\boldsymbol{I}}\left\|\boldsymbol{W}^{\mathrm{T}}\boldsymbol{X}\right\|_1 \tag{4}$$

is a minimization problem and can be approximated by gradient descent with projection (Le *et al.*, 2011). Similarly, we can propose a new gradient ascent algorithm for Eq. (2) (Algorithm 2) to facilitate the analysis of L1-norm PCA.

Algorithm 2 Gradient ascent L1-norm PCA

Initialize $\boldsymbol{W}=\boldsymbol{W}_{\mathrm{old}}$
while the stop criterion is not satisfied **do**
  Compute approximate $\nabla\boldsymbol{W}$ of Eq. (2) at $\boldsymbol{W}_{\mathrm{old}}$
  if $\exists\alpha$ such that $f(\boldsymbol{W}_{\mathrm{orth}})>f(\boldsymbol{W}_{\mathrm{old}})$ where $\boldsymbol{W}_{\mathrm{new}}=\boldsymbol{W}_{\mathrm{old}}+\alpha\nabla\boldsymbol{W}$ (gradient ascent), $\boldsymbol{W}_{\mathrm{orth}}=\boldsymbol{W}_{\mathrm{new}}(\boldsymbol{W}_{\mathrm{new}}^{\mathrm{T}}\boldsymbol{W}_{\mathrm{new}})^{-\frac{1}{2}}$ (projection) **then**
    $\boldsymbol{W}_{\mathrm{old}}=\boldsymbol{W}_{\mathrm{orth}}$
  else
    Break
  end if
end while
return $\boldsymbol{W}=\boldsymbol{W}_{\mathrm{old}}$

Remark 1 Algorithm 2 provides a new perspective on the L1-norm PCA problem, where the optimization can be performed by a simple gradient with projection. The stop criterion in the while loop can be one of the following: the iteration number exceeds some threshold, or the increase of the objective function becomes slow. Since the L1-norm of a matrix can be approximated by $\|\boldsymbol{M}\|_1\approx\sum_{i,j}\sqrt{m_{i,j}^2+\epsilon}$ (denoted as $\sum\sqrt{\boldsymbol{M}\cdot\boldsymbol{M}+\epsilon}$) where ϵ is a small constant and '$\cdot$' denotes element-wise product, we can compute the approximate gradient of Eq. (2) by the chain rule:

$$\frac{\partial\|\boldsymbol{W}\boldsymbol{X}\|_1}{\partial\boldsymbol{W}}\approx\sqrt{(\boldsymbol{W}\boldsymbol{X})\cdot(\boldsymbol{W}\boldsymbol{X})+\epsilon}\,\boldsymbol{W}\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}. \tag{5}$$

The gradient ascent step $\boldsymbol{W}_{\mathrm{new}}=\boldsymbol{W}_{\mathrm{old}}+\alpha\nabla\boldsymbol{W}$ is to increase the objective function, and the projection step $\boldsymbol{W}_{\mathrm{orth}}=\boldsymbol{W}_{\mathrm{new}}(\boldsymbol{W}_{\mathrm{new}}^{\mathrm{T}}\boldsymbol{W}_{\mathrm{new}})^{-\frac{1}{2}}$ is to satisfy the constraint.

Since Eq. (3) is a special case of Eq. (2) where the matrix $\boldsymbol{W}$ degenerates into a vector $\boldsymbol{w}$, each algorithm for Eq. (2) has its greedy version: solving Eq. (2) greedily via Algorithm 1, which requires solving Eq. (3). This also holds for Algorithm 2. Example 1 compares the solutions to Eq. (2) in the simplest case. In the following we will show results in more general situations.

4 Experimental studies

In this section, we show the experimental studies on the L1-PCA-VM problem. Specifically, we compare different solutions to Eq. (2) in both simulation and real-world datasets. 'pcal2' denotes conventional L2-norm PCA (as a baseline comparison); 'pcaKwak' denotes the greedy algorithm in Kwak (2008), or Algorithm 1; 'pcaNieN' and 'pcaNieG' denote the non-greedy algorithm and its greedy versions in Nie *et al.* (2011), respectively; and 'pcaGaN' and 'pcaGaG' denote the gradient ascent algorithm (Algorithm 2) in Section 3 and its greedy version, respectively.

Figs. 1 and 2 illustrate the estimated bases of different PCA methods from 3D data (d=3). The 'triangles' represents different greedy algorithms: 'pcal2', 'pcaKwak', 'pcaNieG', or 'pcaGaG'. Circles and squares denote the non-greedy 'pcaNieG' and 'pcaGaN', respectively. Fig. 1 shows the bases (directions in the 3D space) estimated from data $\boldsymbol{X}_{d\times n}$ (d=3, n=120) (circles) generated by a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$, where $\boldsymbol{\mu}=[0,0,0]^{\mathrm{T}}$, $\boldsymbol{\Sigma}=\mathrm{diag}\{9,1,1/9\}$. When m=1, the bases estimated by different methods are almost the same. When m=2, the bases estimated by different methods can be classified into two types. The first are bases estimated by greedy methods, being approximately $[0,0,1]$ and $[0,1,0]$, and corresponding to the principal axes in observations. The second are bases estimated by non-greedy methods,

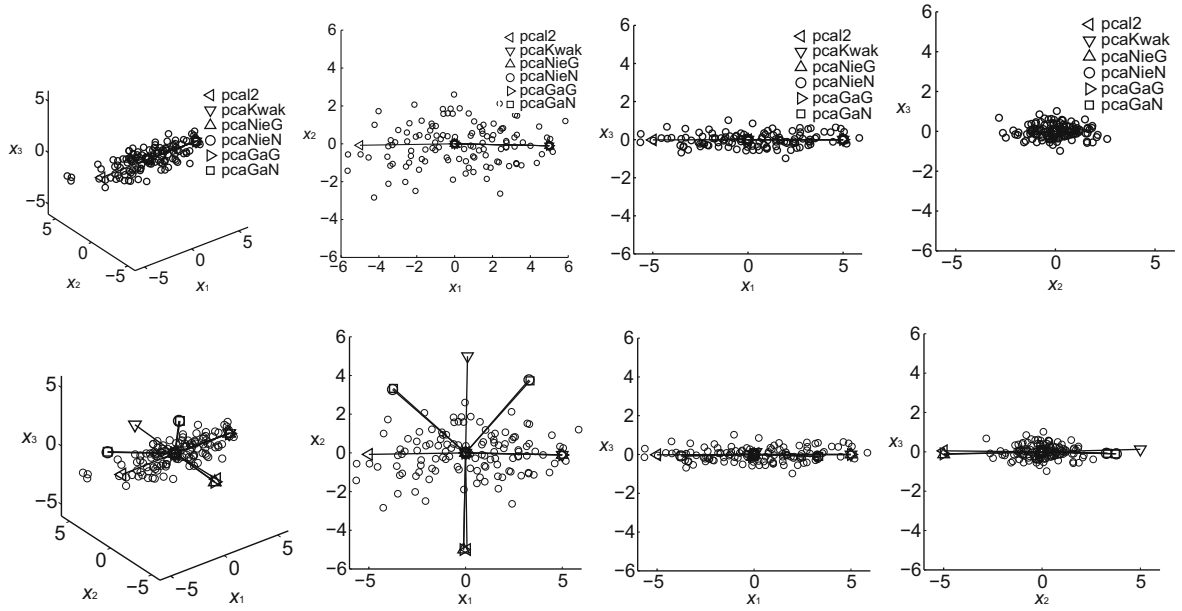

Fig. 1 Bases estimated by different PCA methods in case of 3D data (circles) without outliers. The top and bottom rows correspond to the m=1 and m=2 cases, respectively. The first column shows the bases in the 3D space, and columns 2–4 show the projection of samples and bases onto subspaces x_1-x_2, x_1-x_3, and x_2-x_3, respectively

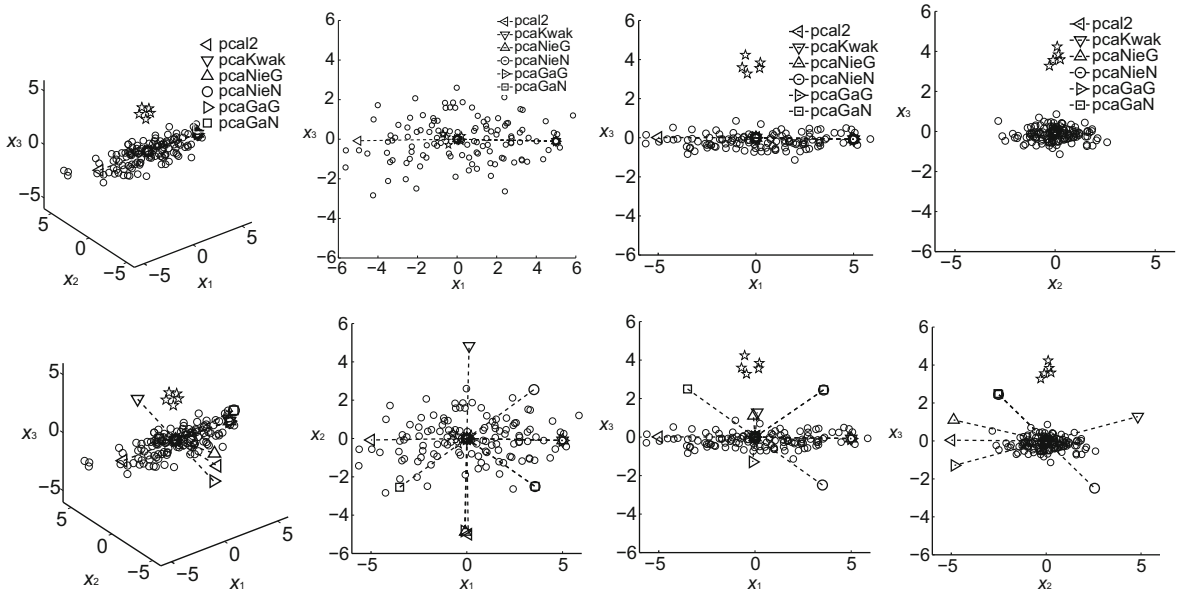

Fig. 2 Bases estimated by different PCA methods in case of 3D data (circles) with outliers (pentagrams). The top and bottom rows correspond to the $m = 1$ and $m = 2$ cases, respectively. The first column shows the bases in the 3D space, and columns 2–4 show the projection of samples and bases onto subspaces x_1-x_2, x_1-x_3, and x_2-x_3, respectively

being $\left[-\sqrt{2}/2, -\sqrt{2}/2, 0\right]$ and $\left[\sqrt{2}/2, -\sqrt{2}/2, 0\right]$ approximately, and corresponding to the principal spaces of observations. While the principal spaces estimated by non-greedy methods are no better than principal axes visually, both classes recover the 2D principal spaces: all bases lie approximately in the

same x_1-x_2 plane.

Fig. 2 shows the effects of five new outlier samples (pentagrams) generated from $\mathcal{N}(\boldsymbol{\mu}_o,\boldsymbol{\Sigma}_o)$ where $\boldsymbol{\mu}_o=[0,0,4]^T$, $\boldsymbol{\Sigma}_o$=diag{0.1,0.1,0.1}, which are used to 'distract' the estimated bases from principal spaces. When m=1, the bases estimated by different methods are almost the same. This corresponds to the case illustrated in Kwak (2008). When m=2, the bases estimated by different methods can also be classified into two types. The first are bases estimated by greedy methods, still being approximately $[0,0,1]$ and $[0,1,0]$, and corresponding to the principal axes in observations. Note that the bases are disturbed to the x_3 axis by the outliers in some sense. The second are bases estimated by non-greedy methods, now different from the $\left[-\sqrt{2}/2,-\sqrt{2}/2,0\right]$ and $\left[\sqrt{2}/2,-\sqrt{2}/2,0\right]$ in the previous case. Moreover, the bases are disturbed much more to the x_3 axis than bases from greedy methods, which are still approximately in the x_1-x_2 plane. Table 1 shows the objective function values achieved by different L1-PCA solutions. When m=2 and m=3, non-greedy methods ('pcaNieG' and 'pcaGaN') can achieve higher objective values than greedy methods ('pcal2', 'pcaKwak', 'pcaNieG', 'pcaGaG'), which shows that non-greedy solutions are better than greedy solutions under Eq. (2).

Integrating the results of Figs. 1 and 2 and Table 1, we see that Eq. (2) is problematic by analyzing its greedy and non-greedy solutions (similar to Example 1). All methods have similar results (base directions and objective values) when m=1. However, non-greedy methods always have higher objective values when m>1 (consistent with Nie *et al.* (2011)). When there are no outliers, all methods return similar principal spaces (the bases lie in approximately the same plane), but the principal axes learned by non-greedy methods are different from those learned by other greedy methods. When there are outliers, non-greedy methods return different bases which form different principal spaces from other greedy methods. Note that the 'true' principal space is the x_1-x_2 plane. Therefore, a better solution for Eq. (2) (evaluated by the objective function) corresponds to worse principal components, showing that Eq. (2) is problematic in learning principal components.

It is important to analyze Eq. (2) in real-world datasets. We apply standard datasets which are popular in pattern recognition tasks including PCA analysis (http://www.cad.zju.edu.cn/home/dengcai/). Table 2 shows the sample dimensionality, sample size, and the number of classes in each dataset.

Table 2 Real-world datasets

Dataset	Dimensionality	Size	Number of classes
Yale	1024	165	15
YaleB	1024	400	38
ORL	1024	2414	40
COIL100	1024	7200	100

Similar to previous examples, we identify the objective function values and bases by different solutions to Eq. (2) in these datasets. Furthermore, we compare the classification accuracies with different dimensionalities in these different PCA methods. To test the outlier effects, we replace 20% samples with random images: each pixel is sampled from $U(a,b)$ where a and b are the minimum and maximum values through all samples, respectively.

We find that the bases estimated by greedy and non-greedy methods for Eq. (2) are very different. For example, Figs. 3–6 visualize the bases estimated by different methods in datasets 'YaleB' and 'COIL100', respectively. Columns from left to right correspond to algorithms 'pcal2', 'pcaKwak', 'pcaNieG', and 'pcaNieN', respectively. The bases estimated by 'pcaGaG' and 'pcaGaN' are similar to those estimated by 'pcaNieG' and 'pcaNieN', respectively, and are not shown. It can be seen that

Table 1 Objective values of different PCA methods in 3D data (without and with outliers)

Outliers?	m	Objective value					
		'pcal2'	'pcaKwak'	'pcaNieG'	'pcaNieN'	'pcaGaG'	'pcaGaN'
No	1	275.0916	275.2542	275.2542	275.2542	275.2541	275.2541
	2	379.0217	379.4763	379.4763	**415.7465**	379.4827	**415.7742**
	3	412.3916	412.6970	412.6970	**513.2117**	412.7083	**510.3827**
Yes	1	277.2599	277.4191	277.4191	277.4191	277.4191	277.4191
	2	382.0930	384.1461	383.1320	**424.4690**	384.1522	**424.4687**
	3	438.2801	447.7581	444.6502	**543.1550**	443.3033	**544.0512**

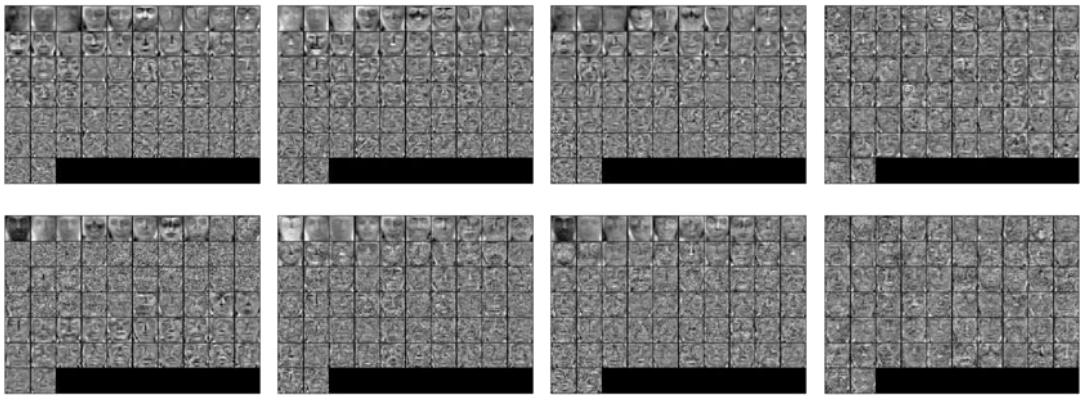

Fig. 3 Bases ($m = 62$) learned from the yale32 database without (top row) and with (bottom row) outliers by different methods (from left to right: 'pcal2', 'pcaKwak', 'pcaNieG', and 'pcaNieN')

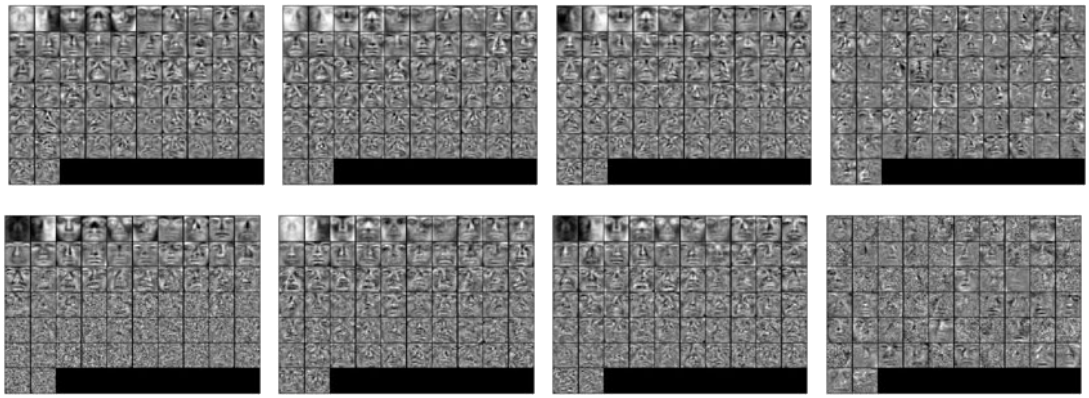

Fig. 4 Bases ($m = 62$) learned from the YaleB database without (top row) and with (bottom row) outliers by different methods (from left to right: 'pcal2', 'pcaKwak', 'pcaNieG', and 'pcaNieN')

greedy algorithms of Eq. (2) learn similar eigenfaces (second and third columns) to conventional PCA (first column), and are more robust to outliers. However, the non-greedy method (fourth column) returns very different bases and is more sensitive to outliers (comparing top and bottom rows).

Fig. 7 compares the classification accuracies and objective values of different PCA methods. Each dataset is randomly partitioned as 80% for training and 20% for testing. The accuracies and objective function values are averaged over 10 iterations using the nearest neighbor classifier. Outlier effects are compared: real lines denote results of raw data; dashed lines denote results of data with outliers (20% samples are replaced by random images with the same labels). It can be seen that non-greedy methods ('pcaNieN' and 'pcaGaN') always achieve higher objective values than greedy methods ('pcal2', 'pcaKwak', 'pcaNieG', and 'pcaGaG') but return no better performance in classification. Moreover, the classification accuracies in the YaleB database show that non-greedy methods perform poorly in comparison with greedy methods, which verifies the fact that the bases from greedy methods can be more useful and informative than non-greedy methods (see the visualized bases in both simulation and real-world datasets).

Experiments on synthetic and real-world datasets show that the L1-norm based PCA formulation (L1-PCA-VM) is problematic in learning principal components from data: non-greedy algorithms (L1-norm maximization algorithm (Nie *et al.*, 2011) and Algorithm 2 in this paper) can achieve higher objective values, but greedy algorithms (e.g., the PCA-L1 algorithm (Kwak, 2008), greedy versions of the L1-norm maximization algorithm (Nie *et al.*, 2011) and Algorithm 2 in this paper) achieve more interpretable solutions (more

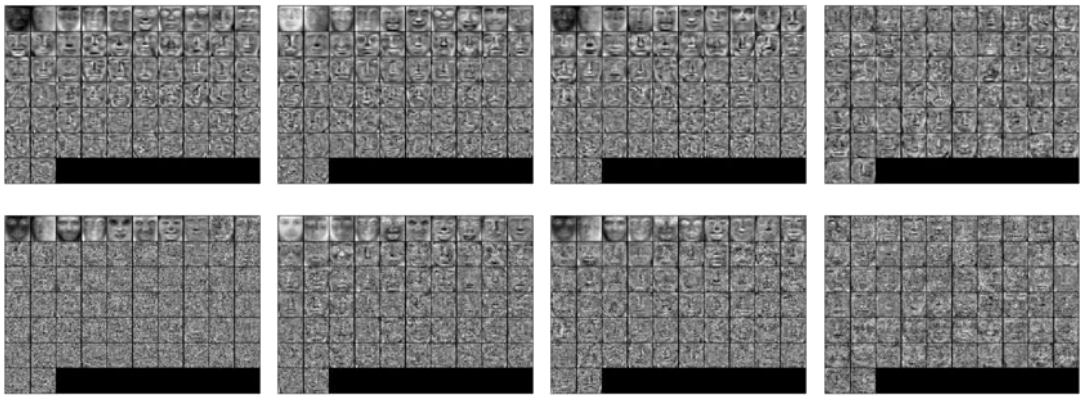

Fig. 5 Bases ($m = 62$) learned from the orl32 database without (top row) and with (bottom row) outliers by different methods (from left to right: 'pcal2', 'pcaKwak', 'pcaNieG', and 'pcaNieN')

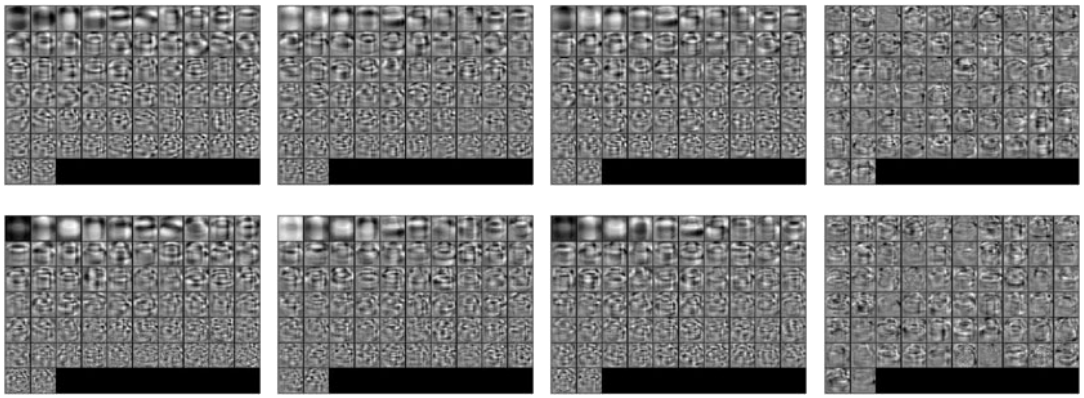

Fig. 6 Bases ($m = 62$) learned from the COIL100 database without (top row) and with (bottom row) outliers by different methods (from left to right: 'pcal2', 'pcaKwak', 'pcaNieG', and 'pcaNieN')

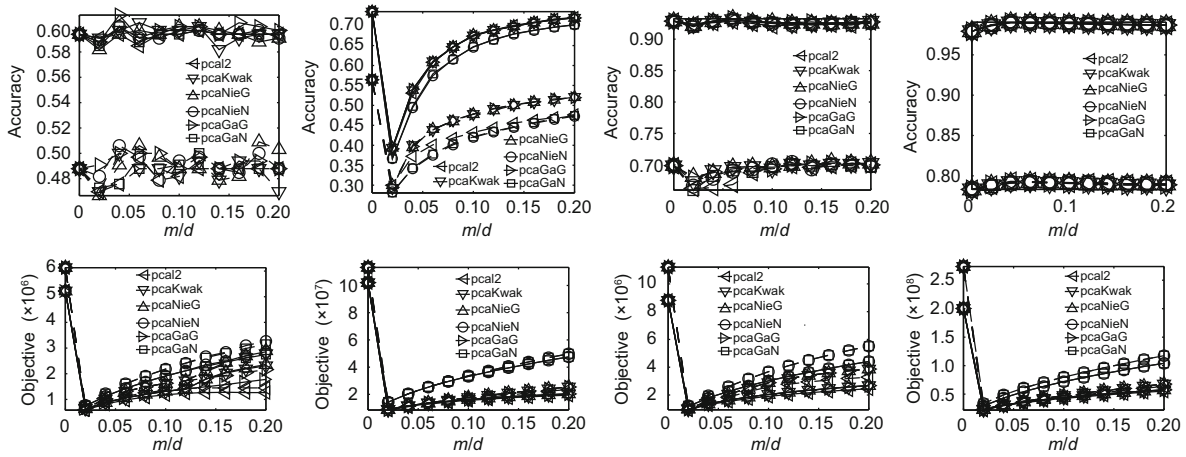

Fig. 7 Classification accuracies (top row) and objective function values (bottom row) of different principal component analysis (PCA) methods in real-world datasets (from left to right: 'yale32', 'yaleb32', 'orl32', and 'coil100') with different reduced dimensionalities (m/d=0, 0.02, 0.04, ..., 0.20, where $m = 0$ means using the raw data for baseline comparison $W = I$)

informative principal components), which is verified by basis visualization and classification performances with and without outlier disturbances. Our interpretation for this phenomenon is that L2- and L1-norm have different properties: L2-norm is rotationally invariant, which means the principal axes and their rotated versions are equivalent in the L2-norm objective function; the L1-norm objective function is variant to rotations of principal axes. Moreover, the solutions from non-greedy algorithms corresponding to larger objective functions are different from those principal axes. With greedy algorithms, the L1-PCA-VM can learn similar bases to traditional L2-norm formulation, and can achieve better solutions when data contains outliers, since greedy algorithms identify the principal axes one by one from updated data, and L1-norm can suppress the outliers compared with L2-norm in this situation.

5 Conclusions

This paper compares different methods for the variance maximization formulation of PCA with L1-norm, and shows that the formulation itself can be problematic in learning principal components and dealing with outliers. A new gradient ascent based method is proposed to facilitate analysis and comparison. We find that the non-greedy algorithm in previous work is almost equivalent to our method: they achieve similar objective function values and return similar principal components. Experimental results in both synthetic and real-world data show that only greedy solutions to L1-PCA-VM can be reasonable in learning better principal components, while non-greedy solutions learn different bases from conventional L2-PCA, and can be sensitive in situations with outliers. We also observe that the non-greedy solutions can extract useful information for classification in some situations, even if the estimated bases are not visually informative.

References

Bishop, C.M., 2006. Pattern Recognition and Machine Learning. Springer.

de la Torre, F., Black, M.J., 2001. Robust Principal Component Analysis for Computer Vision. Proc. 8th IEEE Int. Conf. on Computer Vision, **1**:362-369.

Ding, C., Zhou, D., He, X.F., Zha, H.Y., 2006. R1-PCA: Rotational Invariant L1-Norm Principal Component Analysis for Robust Subspace Factorization. Proc. 23rd Int. Conf. on Machine Learning, p.281-288. [doi:10.1145/1143844.1143880]

Duda, R.O., Hart, P.E., Stork, D.G., 2001. Pattern Classification. Wiley-Interscience.

Hastie, T., Tibshirani, R., Friedman, J., Franklin, J., 2005. The Elements of Statistical Learning: Data Mining, Inference and Prediction. Springer.

Hyvärinen, A., Hurri, J., Hoyer, P.O., 2009. Natural Image Statistics. Springer.

Ke, Q.F., Kanade, T., 2003. Robust Subspace Computation Using L1 Norm. Technical Report, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA.

Ke, Q.F., Kanade, T., 2005. Robust L1 Norm Factorization in the Presence of Outliers and Missing Data by Alternative Convex Programming. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, p.592-599.

Kwak, N., 2008. Principal component analysis based on L1-norm maximization. *IEEE Trans. Pattern Anal. Mach. Intell.*, **30**(9):1672-1680. [doi:10.1109/TPAMI.2008.114]

Le, Q.V., Karpenko, A., Ngiam, J., Ng, A.Y., 2011. ICA with Reconstruction Cost for Efficient Overcomplete Feature Learning. Advances in Neural Information Processing Systems 24, p.1017-1025.

Liu, J., Ji, S., Ye, J., 2009. Multi-task Feature Learning via Efficient L2,1-Norm Minimization. Proc. 25th Conf. on Uncertainty in Artificial Intelligence, p.339-348.

Nakajima, S., Sugiyama, M., Babacan, D., 2011. On Bayesian PCA: Automatic Dimensionality Selection and Analytic Solution. Proc. 28th Int. Conf. on Machine Learning, p.497-504.

Nie, F., Huang, H., Cai, X., Ding, C., 2010. Efficient and Robust Feature Selection via Joint L2,1-Norms Minimization. Advances in Neural Information Processing Systems 23, p.1813-1821.

Nie, F., Huang, H., Ding, C., Luo, D., Wang, H., 2011. Robust Principal Component Analysis with Non-greedy L1-Norm Maximization. Proc. 22nd Int. Joint Conf. on Artificial Intelligence, p.1433-1438.

Wright, J., Ganesh, A., Rao, S., Peng, Y.G., Ma, Y., 2009. Robust Principal Component Analysis: Exact Recovery of Corrupted Low-Rank Matrices via Convex Optimization. Advances in Neural Information Processing Systems 22, p.2080-2088.

Zass, R., Shashua, A., 2007. Nonnegative Sparse PCA. Advances in Neural Information Processing Systems 19, p.1561-1568.

Zhang, Y., Teng, Y., 2010. Adaptive multiblock kernel principal component analysis for monitoring complex industrial processes. *J. Zhejiang Univ.-Sci. C (Comput. & Electron.)*, **11**(12):948-955. [doi:10.1631/jzus.C1000148]