Some Extension of Sparse Principal Component Analysis

Thanh D. X. Duong and Hung V. Nguyen

***Abstract*—Given a covariance matrix, sparse principal component analysis (SPCA) considers the problem of maximizing the variance explained by a particular linear combination of the input variables where the number of nonzero coefficients is constrained. In some applications, the coefficients in this combination are required to be non-negative. Moreover, when loading an input variable is associated an individual cost, we need incorporate weights, which represent the loading cost of input variables, into sparsity constraint. And in this paper, we consider problems of SPCA with weighted sparsity constraint and/or non-negative sparsity constraint. These problems are reduced to solving some semi-definite programming ones via convex relaxation technique. Numerical results show that the method is efficient and reliable in practice.**

***Index Terms*—Iterative re-weighting, non-negative constraint principal component analysis, principal component analysis, semi-definite relaxation, sparse principal component analysis.**

I. Introduction

Sparse decompositions of data are required in many applications. In economics, sparsity increases the efficiency and reduces risk of a portfolio [7], it also implies lower transaction cost in financial asset trading strategies. In computer vision, sparse decomposition is related to the extraction of some concerned pixels which are relevant parts from images [17]. In machine learning, sparsity is closely related to feature selection and to improved generalization of learning algorithms. And in biology, the sparsity is necessary for finding focalized local patterns hidden in gene expression data analysis [1].

Being first introduced by Pearson in [28], and developed independently by Hotelling in [6], principal component analysis (PCA) has now become a popular technique used to reduce multidimensional data sets to lower dimensions for analysis with applications throughout science and engineering, see [20]. This reduction is achieved by transforming to a new set of variables, the principal components, which are uncorrelated and ordered so that the first few retain most of the variation present in all of the original variables. It also can be performed via a singular value decomposition of the data matrix or an eigenvalue decomposition of the data covariance matrix.

A drawback of PCA is the lack of sparseness of the principal vectors - since the principal components are usually linear combinations of all variables and the loadings are typically non-zero - while sparse decompositions of data are required in many applications such as: sparsity increases the efficiency and reduces risk of a portfolio [7]; sparse decomposition is related to the extraction of some concerned pixels which are relevant parts from images [17]; sparsity is closely related to feature selection and to improved generalization of learning algorithms; sparsity is necessary for finding focalized local patterns hidden in gene expression data analysis [1]. This leads to appearance of several methods to find sparse principal components explaining most of the variance present in the data. To achieve this, it is necessary to sacrifice some of the explained variance and the orthogonality of the principal components. Rotation techniques [14] can be considered the first approach. Simple principal components [13] are studied by restricting the loadings to take values from a small set of allowable integers such as 0, 1 and -1. Simple thresholding technique [2] was an ad hoc way to deal with the problem, where the loadings with small absolute value are thresholded to zero. SCoTLASS [9] and SLRA [10], [11] were introduced to get modified principal components with possible zero loadings. ESPCA [21] used discrete spectral formulation based on variational eigenvalue bounds and an effective greedy strategy to give provably optimal solutions via branch-and-bound search. SPCA [12] was proposed via a regression type optimization problem. And DSPCA [4], [16] is the current state-of-the-art method, which relaxes a hard cardinality constraint with a convex approximation. Recently, [26] discusses several options, comparing the variance vs. orthogonality/sparsity tradeoffs they imply. Finally, [30] shows that some naive approaches have the significantly worst convergence rates than the relaxation approach as in [4], [5], [16], [22].

Since the outputs of DSPCA [16] are not satisfied sparsity constraint - i.e. if we hope to find a principal component with less than k non-zero entries, the output often contains more than k non-zero entries - it is reasonable to add a post-processing technique to enhance the sparsity of the principal components. Recently, re-weighted l_1 minimization [3], [19] is a useful technique to enhancing sparsity output of the combinatorial optimization: $\min \|x\|_{l_0}$ subject to $y = \Phi x$,
where $\|x\|_{l_0} = |\{i : x_i \neq 0\}|$. Replacing the linear equation constraints in the above combinatorial optimization by linear matrix inequality constraints, we get an approach to refine the sparsity of the principal components. This approach has been done in [22], where we add non-negative property to the principal components.

In practice, there are many applications in which loading an input variable is associated an individual cost. In these cases, it is essential to incorporate weights, which represent the loading cost of input variables, into PCA. Weighted PCA

Manuscript received July 15, 2012; revised September 10, 2012.

Thanh D. X. Duong is with the John von Neumann Institute, VNU-HCM, Vietnam (e-mail: thanh.duong@jvn.edu.vn).

Nguyen V. Hung is with Japan Advanced Institute of Science and Technology, Japan (e-mail: nvhung@jaist.ac.jp).

10.7763/IJMLC.2012.V2.218

has thus been introduced and used in many applications such as learning from incomplete data in [29], giving an efficient search algorithm for motion data in [24], and face recognition in [27]. To solve this problem, we generalize the sparsity constraint with weighted sparsity constraint [5].

It is remarkable that we cannot obtain the desired sparsity of the output by using the re-weighted l_1 minimization technique [3] even with linear equation constraints. To overcome this obstacle, we present a bisection algorithm exploiting truncation technique to find the principal component not only satisfying the desired sparsity constraint but also explaining the most variance [23].

In practice, the loadings of principal components are required to be non-negative and/or associated an individual cost. In this paper, we consider problems of SPCA with weighted sparsity constraint and/or non-negative sparsity constraint. This paper is organized as follows. The next section is the main results, where we presents a method which directly incorporates a weighted (and/or non-negative) sparsity criterion in the PCA problem formulation, then apply relaxation method and a bisection algorithm to find the best principle components which exactly satisfy the desired sparsity constraints. Section 3 is devoted to compare the proposed method with existing methods on both artificial data and real-life data.

Notation. In this paper, we denote the set of symmetric matrices of size n by $\mathbf{S}^n$, the vector of ones by $\mathbf{1}$, the cardinality (number of non-zero elements) of a vector x by $\mathbf{Card}(x)$, and the number of non-zero coefficients in a matrix X by $\mathbf{Card}(X)$. For $X \in \mathbf{S}^n$, the notation $X \succeq 0$ means that X is positive semi-definite, $|X|$ is the matrix whose elements are the absolute values of the elements of X, and $\mathbf{Tr}(X) := X_{11} + X_{22} + ... + X_{nn}$, which is the sum of diagonal entries.

II. Main Results

In this section, we review semi-definite relaxation method [4], [5], [16], [22], [23] for SPCA with weighted sparsity constraint and/or non-negative sparsity constraint, refine the cardinality constraint via re-weighted l_1 minimization technique, and present a bisection algorithm to find the principle component which satisfies the required sparsity constraint when sacrifice some of the explained variance. Then, we apply the problem to decompose a data matrix into sparse principle components.

A. Semi-definite Relaxation

Let $A \in \mathbf{S}^n$ be a covariance matrix, i.e. $A \succeq 0$, and $w \in R^n$ be a weight vector with $w_i > 0$ for all $i = 1,...,n$. We consider a problem of maximizing the variance of vector $x \in R^n$ while constraining its weighted cardinality (and/or non-negative loadings):

$$\begin{aligned} &\text{maximize } x^T Ax, \\ &\text{subject to } \|x\|_2 = 1, \\ &\sum_{i=1}^{n} w_i \delta(x_i) \le k, (x \ge 0) \end{aligned} \tag{1}$$

where $\delta : R \longrightarrow 0,1$ is defined by $\delta(x) = \begin{cases} 0, & \text{if } x = 0, \\ 1, & \text{if } x \neq 0, \end{cases}$

and the given positive number k restricts the number of non-zero entries of the solution, thus the following inequality should be hold: $\min_{i=1,...,n} w_i \le k \le \sum_{i=1}^{n} w_i.$

Hence, by choosing $w = 1$, the weighted sparsity constraint $\sum_{i=1}^{n} w_i \delta(x_i) \le k$ collapses to the classical sparsity constraint $\mathbf{Card}(x) \le k$. In the case of non-negative property is not required for the loadings, we can remove the last constraint, $x \ge 0$.

Using semi-definite relaxation techniques [5], the problem is approximated by the following problem:

$$\begin{aligned} &\text{maximize } \mathrm{Tr}(AX), \\ &\text{subject to } \mathrm{Tr}(X) = 1, \\ &\sum_{i,j=1}^{n} \sqrt{w_i w_j} \left| X_{ij} \right| \le k, \\ &X \succeq 0, (X \ge 0). \end{aligned} \tag{2}$$

This means, we will solve the semi-definite problem (2) to get solution X, and an approximation solution of (1) is the dominant eigenvector of X (or the non-negative parts of the dominant eigenvector of X if non-negative property is required).

B. Cardinality Constraint Refinement

Let x_* be the approximation solution of (1). It is clear that x_* does not satisfying cardinality constraint $\mathbf{Card}(x) \le k$ in general. Hence, we consider the following cardinality constraint refinement problem:

$$\begin{aligned} &\text{minimize } \sum_{i=1}^{n} w_i \delta(x_i), \\ &\text{subject to } \|x\|_2 = 1, \\ &x^T Ax \ge c_*, (x \ge 0). \end{aligned} \tag{3}$$

where $c_* = x_*^T A x_*$.

Let $X = xx^T \succeq 0$, the second order constraint $x^T Ax \ge c_*$ and the non-convex constraint $\|x\|_2 = 1$ are transformed into linear constraints since $\mathbf{Tr}(AX) = x^T Ax$ and $\mathbf{Tr}(X) = \|x\|_2$. Hence, using Lemma 2.1 in [5], the lifting procedure [1], [8], [16], [25] for semi-definite relaxation gives an equivalent problem of (3) as follows:

$$\begin{aligned} &\text{minimize } \sum_{i,j=1}^{n} w_i w_j \delta(X_{ij}), \\ &\text{subject to } \mathrm{Tr}(X) = 1, \\ &\mathrm{Tr}(AX) \ge c_* \\ &X \succeq 0, (X \ge 0), rank(X) = 1 \end{aligned} \tag{4}$$

Next, apply truncation technique [1], [25] which drops the constraint $\mathbf{rank}(X) = 1$, we can relax the problem (6) as follows:

$$\begin{aligned} & \text{minimize} \sum_{i,j=1}^{n} w_i w_j \delta(X_{ij}), \\ & \text{subject to } \mathrm{Tr}(X) = 1, \\ & \mathrm{Tr}(AX) \geq c_* \\ & X \succeq 0, (X \geq 0). \end{aligned} \tag{5}$$

Using the re-weighted l_1 minimization which is recent technique to enhancing sparsity, see [3], we consider the following relaxation of the problem (5):

$$\begin{aligned} & \text{minimize } \mathrm{Tr}\left(W^T |X|\right), \\ & \text{subject to } \mathrm{Tr}(X) = 1, \\ & \mathrm{Tr}(AX) \geq c_* \\ & X \succeq 0, (X \geq 0). \end{aligned} \tag{6}$$

where $W > 0$ is positive weight matrix. We use a simple iterative algorithm that alternates between estimating X and redefining the weights [3] to give an extension of DSPCA (called EDSPCA in what follows):

1. Set the iteration count m to zero and solve DSPCA problem (4) to get the solution $X^{(0)}$.
2. Update the weights:

$$\text{for each } i, j = \overline{1,n} \;\; w_{ij}^{(m)} = \frac{1}{X_{ij}^{(m)} + \varepsilon}. \tag{7}$$

3. Solve the weighted problem (6) to get the solution $X^{(m)}$.
4. Terminate when $\mathbf{Card}(x^{(m)}) \leq k$ or m attains a specified maximum number of iterations m_{max}, where $x^{(m)}$ is the dominant eigenvector of $X^{(m)}$. Otherwise, increment m and go to step 2.

The semi-definite programming (SDP) problem (2) and (6) can be solved efficiently using interior-point solvers such as SEDUMI [31] or SDPT3 [32]. And we should set a *threshold* for expected non-zero-valued component X_{ij} of the solution of the problems (2) and (6). Moreover, the parameter $\varepsilon > 0$ in (7) should be chosen as $\varepsilon \leq \textit{threshold}$ to provide stability and to ensure that a zero-valued component in X_{ij} does not strictly prohibit a non-zero estimate at the next step.

C. A Bisection Algorithm

It is important that the re-weighted l_1 minimization iteration try to refine the sparsity of the principal component as most as possible, but do not guarantee that the principal component satisfies the desired sparsity constraint. To achieve this, it is necessary to sacrifice some of the explained variance, albeit hopefully not too much. By dropping the entries which have the smallest absolute values, we obtain a truncated principal component which gives a lower bound for the explained variance. Exploiting this observation, we get a bisection algorithm to find the optimal principle component x_* satisfying the required sparsity with a given error $var_{error} > 0$ in the explained variance as follows:

initial Run the cardinality constraint refinement process to get approximation solution x. Set

- the optimal principle component $x_* := x$,
- the upper bound for the explained variance $var_{up} := x_*^T A x_*$,
- the truncated principal component entries $xt_i := x_{*i}$ for all $i \in I$ and $xt_i := 0$ for all $i \notin I$ - where I is the set of indexes of k absolutely largest entries of x_*,
- the lower bound for the explained variance $var_{low} := xt^T Axt$.

while $(var_{up} - var_{low}) > var_{error}$

- set $var_{new} := 1/2(var_{up} + var_{low})$
- run the cardinality constraint refinement process with $c_* := var_{new}$ to get solution x_*
- **if** $\mathbf{Card}(x_*) \leq k$ **then** update $var_{low} := var_{new}$
  else
  + update $var_{up} := var_{new}$
  + $x_* :=$ truncation of x_*

return the optimal principle component x_*.

III. Numerical Experiments

In this section, we will compare the effectiveness of EDSPCA with the other methods mentioned in the introduction. We perform the test on an artificial data set proposed by H. Zou, T. Hastie, R. Tibshirani [12] and a well-known real-life benchmark data set - Pit Props data.

A. Artificial Data

To show the effectiveness of ESDPCA, we consider the simulation example proposed by H. Zou, T. Hastie, R. Tibshirani [12], which was used to test the other methods. In this example, three hidden factors are first created

$$V_1 \sim N(0,290); V_2 \sim N(0,300); V_3 = 0.3V_1 + 0.925V_2 + \varepsilon$$

$\varepsilon_1 \sim N(0,1)$; V_1, V_2 and ε are independent

Then 10 observed variables are generated as the follows

$$X_i = V_1 + \varepsilon_i^1,\; \varepsilon_i^1 \sim N(0,1), i = 1,2,3,4,$$

$$X_i = V_2 + \varepsilon_i^2,\; \varepsilon_i^2 \sim N(0,1), i = 5,6,7,8,$$

$$X_i = V_3 + \varepsilon_i^3,\; \varepsilon_i^3 \sim N(0,1), i = 9,10,$$

ε_i^j are independent, j=1,2,3; i=1,...,10.

To avoid the simulation randomness, the exact covariance matrix which is an infinity amount of data generated from the above model is used to compute principal components using the different approaches. The variance of the three underlying factors is nearly the same (290, 300 and 283.8, respectively). Since the first two are associated with four variables while the last one is associated with only two variables, V_1 and $\mathbf{V_2}$ are almost equally important, and they are both significantly more important than V_3. In [12], the first two principal components explain 99.6% of the total variance. In [16], by choosing the sparsity constraint m=4, DSPCA gives the same results as SPCA and SCoTLASS which are better than simple thresholding method. Moreover, the output of DSPCA also satisfy the the sparsity constraint m=4 (having 4 non-zero entries). Thus, the sparsity constraint refinement post-processing of DSPCA is not required.

Now, we consider the results of DSPCA when choosing the sparsity constraint k=5 in Table 1. The outputs of DSPCA do not satisfy the sparsity constraint when both the first and second principal components have 6 non-zero entries. Hence, the sparsity constraint refinement post-processing is needed. With the same explained variance, the first two principal component of EDSPCA (with threshold = 10^{-2}, $\varepsilon = 10^{-2}$, $var_{error} = 0.5\%$) satisfies the the sparsity constraint after solving 21 and 20 SDP problems, respectively.

B. Pit Props Data

The pit props data (consisting of 180 observations and 13 measured variables) was introduced in [18] and is another benchmark example used to test SPCA. All simple thresholding [2], SCoTLASS [9], SPCA [12], and DSPCA [16] have been tested on this data set. As reported in [12], SPCA performs better than SCoTLASS in the sense that it identifies principal components with 7, 4, 4, 1, 1, and 1 non-zero loadings respectively - while explaining nearly the same variance as SCoTLASS, the result SPCA of is much sparser; and better than simple thresholding in the sense that it explains more variance. As reported in [16], DSPCA performs better than SPCA in the sense that it identifies principal components with 6, 2, 3, 1, 1, and 1 non-zero loadings (with respect to sparsity constraint 5, 2, 2, 1, 1, and 1).

Here, we want to compare the results of EDSPCA - using the same sparsity constraint (5, 2, 2, 1, 1, and 1) - with those of DSPCA. The results are given in Table 2 with threshold = 10^{-2}, $\varepsilon = 10^{-10}$, $var_{error} = (0.5/(2^i-1))\%$ for i=1,…,6. While explaining 76.0% variance - nearly the same as DSPCA (77.3%) - the first six principal components of EDSPCA satisfies the sparsity constraint after solving 5, 1, 16, 1, 1, and 1 SDP problems respectively. It is also remarkable that these results are better than ESPCA (75.9%) in [21]. However, we can see that there is an overlap between the first principal component and the third principal component on entry "ringbut". Hence, it is reasonable to think about a better.

TABLE I: THE FIRST TWO PRINCIPAL COMPONENTS OF DSPCA AND EDSPCA WITH K=5 ON THE ARTIFICIAL DATA

Variable	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	Variance
DSPCA, PC_1	0	0	0	0	.49	.49	.49	.49	.14	.14	50.2%
DSPCA, PC_2	-.49	-.49	-.49	-.49	0	0	0	0	.14	.14	41.9%
EDSPCA, PC_1	0	0	0	0	.46	.46	.46	.46	.40	0	49.7%
EDSPCA, PC_2	.49	.49	.49	.49	0	0	0	0	0	-.15	40.6%

TABLE II: THE FIRST THREE PRINCIPAL COMPONENTS OF DSPCA AND EDSPCA WITH SPARSITY CONSTRAINT 5,2,2,1,1 AND 1 ON THE PIT PROPS DATA

Methods	DSPCA			EDSPCA		
Variable	PC_1	PC_2	PC_3	PC_1	PC_2	PC_3
topdiam	-0.56	0	0	-0.48	0	0
length	-0.58	0	0	-0.49	0	0
moist	0	0.71	0	0	0.71	0
testsg	0	0.71	0	0	0.71	0
ovensg	0	0	0	0	0	0
ringtop	0	0	-0.79	0	0	-0.82
ringbut	-0.26	0	-0.61	-0.40	0	-0.58
bowmax	-0.1	0	0	0	0	0
bowdist	-0.37	0	-0.42	0	0	0
whorls	-0.36	0	0	-0.43	0	0
clear	0	0	0	0	0	0
knots	0	0	0	0	0	0
diaknot	0	0	0.01	0	0	0
Variance	26.6	14.48	13.15	26.2	14.48	12.24
Card	6	2	3	5	2	2

TABLE III: THE FIRST THREE PRINCIPAL COMPONENTS OF EDSPCA WITH SPARSITY CONSTRAINT (4,2 AND 2) AND (5,3 AND 2).

Sparsity	4-2-2			5-2-3		
Variable	PC_1	PC_2	PC_3	PC_1	PC_2	PC_3
topdiam	-0.54	0	0	0.48	0	0
length	-0.55	0	0	0.49	0	0
moist	0	-0.71	0	0	-0.71	0
testsg	0	-0.71	0	0	-0.71	0
ovensg	0	0	0	0	0	0
ringtop	0	0	0.71	0	0	0.68
ringbut	0	-0	0.71	0.42	0	0.54
bowmax	0	0	0	0	0	0
bowdist	-0.47	0	0	0	0	0
whorls	-0.44	0	0	0	0	0
clear	0	0	0	0	0	0
knots	0	0	0	0	0	0
diaknot	0	0	0	0	0	-0.51
Variance	22.78	14.48	13.94	26.2	14.48	14.7

Here, we want to compare the results of EDSPCA - using the same sparsity constraint (5, 2, 2, 1, 1, and 1) - with those of DSPCA. The results are given in Table 2 with threshold = 10^{-2}, $\varepsilon = 10^{-10}$, $var_{error} = (0.5/(2^i-1))\%$ for i=1,…,6. While explaining 76.0% variance - nearly the same as DSPCA (77.3%) - the first six principal components of EDSPCA satisfies the sparsity constraint after solving 5, 1, 16, 1, 1, and 1 SDP problems respectively. It is also remarkable that these results are better than ESPCA (75.9%) in [21]. However, we can see that there is an overlap between the first principal component and the third principal component on entry "ringbut". Hence, it is reasonable to think about a better sparsity constraint as 4, 2, 2, 1, 1, and 1. The outputs for this case are displayed in the left haft of Table 3, where EDSPCA also explains a large amount of the variance - 74.3% - by solving 12, 1, 1, 1, 1, and 1 SDP problems respectively. Finally, with the less sparsity results (5, 2, 3, 1, 1, and 1) than DSPCA , the results of EDSPCA in the right haft of Table 3 explains more variance than DSPCA (78.5% compared with 77.3%). The first six principal components of EDSPCA satisfy the sparsity constraint after solving 5, 1, 15, 1, 1, and 1 SDP problems respectively. Figure 1 shows the cumulative number of non-zero loadings and the cumulative explained variance of EDSPCA compared with DSPCA and SPCA.

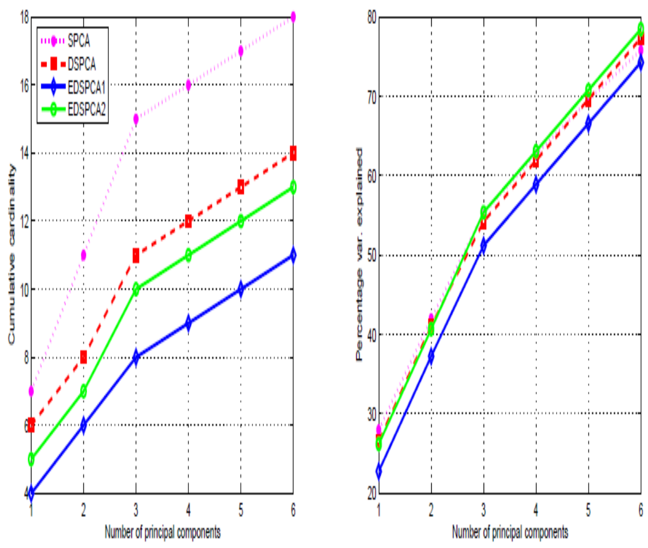

Fig.1.Cumulative cardinality and percentage of total variance explained versus number of principal components for SPCA, DSPCA and EDSPCA with sparsity constraint (4,2,2,1,1 and 1) and (5,2,3,1,1) on the pit props data.

IV. CONCLUSIONS AND PERSPECTIVE

The application specific solution will be discussed elsewhere since we want to keep our method general. By re-weighted l_1 minimization technique and a bisection algorithm, EDSPCA attempted to add an post processing to DSPCA method to find the principal components satisfying sparsity constraints. The drawback of the method is that the SDP problems involved in (2) and (6) contain more than $O(n^2)$ constraints, which make the memory requirements of Newton's method prohibitive for very large-scale problems. This should be the subject of a future investigation by using smoothing technique, which has recently shown to be reducing memory requirements in solving large-scale SDP problems, see [15], [16]. Finally, finding an efficient re-weighted function in (7) is also one of our priorities.

ACKNOWLEDGMENT

The authors thank University of Pedagogy, Ho Chi Minh city, Viet Nam for financial support.

REFERENCES

[1] F. Alizadeh, "Interior point methods in semi-definite programming with applications to combinatorial optimization," *SIAM Journal Optimization,* vol.5, pp.13-51, 1995.

[2] J. Cadima, I. T. Jolliffe, "Loadings and correlations in the interpretation of principal components," *Journal of Applied Statistics,* vol. 22, no. 2, pp. 203-214, 1995.

[3] E. J. Candes, M. B. Wakin, S. Boyd, "Enhancing sparsity by re-weighted l_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877-905, 2008.

[4] A.D'Aspremont, F. Bach, L.El Ghaoui, "Optimal solutions for sparse principal component analysis," *Journal of Machine Learning Research,* vol. 9, pp. 1269-1294, 2008.

[5] D. X. T.Duong, V. Duong, "Principal component analysis with weighted sparsity constraint," *Applied Mathematics and Information Sciences,* vol. 4, pp. 79-91, 2010.

[6] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *Journal of Education Psychology,* vol. 24, no. 7, pp. 417-441, 1993.

[7] R. Jagannathan and T. Ma, "Risk reduction in large portfolios: Why imposing the wrong constraints helps," *Journal of Finance,* vol. 58, pp. 1651-1684, 2003.

[8] L. Lovasz, A. Schrijver, "Cones of matrices and set-functions and 0-1 optimization," *SIAM Journal on Optimization,* vol. 1, pp. 166-190, 1991.

[9] I. T. Jolliffe, N. T. Trendafilov, M. Uddin, "A modified principal component technique based on the LASSO," *Journal of Computational and Graphical Statistics*, vol. 12, no. 3, pp. 531-547, 2003.

[10] Z. Zhang, H. Zha, H. Simon, "Low-rank approximations with sparse factors I: Basic algorithms and error analysis," *SIAM Journal on Matrix Analysis and Application*, vol. 23, no. 3, pp. 706-727, 2002.

[11] Z. Zhang, H. Zha, H. Simon, "Low-rank approximations with sparse factors II: Penalized methods with discrete Newton-like iterations," *SIAM Journal on Matrix Analysis and Application,* vol. 25, no. 4, pp. 901-920, 2004.

[12] H. Zou,T. Hastie, R. Tibshirani, "Sparse principal component analysis.," *Journal of Computation and Graphical Statistics,* vol. 15, no. 2, pp. 265-286, 2006.

[13] S. Vines, "Simple principal components," *Journal of the Royal Statistical Society: Serries C (Applied Statistics)*, vol. 49, no. 4, pp. 441-451, 2000.

[14] I. T. Jolliffe, I.T, "Rotation of principal components: Choice of normalization constraints," *Journal of Applied Statistics,* vol. 22, no. 1, pp. 29-35, 1995.

[15] Nesterov, Y, "Smoothing technique and its application in semi-definite optimization," *Mathematical Programming,* vol. 110, pp. 245-259, 2007.

[16] A. D'Aspremont, L. El Ghaoui, M.I.Jordan, G. R. G. Lanckriet, "A direct formulation for sparse PCA using semi-definite programming," *SIAM Review,* vol. 49, no. 3, pp. 434-448, 2007.

[17] D. D, Lee, H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature,* vol. 401, pp. 788-791, 1999.

[18] J. Jeffers, "Two case studies in the application of principal components," *Applied Statistics,* vol. 6, pp. 225-236, 1967.

[19] S. Boyd and L.Vandenberghe, *Convex optimization,* Cambridge University Press. Cambridge UK, 2004.

[20] I. T. Jolliffe, "*Principal component analysis*," New York: Springer Verlag, 2002.

[21] B. Moghaddam, Y. Weiss, S. Avidan, "Spectral Bounds for Sparse PCA: Exact & Greedy Algorithms," *Advances in Neural Information Processing Systems,* MIT Press, 2006, pp. 915-922.

[22] D. X. T.Duong and V. Duong, "Non-Negative Sparse Principal Component Analysis for Multidimensional Constrained Optimization," *The Tenth Pacific Rim International Conference on Artificial Intelligence (PRICAI 08*), Springer-Verlag Berlin Heidelberg, 2008, pp. 103-114.

[23] D. X. T. Duong, V. Duong, "Exact and Direct Approach for Sparse Principal Component Analysis," *The Second International Conference on Theories and Applications of Computer Science -ICTACS*, 2009.

[24] K. Forbes and E. Fiume, "An efficient search algorithm for motion data using weighted PCA," in *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ACM Press, New York.2005, pp. 67-76.

[25] C. Lemarechal and F. Oustry, "Semi-definite relaxations and Lagrangian duality with application to combinatorial optimization," Rapport de recherche 3710, INRIA, France, 1999.

[26] L. Mackey, "Deflation methods for sparse PCA," Advances in Neural Information Processing Systems, pp. 1017-1024, 2009.

[27] V. D. M. Nhat and S. Lee, "Two-dimensional weighted PCA algorithm for face recognition," in *Computational Intelligence in Robotics and Automation*, 2005, pp. 219-223.

[28] K. Pearson, "On lines and planes of closest fit to systems of points in space," Philosophical Magazine, pp. 559-572, 1901.

[29] D. Skocaj and A. Leonardis, "Weighted and robust incremental method for subspace learning," *Ninth IEEE International Conference on Computer Vision ICCV 2003,* 2003, pp. 1494-1501.

[30] A. A. Amini and M. Wainwright, "High-dimensional analysis of semi-definite relaxations for sparse principal components," *The Annals of Statistics*, pp. 2877-2921, March 2009.

[31] J. Sturm, "Using SEDUMI 1.0x, a MATLAB toolbox for optimization over symmetric cones," Optimization Methods and Software, vol. 11, pp. 625-653, 1999.

[32] K. C. Toh, M. J.T odd, and R. HTutuncu, "SDPT3- a MATLAB software package for semi-definite programming," Optimization Methods and Software, vol. 11, pp. 545-581, 1999.

Thanh D. X. Duong was born in Vietnam, in 1982. Duong received BS. Maths & Informatics (2003), MA. Maths (2007), MS. Computer Science (2008) and PhD. Applied Mathematics (2009), all with high distinction from HCM University of Science.

He is currently leader of Quantitative Strategies and Analysis at VietFund Management. Concurrently, he is affiliate leader of Alternative Research at John von Neumann Institute - Vietnam National University HCM and a certified member of Professional Risk Managers' International Association. Dr. Duong was at Japan Advanced Institute of Science and Technology doing research on statistical machine learning in 2010.

Hung V. Nguyen was born in Vietnam, in 1981. Nguyen received BS. Maths & Informatics (2003), MS. Computer Science (2007), all with high distinction from HCM University of Science. Since 2009, he has been a PhD candidate in Computer Vision of Japan Advanced Institute of Science and Technology.

He is currently a senior lecturer of Pedagogy University, Ho Chi Minh city, Vietnam. Concurrently, he is a research assistant at Japan Advanced Institute of Science and Technology.