Comput Stat (2016) 31:1403–1427
DOI 10.1007/s00180-015-0608-4

ORIGINAL PAPER

Sparse principal component analysis subject to prespecified cardinality of loadings

Kohei Adachi[1] · **Nickolay T. Trendafilov**[2]

Received: 17 February 2015 / Accepted: 6 July 2015 / Published online: 22 July 2015

Abstract Most of the existing procedures for sparse principal component analysis (PCA) use a penalty function to obtain a sparse matrix of weights by which a data matrix is post-multiplied to produce PC scores. In this paper, we propose a new sparse PCA procedure which differs from the existing ones in two ways. First, the new procedure does not sparsify the weight matrix. Instead, the so-called loadings matrix is sparsified by which the score matrix is post-multiplied to approximate the data matrix. Second, the cardinality of the loading matrix i.e., the total number of nonzero loadings, is pre-specified to be an integer without using penalty functions. The procedure is called unpenalized sparse loading PCA (USLPCA). A desirable property of USLPCA is that the indices for the percentages of explained variances can be defined in the same form as in the standard PCA. We develop an alternate least squares algorithm for USLPCA which uses the fact that the PCA loss function can be decomposed as a sum of a term irrelevant to the loadings, and another one being easily minimized under cardinality constraints. A procedure is also presented for selecting the best cardinality using information criteria. The procedures are assessed in a simulation study and illustrated with real data examples.

Keywords Penalty-free approach · Sparse loadings · Alternating least squares · Percentages of explained variances

✉ Kohei Adachi
adachi@hus.osaka-u.ac.jp

Nickolay T. Trendafilov
nickolay.trendafilov@open.ac.uk

[1] Graduate School of Human Sciences, Osaka University, 1-2 Yamadaoka, Suita, Osaka 565-0871, Japan

[2] Department of Mathematics and Statistics, Open University, Walton Hall, Milton Keynes MK7 6AA, UK

1 Introduction

For an n-observations $\times$ p-variables column-centered data matrix $\mathbf{X}$, principal component analysis (PCA) can be formulated as minimizing the least squares function

$$LS = \|\mathbf{X} - \mathbf{F}\mathbf{A}'\|^2 = \|\mathbf{X} - \mathbf{X}\mathbf{W}\mathbf{A}'\|^2 \tag{1.1}$$

over $\mathbf{W}$ ($p \times m$) and $\mathbf{A}$ ($p \times m$) (e.g., Izenman 2008; Zou et al. 2006). Here, $\| \bullet \|^2$ indicates the squared Frobenius norm and $\mathbf{F} = \mathbf{X}\mathbf{W}$ is an $n \times m$ matrix of PC scores, where $m \leq \min(n, p)$ is the number of components. That is, PCA is regarded as an approximation of $\mathbf{X}$ by a lower rank matrix $\mathbf{F}\mathbf{A}'$. It is well known that the problem is solved through the singular value decomposition (SVD) of $\mathbf{X}$ (Eckart and Young 1936; Takane 2014).

The term "loading matrix" is used by some authors for $\mathbf{A}$, and by others for $\mathbf{W}$. To avoid this confusion, we call $\mathbf{W}$ weight matrix as it weighs the variables in $\mathbf{X}$, while we call $\mathbf{A}$ loading matrix as it describes how the variables load the components in $\mathbf{F}$. The matrix $\mathbf{W}$ or $\mathbf{A}$ is interpreted to capture the relationships among variables and components. In either case, the interpreted matrix is desired to be sparse, i.e., to have a great number of zero elements, since a sparse matrix is easily interpreted by focusing only on the variables and components linked with nonzero elements. However, such sparse $\mathbf{A}$ or $\mathbf{W}$ cannot be obtained by the standard PCA. For this reason, a number of modified PCA procedures have been proposed in the last decade, which produce sparse solutions (Trendafilov 2014). Such procedures are called *sparse* PCA.

All existing sparse PCA procedures produce sparse weight matrix $\mathbf{W}$ (not $\mathbf{A}$). Also, most of them are using penalty functions that penalize$\mathbf{W}$ for having nonzero elements. Such examples are SCoTLASS (Jolliffe et al. 2003), SPCA (Zou et al. 2006), and sPCA-rSVD (Shen and Huang 2008). Further development of the penalty-based procedures has been proposed by Journée et al. (2010), d'Aspremont et al. (2008), and Witten et al. (2009). They are all formulated in a similar manner: the sum of $\lambda \times$ P($\mathbf{W}$) and the loss function as in (1.1) is minimized or the function is minimized subject to P($\mathbf{W}$) $\leq 1/\lambda$. Here, P($\mathbf{W}$) stands for a penalty function, and the weight λ for the penalty is a tuning parameter(s) specifying the relative importance of P($\mathbf{W}$). Thus, the penalty weight λ controls the number of nonzero elements, which is called *cardinality*: a larger value of λ leads to $\mathbf{W}$ of lower cardinality, i.e., being sparser. Thus, the exact cardinality of the solution is not known in advance, and it is found after the sparsifying procedure is carried out with particular λ.

In this paper, we propose a new sparse PCA procedure which differs from the existing ones in the following points:

[1] The new procedure gives sparse loading matrix $\mathbf{A}$ rather than $\mathbf{W}$.
[2] It does not use a penalty function, and the cardinality of $\mathbf{A}$ is prespecified.

By these features, we refer to our proposed procedure as unpenalized sparse loading PCA (USLPCA). Recently Van Deun et al. (2011) proposed a similar approach to sparsify $\mathbf{A}$ in simultaneous PCA of several matrices. However, in contrast to USLPCA, they use penalty functions to achieve sparse $\mathbf{A}$. Therefore, our proposed USLPCA with

both features [1] and [2] is new, to the best of our knowledge. The implications and the benefits of [1] and [2] are discussed in detail in the next three paragraphs.

Whether **W** or **A** is to be sparsified depends on "in what the interest is" when interpreting a PCA solution. If the interest is in how the original variables are presented/summarized by the components **F** = **XW**, then the weight matrix **W** should be sparsified, because it expresses how the variables are weighted to form the components. On the other hand, when the interest is in how the original variables are *explained* by the components, **A** is to be sparsified, since **A** is the coefficient matrix in the regression of **X** onto **F** as given in (1.1). Recall, that this interpretation is assumed in the software package SPSS which produces **A** by default output, not **W** (SPSS Inc. 1997). Thus, USLPCA is to be used for interpreting how variables are explained by components.

There is another important implication from [1]. It helps to avoid the difficulty of the existing procedures sparsifying **W**, which destroys the components' orthogonality, a fundamental feature of the standard PCA. That is, in most of the existing procedures, the columns of **F** = **XW** are correlated, which complicates the definition of the percentage of explained variance (PEV) indicating to what extent the variances of variables are explained by components (e.g., Zou et al. 2006; Shen and Huang 2008). On the other hand, when **W** is not sparsified, **F** can be constrained as $\mathbf{F}'\mathbf{F}$ being a diagonal matrix **D**. In USLPCA, **D** is set to the n times of the $m \times m$ identity matrix $\mathbf{I}_m$, i.e. the constraint

$$\frac{1}{n}\mathbf{F}'\mathbf{F} = \mathbf{I}_m. \tag{1.2}$$

is used. Here, it should be noticed that (1.2) and $\mathbf{F}'\mathbf{F} = \mathbf{D}$ are not essentially different, since of $\mathbf{F}\mathbf{D}^{-1/2}\mathbf{D}^{1/2}\mathbf{A}' = \mathbf{F}\mathbf{A}'$, which implies that **F** with $\mathbf{F}'\mathbf{F} = \mathbf{D}$ can be transformed to meet (1.2) without changing (1.1) and the zero elements in **A**. As shown in Sect. 4, this column-orthonormality of **F** in USLPCA allows PEV to be readily defined in the same manner as in the standard PCA. In Sect. 4, it would also be described that (1.2) equalizes the nonzero loadings in **A** to the correlation coefficients between components and the original variables when the latter are standardized. Then, obtaining a sparse **A** is very beneficial as it clearly describes the correlation structure of the dimension reduction.

Prespecifying cardinality in [2] is simply achieved by incorporating the constraint

$$\mathrm{Card}(\mathbf{A}) = c \tag{1.3}$$

in the formulation of PCA, where Card(**A**) stands for the cardinality of **A**, i.e., the number of its nonzero elements, and c is a specified integer. That is, (1.1) is minimized subject to (1.2) and (1.3) in USLPCA. Obviously, the cardinality of **A** can be predetermined to be a desired integer by substituting it into c in (1.3). Here, it should be noticed that c is also a tuning parameter to be chosen by users, as the penalty weight λ in the existing penalized approaches. However, c and λ differ in that c is an integer within a restricted range, while λ can take any positive real number. The former c is one of 1, 2, …, pm, thus we can simply obtain the solutions for all possible c to choose the solution with the best cardinality. In contrast, we cannot consider all possible values of the penalty weight λ, thus its limited values must be selected

in the penalized approaches. However, this selection is not easy, since the resulting cardinality is unknown in advance as already stated.

An example of the existing approaches for concretely illustrating their differences from USLPCA is Zou et al's (2006) SPCA, which is apparently similar to USLPCA in that the loss function is defined using (1.1). However, a linear combination of (1.1) and the penalty functions for sparsifying $\mathbf{W}$ is minimized over $\mathbf{W}$ and $\mathbf{A}$ in SPCA, where the loading matrix $\mathbf{A}$ (not $\mathbf{F}$) is constrained as $\mathbf{A}'\mathbf{A} = \mathbf{I}_m$.

Here, we must mention the differences of sparse PCA from the rotation which can be used for facilitating the interpretation of loadings in the standard PCA (Jolliffe 2002; Trendafilov and Adachi 2015). Here, the rotation refers to obtaining orthonormal $\mathbf{T}$ so that a number of the elements in $\mathbf{AT}$ are close to zero, by exploiting the property that $\mathbf{FA}'$ in (1) equals $\mathbf{FTT}'\mathbf{A}'$ and $\mathbf{FT}$ can be substituted for $\mathbf{F}$ in (1.2): $\mathbf{AT}$ can also be regarded as the loading matrix. A crucial difference of the rotation from sparse PCA is that the loadings in rotated $\mathbf{AT}$ cannot be exactly zero. Thus, users must decide, in a subjective manner, which elements can be viewed as approximately zero. Another important difference is that the rotation criteria are functions of $\mathbf{AT}$ only, and do not involve the data. This implies that it cannot be known how $\mathbf{AT}$ influences the fit to the underlying data set.

The remaining parts of the paper are organized as follows: the incorporation of the cardinality-constrained minimization with (1.3) is described in Sect. 2, which follows from the decomposition of sums-of-squares for the PCA loss function (1.1). In Sect. 3, the algorithm for USLPCA is presented. The interpretation of sparse loadings are detailed and the PEV indices are introduced in Sect. 4. A procedure for selecting the cardinality c in (1.3) is described in Sect. 5. A simulation study is reported in Sect. 6, and real data examples are considered in Sect. 7.

2 Unpenalized sparse loading PCA

In USLPCA, the PCA loss function (1.1) is minimized under the orthonormality condition (1.2) for components and the cardinality constraint (1.3) for loadings. USLPCA is thus formulated as

$$\min_{\mathbf{F},\mathbf{A}} LS(\mathbf{F},\mathbf{A}) = \|\mathbf{X} - \mathbf{FA}'\|^2 \text{ subject to Card}(\mathbf{A}) = c \text{ and } \frac{1}{n}\mathbf{F}'\mathbf{F} = \mathbf{I}_m. \quad (2.1)$$

For simplicity, we have formulated USLPCA without using $\mathbf{W}$ explicitly. By substituting $\mathbf{F} = \mathbf{XW}$, (2.1) is rewritten as $\min_{\mathbf{W},\mathbf{A}} \|\mathbf{X} - \mathbf{XWA}'\|^2$ subject to Card$(\mathbf{A}) = c$ and $n^{-1}\mathbf{W}'\mathbf{X}'\mathbf{XW} = \mathbf{I}_m$.

The key point of USLPCA is to use the fact that (1.2) leads to the decomposition of sum-of-squares for the loss function (1.1):

$$\|\mathbf{X} - \mathbf{FA}'\|^2 = \|\mathbf{X} - \mathbf{FB}'\|^2 + n\|\mathbf{B} - \mathbf{A}\|^2, \quad (2.2)$$

with $\mathbf{B}$ being the cross-product matrix of p-variables $\times$ m-components:

$$\mathbf{B} = \frac{1}{n}\mathbf{X}'\mathbf{F}. \quad (2.3)$$

The equality (2.2) is proved as follows: (1.1) is rewritten as $\|\mathbf{X}-\mathbf{FA}'\|^2 = \|\mathbf{X}-\mathbf{FB}'+\mathbf{FB}-\mathbf{FA}'\|^2$, which implies that the decomposition (2.2) holds true if $(\mathbf{X}-\mathbf{FB}')'(\mathbf{FB}'-\mathbf{FA}')$ equals the matrix of zeros $\mathbf{O}$. This is fulfilled, using (1.2) and (2.3), as

$$\begin{aligned}(\mathbf{X}-\mathbf{FB}')'(\mathbf{FB}'-\mathbf{FA}') &= \mathbf{X}'\mathbf{FB}'-\mathbf{X}'\mathbf{FA}'-\mathbf{BF}'\mathbf{FB}'+\mathbf{BF}'\mathbf{FA}' \\ &= n\mathbf{BB}'-n\mathbf{BA}'-n\mathbf{BB}'+n\mathbf{BA}'=\mathbf{O}. \end{aligned} \tag{2.4}$$

The decomposition (2.2) shows that the term $g(\mathbf{A}) = \|\mathbf{B}-\mathbf{A}\|^2$ is only relevant to $\mathbf{A}$, which implies that the minimization of (1.1) over $\mathbf{A}$ for a fixed $\mathbf{F}$ amounts to $\min_{\mathbf{A}} g(\mathbf{A})$, where $g(\mathbf{A})$ is a function in which a sparse loading matrix $\mathbf{A}$ is simply matched with the matrix $\mathbf{B}$ in (2.3). Thus, $\min_{\mathbf{A}} g(\mathbf{A})$ with the cardinality constraint (1.3) is easily attained, as shown in Sect. 3.3.

The algorithm described in the next section gives the solution for a specific value c in the cardinality constraint (1.3). Thus, it remains to choose the best c value. For this task, we adopt the following procedure:

$$\text{Choose the value } c \text{ with the best } I(c) \text{ among } c = c_{\min}, \ldots, c_{\max}, \tag{2.5}$$

where $I(c)$ is an index of the goodness of the USLPCA solution obtained for a specific c, and $c_{\min}/c_{\max}$ expresses the reasonable minimum/maximum of c. The choice of the index $I(c)$ is described in Sect. 5.

As discussed in Sect. 1, the largest interval of $[c_{\min}, c_{\max}]$ is obviously $[1, pm]$. It can be reasonably reduced as

$$c_{\min} = p, \tag{2.6}$$

$$c_{\max} = pm - \frac{m(m-1)}{2}. \tag{2.7}$$

Here, (2.6) prevents $\mathbf{A}$ from having an empty row if c goes below the limit p, i.e., the number of variables. On the other hand, it is considered in (2.7) that $M = m(m-1)/2$ elements in $\mathbf{A}$ can be set to zeros without the change in the values of loss function (1.1): they are equivalent among $c = pm - M, \ldots, pm$.

3 Algorithm

The USLPCA algorithm is outlined in Sect. 3.1. It comprises of two steps which are alternately iterated until convergence. The two steps are described in details respectively in Sects. 3.2 and 3.3, while the algorithm as a whole is considered/recounted in Sect. 3.4.

3.1 Outline

The solution for USLPCA (2.1) can be obtained by alternately iterating the update of $\mathbf{F}$ and $\mathbf{A}$. Indeed, making use of (1.2) and (2.3), we can expand the loss function (1.1)

in USLPCA as

$$LS(\mathbf{F}, \mathbf{A}) = \mathrm{tr}\mathbf{X}'\mathbf{X} + \mathrm{tr}\mathbf{A}\mathbf{F}'\mathbf{F}\mathbf{A}' - 2\mathrm{tr}\mathbf{X}'\mathbf{F}\mathbf{A} = n\mathrm{tr}\mathbf{S} + n\mathrm{tr}\mathbf{A}\mathbf{A}' - 2n\mathrm{tr}\mathbf{B}\mathbf{A}', \quad (3.1)$$

with

$$\mathbf{S} = \frac{1}{n}\mathbf{X}'\mathbf{X} \quad (3.2)$$

being the sample covariance matrix. Here, the loss function is found to be a function of $\mathbf{B}$ (and $\mathbf{A}$), though it must be kept in mind that $\mathbf{B}$ is a function of the PC score matrix $\mathbf{F}$ satisfying (1.2) as seen in (2.3). The problem (2.1) can thus be attained by alternately iterating

[B-step] minimizing (1.1) or (3.1) over $\mathbf{B}$ subject to (1.2) and (2.3) with $\mathbf{A}$ kept fixed,
[A-step] minimizing (1.1) over $\mathbf{A}$ subject to (1.3) with $\mathbf{B}$ being kept fixed,

until convergence is reached.

3.2 B-step

As $\mathbf{B}$ is a function of $\mathbf{F}$, we first consider minimizing (1.1) over $\mathbf{F}$ subject to (1.2) for a given $\mathbf{A}$. Since (1.1) is expanded as (3.1), the minimization is equivalent to maximizing $n\mathrm{tr}\mathbf{B}\mathbf{A}' = \mathrm{tr}\mathbf{X}'\mathbf{F}\mathbf{A}' = \mathrm{tr}(\mathbf{X}\mathbf{A})'\mathbf{F}'$ subject to (1.2). It is attained for

$$\mathbf{F} = \sqrt{n}\mathbf{K}\mathbf{L}' = \mathbf{X}\mathbf{A}\mathbf{L}\boldsymbol{\Lambda}^{-1}\mathbf{L}', \quad (3.3)$$

where $\mathbf{K}$ and $\mathbf{L}$ are given by the SVD of $n^{-1/2}\mathbf{X}\mathbf{A}$ defined as

$$\frac{1}{\sqrt{n}}\mathbf{X}\mathbf{A} = \mathbf{K}\boldsymbol{\Lambda}\mathbf{L}' \quad (3.4)$$

with $\mathbf{K}'\mathbf{K} = \mathbf{L}'\mathbf{L} = \mathbf{I}_p$ and $\boldsymbol{\Lambda}$ a diagonal matrix (e.g., Seber 2008). This SVD can be rewritten as $\mathbf{K} = n^{-1/2}\mathbf{X}\mathbf{A}\mathbf{L}\boldsymbol{\Lambda}^{-1}$ which leads to the last identity in (3.3). It implies that $\mathbf{F}$ is column-centered:

$$\mathbf{1}_n'\mathbf{F} = \mathbf{0}_m', \quad (3.5)$$

as $\mathbf{X}$ is, with $\mathbf{1}_n$ the $n \times 1$ vector of ones and $\mathbf{0}_m$ the $m \times 1$ zero vector.

Using (3.3) in the definition (2.3) for $\mathbf{B}$, we find that

$$\mathbf{B} = \frac{1}{n}\mathbf{X}'\mathbf{X}\mathbf{A}\mathbf{L}\boldsymbol{\Lambda}^{-1}\mathbf{L}' = \mathbf{S}\mathbf{A}\mathbf{L}\boldsymbol{\Lambda}^{-1}\mathbf{L}' \quad (3.6)$$

is to be obtained in this step. Here, it should be noted in (3.6) that $\mathbf{K}$ is not included and the matrix product $\mathbf{L}\boldsymbol{\Lambda}^{-1}\mathbf{L}'$ can be obtained through the eigenvalue decomposition (EVD)

$$\mathbf{A}'\mathbf{S}\mathbf{A} = \mathbf{L}\boldsymbol{\Lambda}^2\mathbf{L}' \quad (3.7)$$

following from (3.2) and (3.4). This EVD and (3.6) show that, in the B-step, [1] **F** may not be obtained and [2] the original data matrix **X** may not be available only if the covariance matrix **S** in (3.7) is given.

3.3 A-step

For fixed **B**, the minimization of (1.1) over constrained **A** is equivalent to the minimization of $g(\mathbf{A}) = \|\mathbf{B}-\mathbf{A}\|^2$, since of the decomposition (2.2). Using $\mathbf{A} = (a_{ij})$ and $\mathbf{B} = (b_{ij})$, we can rewrite $g(\mathbf{A})$ as

$$g(\mathbf{A}) = \|\mathbf{B}-\mathbf{A}\|^2 = \sum_{(i,j)\in Z} b_{ij}^2 + \sum_{(i,j)\in Z^{\perp}} (b_{ij}-a_{ij})^2 \geq \sum_{(i,j)\in Z} b_{ij}^2. \tag{3.8}$$

Here, Z denotes the set of the $q = pm - c$ indexes $(i,\ j)$'s indicating the locations of the loadings a_{ij} to be zero. The complement set $Z^{\perp}$ is the set containing the c indexes $(i,\ j)$'s of the nonzero a_{ij}. The inequality in (3.8) shows that $g(\mathbf{A})$ attains its lower limit $\Sigma_{(i,j)\in Z}\, b_{ij}^2$ when the non-zero loadings a_{ij} with $(i,\ j)\in Z^{\perp}$ are taken equal to the corresponding b_{ij}. Moreover, the limit $\Sigma_{(i,j)\in \mathbf{Z}}\, b_{ij}^2$ is minimal, when Z contains the indexes for the q smallest b_{ij}^2 among all squared elements of **B**. Thus, $g(\mathbf{A})$ is minimized for $\mathbf{A} = (a_{ij})$ being

$$a_{ij} = \begin{cases} 0 & \textit{iff } b_{ij}^2 \leq b_{[q]}^2 \\ b_{ij} & \text{otherwise} \end{cases} \tag{3.9}$$

with $b_{[q]}^2$ the qth smallest value among all b_{ij}^2.

The loading matrix **A** updated by (3.9) satisfies

$$a_{ij}^2 = a_{ij}b_{ij}, \quad \text{or equivalently,} \quad \mathbf{A}\bullet\mathbf{A} = \mathbf{A}\bullet\mathbf{B}, \tag{3.10}$$

where $\bullet$ denotes the Hadamard element-wise matrix product. This property is important to prove the expression of the explained variance discussed later.

3.4 Whole steps

The value of loss function (1.1) or (3.1) attained after the A-step is given by

$$LS(\mathbf{A}) = n\mathrm{tr}\mathbf{S} + n\mathrm{tr}\mathbf{A}\mathbf{A}' - 2n\mathrm{tr}\mathbf{B}\mathbf{A}' = n\mathrm{tr}\mathbf{S} - n\mathrm{tr}\mathbf{A}\mathbf{A}', \tag{3.11}$$

which is derived using (3.10) in (3.1). Further, the loss function value (3.11) is divided by $n\mathrm{tr}\mathbf{S}$ to yield

$$LS_{\mathrm{N}}(\mathbf{A}) = 1 - \frac{\mathrm{tr}\mathbf{A}\mathbf{A}'}{\mathrm{tr}\,\mathbf{S}}, \tag{3.12}$$

which is convenient for checking convergence, as it is normalized so as to take values within the range [0, 1].

We should note that (3.12) does not include the original data matrix $\mathbf{X}$, which is also unnecessary in the B- and the A-steps. In other words, USLPCA can be used if the covariance matrix $\mathbf{S}$ is available only. This is convenient when $n > p$, as the $p \times p$ matrix $\mathbf{S}$ is smaller than $\mathbf{X}$ ($n \times p$). Further, the updating of $\mathbf{F}$ described in Sect. 3.2 can be avoided. Thus, the USLPCA algorithm for obtaining the optimal $\mathbf{A}$ can be shorten as follows:

[1] Initialize $\mathbf{A}$
[2] Perform EVD (3.7) to obtain $\mathbf{B}$ with (3.6)
[3] Obtain $\mathbf{A}$ with (3.9)
[4] Finish if $\Delta LS_{\mathrm{N}}(\mathbf{A}) \leq \varepsilon$; otherwise go back to [2]

Here, $\Delta LS_{\mathrm{N}}(\mathbf{A})$ denotes the change in (3.12) from the previous round, and ε is set to 0.1^7 in this paper. This algorithm is run multiple times by starting with different initial $\mathbf{A}$ in the procedure described in Appendix "Multiple-runs procedure". Among the resulting multiple solutions, we select the $\mathbf{A}$ with the lowest $LS_{\mathrm{N}}(\mathbf{A})$ value as the optimal one, in order to avoid local minimizers. After those processes, the PC score matrix $\mathbf{F}$ can be obtained using the optimal $\mathbf{A}$, $\mathbf{X}$, and (3.7) in (3.3), if $\mathbf{F}$ is of interest.

4 Interpreting solutions

As described in Sect. 1, the weight matrix $\mathbf{W}$ is sparsified in the existing sparse PCA, which implies that $\mathbf{W}$ is supposed to be interpreted. On the other hand, the loading matrix $\mathbf{A}$ is to be interpreted in USLPCA with $\mathbf{A}$ sparsified. In Sect. 4.1, we start with contrasting the interpretations of $\mathbf{W}$ and $\mathbf{A}$. In Sect. 4.2, we show that the nonzero loadings in USLPCA are also the covariances of components to variables and further equal the correlation coefficients when $\mathbf{X}$ is standardized. Finally, the indices of the percentage of explained variances are introduced in Sect. 4.3.

4.1 Interpreting weights versus interpreting loadings

In this section, we compare the proposed USLPCA with other approaches in which the weights in $\mathbf{W}$ are sparsified.

The role of $\mathbf{W}$ is to weight the original variables to form $\mathbf{F} = \mathbf{XW}$, which is rewritten in the vector form as

$$\mathbf{f}_j = \mathbf{X}\mathbf{w}_j = w_{1j}\mathbf{x}_1 + \cdots + w_{pj}\mathbf{x}_p = \sum_{i \in M_j} w_{ij}\mathbf{x}_i \quad (j = 1, \ldots, m)\,. \tag{4.1}$$

Here, $\mathbf{f}_j$ and $\mathbf{w}_j$ are the jth columns of $\mathbf{F}$ and $\mathbf{W} = (w_{ij})$, respectively, $\mathbf{x}_i$ denotes the ith column (variable) of $\mathbf{X}$, and M_j denotes the set of indexes $\{i\}$ corresponding to the nonzero weights in $\mathbf{w}_j$. That is, component j is interpreted as summarizing the variables $\mathbf{x}_j$ weighted by nonzero w_{ij}. This explains why the existing sparse PCA procedures produce sparse $\mathbf{W}$.

On the other hand, USLPCA solutions can be interpreted with the equation

$$\mathbf{X} = \mathbf{F}\mathbf{A}' + \mathbf{E}, \tag{4.2}$$

which may be called a PCA model, since the squared Frobenius norm of the error matrix $\mathbf{E}$ in (4.2) leads to the PCA loss function (1.1). Using $\tilde{\mathbf{a}}_i'(1 \times m)$ for the ith row of $\mathbf{A} = (a_{ij})$ and $\mathbf{e}_i$ for the ith column of $\mathbf{E}$, (4.2) is rewritten in the vector from

$$\mathbf{x}_i = \mathbf{F}\tilde{\mathbf{a}}_i = a_{i1}\mathbf{f}_1 + \cdots + a_{im}\mathbf{f}_m + \mathbf{e}_i = \sum_{j \in N_i} a_{ij}\mathbf{f}_j + \mathbf{e}_i \tag{4.3}$$

with N_i is the set of index $\{j\}$ corresponding to nonzero loadings in $\tilde{\mathbf{a}}_i$. They can be regarded as the coefficients in the multiple regression of $\mathbf{x}_i$ onto $\mathbf{f}_j$'s. That is, variable i is interpreted as a dependent variable explained by the components with nonzero a_{ij}. To what extent the variable are explained by the components, or equivalently, the smallness of the sizes of errors in $\mathbf{e}_i$, are indicated by the percentage of explained variances introduced in Sect. 4.3.

Beside the above interpretation for each variable, USLPCA solutions can also be interpreted component-wise. For describing it, we use $\mathbf{a}_j$ for the jth column of $\mathbf{A}$ to rewrite (4.2) as $\mathbf{X} = \mathbf{f}_1\mathbf{a}_1' + \cdots + \mathbf{f}_m\mathbf{a}_m' + \mathbf{E} = \mathbf{f}_j\mathbf{a}_j' + (\Sigma_{k \neq j}\mathbf{f}_k\mathbf{a}_k' + \mathbf{E})$, i.e.,

$$\left[\mathbf{x}_1, \ldots, \mathbf{x}_p\right] = \mathbf{f}_j\left[a_{1j}, \ldots, a_{pj}\right] + \mathbf{H}_{[j]}. \tag{4.4}$$

Here, $\mathbf{a}_j' = [a_{1j}, \ldots, a_{pj}]$, and $\mathbf{H}_{[j]} = \Sigma_{k \neq j}\mathbf{f}_k\mathbf{a}_k' + \mathbf{E}$ whose columns are uncorrelated with those of $\mathbf{f}_j\mathbf{a}_j' = \mathbf{f}_j[a_{1j}, \ldots, a_{pj}]$ as proved in Appendix "No correlation for components and errors". Equation (4.4) thus allows component j to be interpreted as a common factor exclusively explaining the variables associated with nonzero a_{ij}. In Sect. 4.3, the percentage is introduced that indicates how well the component explains the variables.

4.2 Nonzero loadings as covariances

Since $\mathbf{F}$ is column-centered with (3.5) as $\mathbf{X}$ is, the cross-product matrix $\mathbf{B} = n^{-1}\mathbf{X}'\mathbf{F}$ in (2.3) contains the covariances between p variables and m components. By taking this into account in (3.9), we can find that the nonzero loadings in the resulting $\mathbf{A}$ equal the corresponding covariances in $\mathbf{B}$: nonzero a_{ij} equals the covariance between the ith original variable and the jth component. Further, this implies that the nonzero loadings equal the correlation coefficients of variables to components, when a data set to be analyzed is a standardized data matrix $\mathbf{X}$ with unit variances or a correlation matrix $\mathbf{S}$, since the PC scores, constrained as (1.2), have unit variances. It allows us to easily capture the magnitudes of loadings, as their ranges are restricted within $[-1, 1]$.

The above equivalence of nonzero loadings to covariances motivates us to relate the optimization in USLPCA to covariances as follows: using (1.2) and (2.3), the decomposed loss function (2.2) can be rewritten as $n\mathrm{tr}\mathbf{S} - n\|\mathbf{B}\|^2 + n\|\mathbf{B} - \mathbf{A}\|^2$, whose min-

imization amounts to maximizing $\|\mathbf{B}\|^2 - \|\mathbf{B} - \mathbf{A}\|^2$. USLPCA can thus be viewed as maximizing the sum of squared covariances $\|\mathbf{B}\|^2$ subject to $\mathbf{B}$ approximating sparse $\mathbf{A}$.

4.3 Percentages of explained variances

From the standardized loss function value (3.12), we can derive a goodness-of-fit index

$$PEV = 100 \times \frac{\text{tr}\,\mathbf{A}\mathbf{A}'}{\text{tr}\,\mathbf{S}} = 100 \times \frac{\left\|\mathbf{F}\mathbf{A}'\right\|^2}{\|\mathbf{X}\|^2} = 100 \times \left(1 - \frac{\left\|\mathbf{X} - \mathbf{F}\mathbf{A}'\right\|^2}{\|\mathbf{X}\|^2}\right). \quad (4.5)$$

Here, the last identity is added to stress that this index directly follows from the loss function (1.1). Note, that it attains the value (3.11) divided by $n\text{tr}\mathbf{S} = \|\mathbf{X}\|^2$ to give (3.12), one minus which leads to (4.5). This index can be called *total percentage of explained variance* (PEV), as the denominator tr$\mathbf{S}$ in (4.5) is the total variance of variables, while the numerator tr$\mathbf{A}\mathbf{A}'$, which is found to equal $n^{-1}\|\mathbf{F}\mathbf{A}'\|^2$ using (1.2), is the total variance for $\mathbf{F}\mathbf{A}'$, since (3.5) implies $\mathbf{F}\mathbf{A}'$ being column-centered.

The total PEV (4.5) can be decomposed as a sum of

$$PEV\,(j) = 100 \times \frac{\left\|\mathbf{a}_j\right\|^2}{\text{tr}\,\mathbf{S}} \quad (4.6)$$

over $j = 1, \ldots, m$. Since (1.2) leads to $\|\mathbf{a}_j\|^2 = n^{-1}\|\mathbf{f}_j\mathbf{a}_j'\|^2$, the percentage (4.6) is regarded as the amount of total variance of variables explained by component j and used for the interpretation of the component with (4.4).

The PEV for each variable is derived from (4.5), which can be rewritten as $n\sum_{i=1}^{p}(s_{ii} - \|\tilde{\mathbf{a}}\|^2) = n\sum_{i=1}^{p} s_{ii}(1 - \|\tilde{\mathbf{a}}_i\|^2/s_{ii}) \geq 0$ with s_{ii} the ith diagonal element of $\mathbf{S}$, i.e., the variance of variable i. This gives the statistic

$$PEV[i] = 100 \times \frac{\|\tilde{\mathbf{a}}_i\|^2}{s_{ii}}. \quad (4.7)$$

Since (1.2) implies $\|\tilde{\mathbf{a}}_i\|^2 = n^{-1}\|\mathbf{F}\tilde{\mathbf{a}}_i\|^2$, the percentage (4.7) expresses the amount of the variance of variable i explained by the components associated with the nonzero loadings in $\tilde{\mathbf{a}}_i$. Therefore, (4.7) is used for the interpretation with the model (4.3) which expresses the regression of a variable onto components.

In the same forms as (4.5), (4.6), and (4.7), the PEV indices are defined for the standard PCA, which can be formulated as minimizing (1.1) subject to (1.2) and $\mathbf{A}'\mathbf{A}$ being a diagonal matrix. The solution of $\mathbf{A}$ is expressed as $\mathbf{A} = \mathbf{R}\boldsymbol{\Delta}$, with $\boldsymbol{\Delta}$ is the $m \times m$ diagonal matrix including the m largest singular values of $n^{-1/2}\mathbf{X}$ and the corresponding right singular vectors being the columns of $\mathbf{R}$. The substitution of $\mathbf{A} = \mathbf{R}\boldsymbol{\Delta}$ into (4.5), (4.6), and (4.7) gives the PEV indices for PCA, with its total PEV expressed as $PEV_{\text{PCA}} = \text{tr}\boldsymbol{\Delta}^2/\text{tr}\mathbf{S}$. The same form of the PEV definition between USLPCA and PCA implies that the increase in the cardinality value c in (1.3) allows the total PEV of USLPCA (4.5) to approach and finally equal PEV_{PCA}. It suggests

that we can find the acceptable cardinality for USLPCA with the corresponding value of (4.5) being not substantially less than PEV_{PCA}. However, whether the difference in PEV values is substantial must be decided subjectively. A procedure without such a decision is discussed in the next section.

5 Cardinality selection by information criteria

The cardinality selection problem (2.5) can be viewed as a model selection problem for the optimal combination of the goodness-of-fit for a data set, and the number of parameters to be estimated. Here, the latter is directly related to the cardinality which is the number of the loadings whose values are to be estimated. For such a model selection, indices usually called information criteria can be used, which include AIC and BIC as popular ones (Akaike 1974; Schwarz 1978). Their useful feature is that a model with the least index value is selected as the optimal one: model selection can be attained numerically without subjective decision.

The information criteria are based on the maximum likelihood (ML) method, while USLPCA is formulated as a least squares (LS) method. However, USLPCA can be reformulated as an ML procedure, by assuming that $\mathbf{X}$ is generated with the PCA model (4.2) with each error in $\mathbf{E} = (e_{ti})$ distributed independently and identically according to the normal distribution with its mean 0 and variance σ^2:

$$e_{ti} \sim N(0, \sigma^2). \tag{5.1}$$

The model (4.2) with the normality assumption (5.1) leads to the log likelihood whose part relevant to $\mathbf{F}$ and $\mathbf{A}$ is expressed as

$$l(\mathbf{F}, \mathbf{A}) = -\frac{np}{2} \log \|\mathbf{X} - \mathbf{FA}\|^2. \tag{5.2}$$

This is derived using the fact that the ML estimate of the variance must satisfy $\sigma^2 = (np)^{-1}\|\mathbf{X} - \mathbf{FA}\|^2$, as described in Appendix "Likelihood for PCA model". The ML version of USLPCA is formulated as maximizing (5.2) subject to (1.2) and (1.3), which is equivalent to the LS-based one in Sects. 2 and 3.

By substituting the USLPCA solution into (5.2), it can be rewritten as

$$l(\mathbf{A}) = -\frac{np}{2} \log\{LS_{\mathrm{N}}(\mathbf{A}) \times n\mathrm{tr}\mathbf{S}\} = -\frac{np}{2} \log\left(1 - \frac{PEV}{100}\right) - Const \tag{5.3}$$

with $Const = (np/2)\log(n\mathrm{tr}\mathbf{S})$ irrelevant to $\mathbf{A}$. Here, we have used that $\|\mathbf{X} - \mathbf{FA}\|^2$ attains the value expressed as (3.11), which is the product of $n\mathrm{tr}\mathbf{S}$ and (3.12) with the latter leading to the *PEV* in (4.5). Using (5.3), AIC and BIC are given by

$$\mathrm{AIC}(c) = -2l(\mathbf{A}) + 2 \times \eta(c) \tag{5.4}$$

$$\mathrm{BIC}(c) = -2l(\mathbf{A}) + \log(np) \times \eta(c) \tag{5.5}$$

as functions of cardinality c, where the number of parameters $\eta(c)$ is just the cardinality of the loading matrix $\mathbf{A}$ with $\eta(c) = c$, since (5.3) is a function of only the loading matrix $\mathbf{A}$ with the values of its c elements to be estimated.

We use either (5.4) or (5.5) as the index $I(c)$ in the cardinality selection problem (2.5). Thus, it is rewritten as

$$\text{Choose } \hat{c} = \underset{c_{\min} \le c \le c_{\max}}{\arg\min} \; BIC\,(c) \text{ as the optimal cardinality} \tag{5.6}$$

if (5.5) is used; otherwise $BIC(c)$ is replaced by AIC(c). Here, $c_{\min}$ and $c_{\max}$ are given by (2.6) and (2.7). That is, the best cardinality can be selected through the runs of the USLPCA algorithm with c set to $c_{\min}, \ldots, c_{\max}$. As a result, we have Δ_c solutions with

$$\Delta_c = c_{\max} - c_{\min} + 1. \tag{5.7}$$

Those solutions give the Δ_c BIC/AIC values, among which the least one gives the best cardinality with (5.6) or its AIC version.

6 Simulation study

We performed a simulation study to assess [1] how often local minima arise and to what degree local minimizers differ from the optimal solution in USLPCA, [2] how well the true loadings are recovered by USLPCA when the cardinality c in (1.3) is set to the true value, and [3] how exactly the true cardinality is identified by AIC and BIC. The procedure for synthesizing data is described in Sect. 6.1, which is followed by the assessment of [1] in Sect. 6.2, that of [2] in 6.3, and the results for [3] in 6.4.

6.1 Data synthesis procedure

We generate data matrices $\mathbf{X}$ having more observations than variables with $n = 150 > p = 15$ and horizontal ones having more variables with $n = 15 < p = 150$, on the basis of the PCA model (4.2) subject to (1.2), (1.3), and (5.1). Here, the approximate PEV for $\mathbf{X} = \mathbf{F}\mathbf{A}' + \mathbf{E}$,

$$APEV = 100 \times \frac{\left\|\mathbf{F}\mathbf{A}'\right\|^2}{\|\mathbf{F}\mathbf{A}'\|^2 + \|\mathbf{E}\|^2} \cong 100 \times \frac{\left\|\mathbf{F}\mathbf{A}'\right\|^2}{\left\|\mathbf{F}\mathbf{A}' + \mathbf{E}\right\|^2} = 100 \times \frac{\left\|\mathbf{F}\mathbf{A}'\right\|^2}{\|\mathbf{X}\|^2}, \tag{6.1}$$

is controlled to be 80, 60, or 40. As described below, $\mathbf{F}$ and $\mathbf{E}$ are mutually independently generated, so that $\mathbf{F}'\mathbf{E}$ is close to the zero matrix. Thus, $\|\mathbf{F}\mathbf{A}' + \mathbf{E}\|^2 \cong \|\mathbf{F}\mathbf{A}'\|^2 + \|\mathbf{E}\|^2$, and (4.5) is approximated by (6.1). Its values 80, 60, and 40 correspond to the error sizes being small, medium, and large, respectively. A set of $\mathbf{F}$, $\mathbf{A}$, and $\mathbf{E}$ is synthesized by setting $m = 3$ with the following steps.

[1] An integer within the interval $[p, pm/2]$ is randomly chosen as the cardinality of the true $\mathbf{A}$.

[2] The locations of the nonzero elements in $\mathbf{A}$ are randomly chosen subject to that each row of $\mathbf{A}$ includes at least one nonzero loading and each column includes at least two nonzero ones.
[3] Each of the nonzero elements in $\mathbf{A}$ is drawn randomly from $U(0.5, 1)$ or $U(-1, -0.5)$, with $U(\alpha, \beta)$ the uniform distribution over the range $[\alpha, \beta]$.
[4] $\mathbf{F}$ is filled with the standard normal variables and then orthonormalized so as to satisfy (1.2).
[5] Each element of $\mathbf{E}$ is drawn with (5.1) independently of $\mathbf{F}$, where σ^2 is set so that *APEV* is 0.8, 0.6, or 0.4.

For each of the 2 ($n > p$ and $n < p$) $\times$ 3 (*APEV* values) combinations, we generated 100 data matrices. For the resulting data sets, we set m at 3 to carry out USLPCA with the cardinality selection procedure from Sect. 5.

6.2 Local minima

For Card($\mathbf{A}$) set to a specific c, the USLPCA algorithm is run multiple (K_c) times in the procedure described in Appendix "Multiple-runs procedure". As defined there, the solution $\mathbf{A}$ resulting from the kth run ($k = 1, \ldots, K_c$) with this c is denoted by $\mathbf{A}_{ck}$, and $\mathbf{A}_c = \text{argmin}_{1 \le k \le K_c} LS_{\text{N}}(\mathbf{A}_{ck})$. Let $\mathbf{A}_c$ be the optimal solution. We define $\mathbf{A}_{ck}$ as a local minimizer (LM), if the *averaged absolute difference* of $\mathbf{A}_{ck} = (a_{ij}^{(ck)})$ to $\mathbf{A}_c = (a_{ij}^{(c)})$, which is defined as $\text{AAD}(\mathbf{A}_{ck}, \mathbf{A}_c) = (pm)^{-1} \Sigma_i \Sigma_j |a_{ij}^{(ck)} - a_{ij}^{(c)}|$, is greater than 0.001. We count the frequency L_c with which LMs are observed during the K_c runs of the algorithm, and obtain the average of the LMs' proportion of L_c/K_c over c, i.e., $P_{\text{LM}} = \Delta_c^{-1} \sum_{c=c_{\min}}^{c_{\max}} L_c/K_c$, where Δ_c is given in (5.7). Further, we define the difference of LMs to the optimal solution as $D_{\text{LM}}(c) = L_c^{-1} \Sigma_\Gamma \text{AAD}(\mathbf{A}_{ck}, \mathbf{A}_c)$ and obtained its average over c, i.e., $D_{LM} = \Delta_c^{-1} \sum_{c=c_{\min}}^{c_{\max}} D_{\text{LM}}(c)$, where Γ denotes the set of $\mathbf{A}_{ck}$ being LMs.

Table 1 shows the percentiles and averages of P_{LM} and D_{LM} over 100 data sets. There, P_{LM} are found to be very high, in particular, when *APEV* is lower, i.e., errors are larger. In particular, the 10 percentile of P_{LM} in the bottom row is 0.94, which shows

Table 1 Statistics of the proportions of local minimizers (LMs) and their differences from the optimal solution

Data form	*APEV*	Proportion (P_{LM})				Difference (D_{LM})			
		10 %	50 %	90 %	Ave.	10 %	50 %	90 %	Ave.
$n > p$	80	0.793	0.837	0.881	0.837	0.106	0.133	0.179	0.138
	60	0.827	0.866	0.918	0.867	0.109	0.141	0.194	0.146
	40	0.851	0.904	0.951	0.902	0.120	0.158	0.217	0.166
$p > n$	80	0.743	0.784	0.829	0.784	0.084	0.100	0.123	0.102
	60	0.835	0.871	0.919	0.873	0.082	0.102	0.131	0.105
	40	0.940	0.965	0.980	0.962	0.115	0.151	0.204	0.155

that the 94 percents of the solutions were LMs for the 90 % (=100 − 10 percentile) of data sets of $n < p$ with $APEV = 40\,\%$. The D_{LM} values are also found to be large: those averages exceed 0.1 for all conditions, which implies that an LM corresponding to the optimal loading 0.5 is less than 0.4 or larger than 0.6. We can thus conclude that USLPCA is sensitive to local minima and LMs are not similar to the optimal solutions. Despite these rather disappointing results, the optimal USLPCA solutions are close to the true values as shown next.

6.3 Recovery of true loadings

Let us use $\hat{\mathbf{A}}$ for the solution $\mathbf{A}_c$ obtained for c being the true cardinality. We define two recovery indices for each pair of the true $\mathbf{A} = (a_{ij})$ and its estimate $\hat{\mathbf{A}} = (\hat{a}_{ij})$. One is the *misidentification rate of zero loadings* $\mathrm{MR}_0 = 1 - N_{00}/N_0$, where N_0 is the number of the true zero loadings and N_{00} is the number of $(i,\ j)$'s with $a_{ij} = \hat{a}_{ij} = 0$. The other index is the averaged absolute differences of $\hat{\mathbf{A}}$ to its true counterpart, i.e., $\mathrm{AAD}(\hat{\mathbf{A}}, \mathbf{A}) = (pm)^{-1} \Sigma_i \Sigma_j |\hat{a}_{ij} - a_{ij}|$. Table 2 presents the percentiles and averages of the indices over 100 solutions.

First, let us note the results for the data sets of $n > p$ in Table 2. The Panel for MR_0 in Table 2 shows that no misidentification occurred for any data set of $n > p$, except the rare cases in the condition of $APEV = 40\,\%$ with the average 0.005. The statistics of $\mathrm{AAD}(\hat{\mathbf{A}}, \mathbf{A})$ are also found to be satisfactorily small. Indeed, even the worst value 0.069 (the 90 percentile for $APEV = 40\,\%$) is not considered to indicate large differences between $\mathbf{A}$ and $\hat{\mathbf{A}}$ in that the examples of $[a_{ij},\ \hat{a}_{ij}]$ giving the value 0.069 such as [0.500, 0.569] and [−0.800, −0.869] never impress us as showing bad recovery. We can thus conclude that sparse loadings were recovered fairly well for $n > p$, when the cardinality is set to the true value.

Next, we consider the case $p > n$. Compared with $n > p$, the recovery for $p > n$ is worse and clearly depends on $APEV$. When it is 80 %, the statistics of MR_0 and $\mathrm{AAD}(\hat{\mathbf{A}}, \mathbf{A})$ are small enough and show fairly good recovery, but they are not small for $APEV = 40\,\%$. Also for the 10 % of the data sets for $APEV = 60\,\%$, the recovery is found to be unsatisfactory, as the 90 percentile of $\mathrm{AAD}(\hat{\mathbf{A}}, \mathbf{A})$ exceeds 0.1. The results

Table 2 Statistics of the indices for the recovery of sparse loadings

Data form	*APEV*	Misidentification rate (MR_0)				Difference ($\mathrm{AAD}(\hat{\mathbf{A}}, \mathbf{A})$)			
		10 %	50 %	90 %	Ave.	10 %	50 %	90 %	Ave.
$n > p$	80	0.000	0.000	0.000	0.000	0.009	0.013	0.019	0.014
	60	0.000	0.000	0.000	0.000	0.020	0.027	0.035	0.027
	40	0.000	0.000	0.000	0.005	0.039	0.052	0.069	0.055
$p > n$	80	0.000	0.004	0.013	0.006	0.031	0.042	0.054	0.042
	60	0.025	0.047	0.069	0.047	0.062	0.083	0.103	0.082
	40	0.086	0.124	0.185	0.131	0.123	0.154	0.197	0.157

Table 3 Statistics of the relative biases of the cardinality selected by AIC and BIC

Data form	*APEV*	AIC				BIC			
		10 %	50 %	90 %	Ave.	10 %	50 %	90 %	Ave.
$n > p$	80	0.107	0.214	0.286	0.201	0.000	0.000	0.036	0.018
	60	0.111	0.214	0.321	0.220	0.000	0.036	0.071	0.025
	40	0.146	0.250	0.357	0.246	0.000	0.036	0.071	0.036
$p > n$	80	0.152	0.195	0.251	0.203	0.000	0.007	0.020	0.009
	60	0.145	0.186	0.241	0.188	−0.144	−0.081	−0.030	−0.081
	40	0.057	0.121	0.185	0.125	−0.224	−0.122	−0.030	−0.127

imply that good recovery of the loadings is guaranteed for data sets with *PEV* of 80 %, but is not true for those with *PEV* less than 60 %.

6.4 Identification of true cardinality

Let us use c for the true cardinality and $\hat{c}$ for the cardinality selected by AIC or BIC. For each data set, we obtain the relative bias $(\hat{c} - c)/\Delta_c$, and its percentiles over the 100 data sets are shown in Table 3. For the data sets of $n > p$, we find that AIC overestimates the cardinality, but it is fairly well identified by BIC, which shows that BIC should be used for the cases of $n > p$. On the other hand, the behavior of AIC and BIC depend on *APEV* for the data sets with $p > n$. That is, although BIC identified the cardinality well for the cases with *APEV* = 80 %, it is found that the overestimation of the cardinality by AIC is reduced and the underestimation by BIC is reinforced with the decrease in *APEV*. Eventually, AIC works better for data sets with $p > n$ and less *APEV*, so its performance is almost equivalent to that of BIC in the absolute values of averages for *APEV* = 40 %. As BIC is still superior for *APEV* = 60 %, the results suggest that BIC should be used for $APEV \geq 60\,\%$, but it is inconclusive whether AIC or BIC is to be used for $APEV \leq 40\,\%$.

7 Examples

In this section, we illustrate USLPCA with two data matrices of $n > p$ and $n < p$, respectively. Each data set is standardized or given as a correlation matrix, so that the nonzero loadings to be obtained stand for the correlations of variables to components. The zero loadings are left blank in the following tables.

7.1 Pitprop data

The first example is Jeffers's (1967) Pitprop data matrix of $n = 180$ by $p = 13$, which has been used as a benchmark for testing sparse PCA procedures. We carried out USLPCA with $m = 6$ following the previous studies. Then, local minimizers were often obtained with the average proportion $P_{\mathrm{LM}} = 0.852$ and they are not similar to the

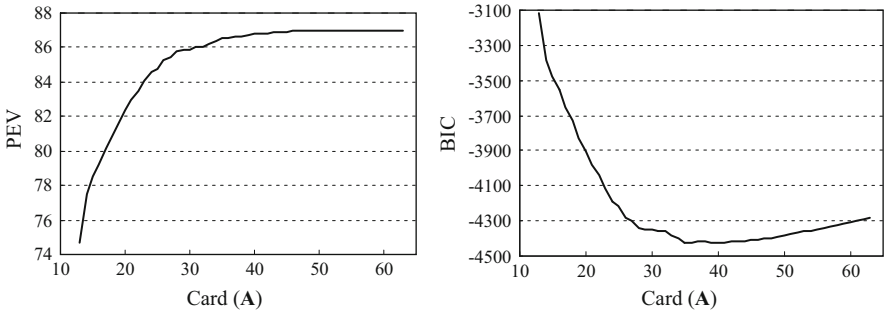

Fig. 1 Total PEV and BIC against the cardinality of **A** for Pitprop data

Table 4 USLPCA loadings with the least BIC for Pitprop data together with PCA's PEV in the final row and column

Variable	USLPCA: Card(**A**) = 37							PCA
	C1	C2	C3	C4	C5	C6	PEV	PEV
topdiam	0.66	−0.68					90.6	90.9
length	0.68	−0.68					92.3	92.5
moist		−0.52		0.84	0.06		97.6	97.8
testsg		−0.42	0.29	0.81		−0.23	**97.6**	97.5
ovensg		0.25	0.56			−0.69	86.1	86.8
ringtop	0.40		0.82	0.16			**86.5**	86.4
ringbut	0.75		0.58		−0.16		**92.8**	92.7
bowmax	0.76		−0.28				66.2	68.4
bowdist	0.72	−0.33					62.5	64.0
whoris	0.80				−0.48		86.9	87.2
clear	0.11		−0.11	0.11	0.96		**96.5**	95.9
knots	−0.46	−0.44	0.36		0.13	0.53	**82.7**	80.4
diaknot	−0.39	−0.69	−0.28	−0.32		−0.30	90.0	90.7
PEV	28.8	17.0	13.2	11.6	9.3	7.0	86.8	87.0
PEV_{PCA}	32.5	18.3	14.5	8.5	7.0	6.3	87.0	

optimal one with the average of $\text{AAD}(\mathbf{A}_{ck}, \mathbf{A}_c)$ being $D_{\text{LM}} = 0.151$. The resulting PEV and BIC values are plotted against Card(**A**) in Fig. 1. For choosing Card(**A**), we use BIC which showed the good performances for the cases of $n > p$ in the simulation study.

BIC showed the least value for Card(**A**) = 40 among all possible cardinalities. The corresponding loadings are shown in Table 4. The resulting total PEV 86.8 is found to be almost equivalent to the PEV 87.0 for the standard PCA: the USLPCA solution approximates the data very well, nearly as PCA does, and is more interpretable with $13 \times 6 - 40 = 38$ vanishing loadings. Bold font is used in Table 4 for the PEV for variables which exceeds the corresponding ones for PCA: it is noted that five variables are better explained by the sparse USLPCA components, with PEV for the other variables not very different between USLPCA and PCA.

Although the solution in Table 4 is optimal according to the BIC-based cardinality selection, the 40 nonzero loadings still seem too many to capture quickly the underlying variable-component relationships. This motivates us to choose a sparser solution using PEV rather than BIC. A procedure for the choice is trying a so-called scree test for the PEV plot in Fig. 1, i.e., to find the Card($\mathbf{A}$) value at which the increment in PEV begins to be less pronounced. However, such values are found at several points, among which we cannot choose the best one. In place of this approach, it can be considered to use a benchmark PEV value. As the value we choose the PEV 80 which is the integer $\times$ 10 % closest to the PEV 87.0 for the standard PCA, to select the solution of Card($\mathbf{A}$) = 17 with its PEV nearly greater than 80. The solution is depicted in Table 5, where the variables are found to be clearly clustered with every variable loading only one or two components.

In contrast to the USLPCA solutions with sparse loadings $\mathbf{A}$, the weights in $\mathbf{W}$ are sparsified in the existing procedures. As an example of the latter, Table 6 shows the sparse $\mathbf{W}$ obtained by Zou et al's (2013, Table 3) SPCA, which is related to USLPCA as mentioned in Sect. 1. All six components in Table 6 can be considered to correspond to those in Table 5, except the differences of a few nonzero loadings. However, the interpretation of $\mathbf{W}$ and $\mathbf{A}$ is quite different as discussed in Sect. 4.1. This difference is illustrated in the next paragraph.

For example, the sparse $\mathbf{W}$ in Table 6 shows that the PC scores for Component 3 (C3) are defined by four variables as

$$\text{C3} = .64 \times \text{ovensg} + .59 \times \text{ringtop} + .49 \times \text{ringbut} - .02 \times \text{diaknot}.$$

Table 5 USLPCA loadings with Card($\mathbf{A}$) = 17 for Pitprop data together with PCA's EV in the final row and column

Variable	USLPCA: Card($\mathbf{A}$) = 17							PCA
	C1	C2	C3	C4	C5	C6	PEV	PEV
topdiam	0.89						79.2	90.9
length	0.91						82.9	92.5
moist		0.96					92.4	97.8
testsg		0.94					88.6	97.5
ovensg			0.81				64.9	86.8
ringtop	0.37		0.79				76.7	86.4
ringbut	0.67		0.62				83.4	92.7
bowmax	0.61				−0.51		63.4	68.4
bowdist	0.80						63.5	64.0
whoris	0.75			0.44			75.1	87.2
clear				−0.98			95.3	95.9
knots					0.92		85.5	80.4
diaknot						−0.96	91.6	90.7
PEV	29.1	13.9	12.8	8.8	8.6	7.0	80.2	87.0
PEV_{PCA}	32.5	18.3	14.5	8.5	7.0	6.3	87.0	

Table 6 Zou et al.'s (2006) solution for Pitprop data, where the signs of weights in C1 and C6 have been changed from the original table

Variable	C1	C2	C3	C4	C5	C6
topdiam	0.48					
length	0.48					
moist		0.79				
testsg		0.62				
ovensg	−0.18		0.64			
ringtop			0.59			
ringbut	0.25		0.49			
bowmax	0.34	−0.02				
bowdist	0.42					
whoris	0.40					
clear				−1.00		
knots		0.01			−1.00	
diaknot			−0.02			−1.00

On the other hand, the USLPCA loadings **A** in Table 5 show that those same variables without diaknot are explained by Component 3 as

$$\left[\text{ovensg, ringtop, ringbut}\right] = [.81, .79, .62] \times \text{C3} + \mathbf{e}', \tag{7.1}$$

which is a row vector version of (4.4) and $\mathbf{e}'$ (1×7) expresses the part $\Sigma_{k \neq j} \mathbf{f}_k \mathbf{a}_k' + \mathbf{E}$ in (4.4). Equation (7.1) shows that Component 1 is interpreted as a common factor explaining those variables. The explanation power of Component 3 can be assessed by the corresponding PEV 12.8. Beside this column-wise interpretation of **A**, we can also interpret **A** in a row-wise manner. For example, we find in Table 4 that the variable "ringbut" is explained by Components 1 and 3 as

$$\text{ringbut} \ = 0.67 \times \text{C1} + 0.62 \times \text{C3} + \ \text{error}, \tag{7.2}$$

where the corresponding PEV shows that the 83.4 % of the variance of "ringbut" is explained by the two components.

The USLPCA loadings in Tables 4 and 5 are also the correlation coefficients as described in Sect. 4.2, since the data set is given as a correlation matrix. For example, it is found in Table 5 that the coefficient between "clear" and C4 is −0.98, which is close to the lower limit −1 and shows their very high negative correlation. On the other hand, the coefficient 0.37 between "ringtop" and C1 stands for that their relation is not high as 0.37 is rather close to the zero implying no correlation in the range of the zero to the upper limit one.

An anonymous reviewer was interested in the transition of nonzero loadings into zero ones and vice versa with respect to the increase of Card(**A**), i.e., the cases of

$$a_{ij}^{(c)} \neq 0 \quad \text{and} \quad a_{ij}^{(c+1)} = 0 \tag{7.3}$$

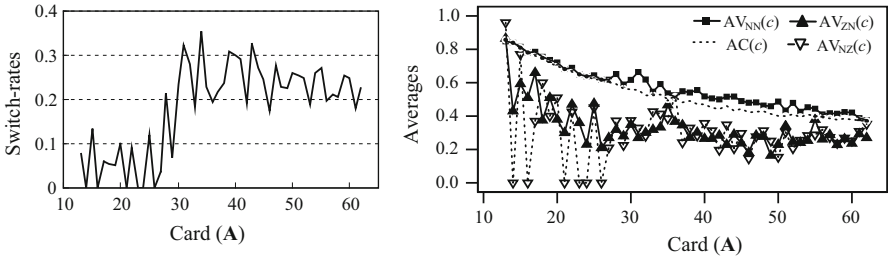

Fig. 2 Switch-rates and the averages of absolute loadings against the cardinality of **A**

and

$$a_{ij}^{(c)} = 0 \quad \text{and} \quad a_{ij}^{(c+1)} \neq 0, \tag{7.4}$$

where $a_{ij}^{(c)}$ denotes the (i, j) elements of the resulting **A** for $\mathrm{Card}(\mathbf{A}) = c$. We thus obtained the switch-rate $SR_c = q_c/c$, with q_c the number of the indices (i, j) with (7.3). It implies that $q_c + 1$ is the number of (i, j) with (7.4). The rate SR_c is plotted against $\mathrm{Card}(\mathbf{A}) = c$ left in Fig. 2. We find that SR_c is low (the same loadings steadily remain zero/nonzero) when $\mathrm{Card}(\mathbf{A})$ is smaller, in contrast to the cases of greater $\mathrm{Card}(\mathbf{A})$ where the switchover is remarkable.

The contrast may be related to the results shown right in Fig. 2, where four curves express the different averages of absolute nonzero loadings. In the figure, except for $\mathrm{Card}(\mathbf{A})$ being $c_{\min}$ and 36, the simple average of absolute nonzero loadings $\mathrm{AV}(c) = c^{-1}\Sigma_{i,j}\left|a_{ij}^{(c)}\right|$ is found to exceed the average $\mathrm{AV_{NZ}}(c) = q_c^{-1}\Sigma_{(i,j)\in NZ}\left|a_{ij}^{(c)}\right|$ of the nonzero $a_{ij}^{(c)}$ switched into zero, where NZ denotes the set of (i, j) satisfying (7.3). It shows that the nonzero $a_{ij}^{(c)}$ with smaller $\left|a_{ij}^{(c)}\right|$ tends to become zero. Further, we can find that $\mathrm{AV}(c)$ decreases with an increase in c and approaches $\mathrm{AV_{NZ}}(c)$ when c is greater. This suggests that, for greater c, a number of $\left|a_{ij}^{(c)}\right|$ are small enough to switch $a_{ij}^{(c)}$ into zero, which leads to the uncertainty for what nonzero loadings are turned into zero, i.e., higher switch-rates. The other two curves in the right figure express the average $\mathrm{AV_{ZN}}(c) = T_{ZN}^{-1}\Sigma_{(i,j)\in ZN}\left|a_{ij}^{(c+1)}\right|$ for the nonzero $a_{ij}^{(c+1)}$ switched from $a_{ij}^{(c)} = 0$, and $\mathrm{AV_{NN}}(c) = T_{NN}^{-1}\Sigma_{(i,j)\in NN}\left|a_{ij}^{(c)}\right|$ for the nonzero $a_{ij}^{(c)}$ being not switched. Here, ZN is the set of (i, j) with (7.4), and NN is the set of the (i, j) satisfying $a_{ij}^{(c)} \neq 0$ and $a_{ij}^{(c+1)} \neq 0$, with T_{ZN} and T_{NN} the numbers of (i, j) in ZN and NN, respectively. We can find that $\mathrm{AV_{NZ}}(c) < \mathrm{AV_{NN}}(c)$ and $\mathrm{A_{ZN}}(c) < \mathrm{A_{NN}}(c)$ except for $\mathrm{Card}(\mathbf{A}) = c_{\min}$: the absolute values of the switched loadings tend to be smaller than those of the ones which did not switch. It suggests that the interpretation for $\mathrm{Card}(\mathbf{A}) = c + 1$ cannot be changed drastically from that for $\mathrm{Card}(\mathbf{A}) = c$, if one note the loadings with great absolutes.

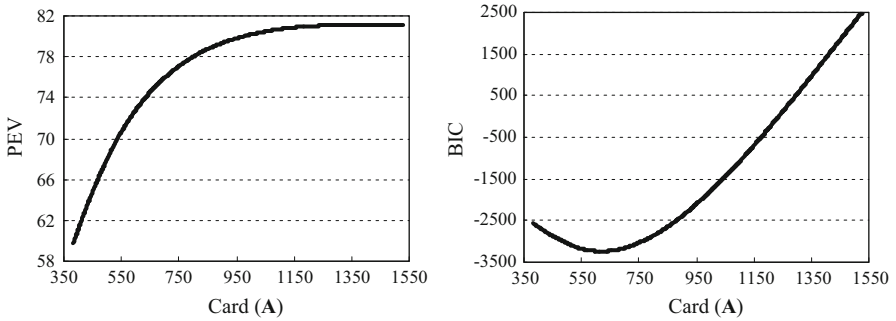

Fig. 3 Total PEV and BIC against the cardinality of **A** for gene expression data

7.2 Gene expression data

The second example is the yeast cell cycle data matrix of $n = 17$ time points (observations) by $p = 384$ genes (variables) presented by Yeung and Ruzzo (2001). The data matrix, publicly available at http://faculty.washington.edu/kayee/pca, have been first log-transformed and then standardized so that the column averages and variances are zero and one, respectively. The 384 genes are categorized into five phases of cell cycles, which suggests that $m = 5$ components corresponding to the five phases might be obtained. However, the preliminary trial detailed in Appendix "Component-wise constrained USLPCA" showed that the solution with $m = 5$ included a component whose PEV was very low. Such a trivial component was also extracted by Enki and Trendafilov (2012, p. 620), where the genes were classify into five groups, but one of them cannot be well related to the phases. We thus reduced m to 4 for performing USLPCA for the data set. Then, local minimizers were often obtained with $P_{\mathrm{LM}} = 0.771$ and they are not similar to the optimal one with $D_{\mathrm{LM}} = 0.119$. The resulting PEV and BIC values are plotted against Card(**A**) in Fig. 3, where PEV values are found to range from 60 to 81 %. For such PEV values, BIC is working better than AIC as suggested by the results for $p > n$ and the approximate $PEV = 60$ and 80 % in Sect. 6.4. In Fig. 3, BIC is found to give the least value for Card(**A**) = 616 among all possible cardinality, with the corresponding PEV 73.4.

The loadings for Card(**A**) = 616 are presented block-wise in Fig. 4. There, the blocks (genes × components) correspond to the five phases, with white, red, and blue standing for zero, positive values, and negative ones. The PEV for components 1, 2, 3, and 4 were 16.0, 37.0, 13.4, and 7.0, respectively. The solution is considered to be reasonable, as each phase has a unique feature of loadings: [a] The genes in Phases 1, 2, and 4 are positively loaded by Components 1, 2, and 3, respectively; [b] Phases 5 are characterized by positive loadings for Component 4 and negative ones for 2; [c] Phases 3 consists of the genes positively loaded by Component 2 or 3 and by both.

8 Discussion

In this paper, we proposed an unpenalized sparse PCA procedure USLPCA, in which component loadings rather than weights are sparsified without using penalty functions.

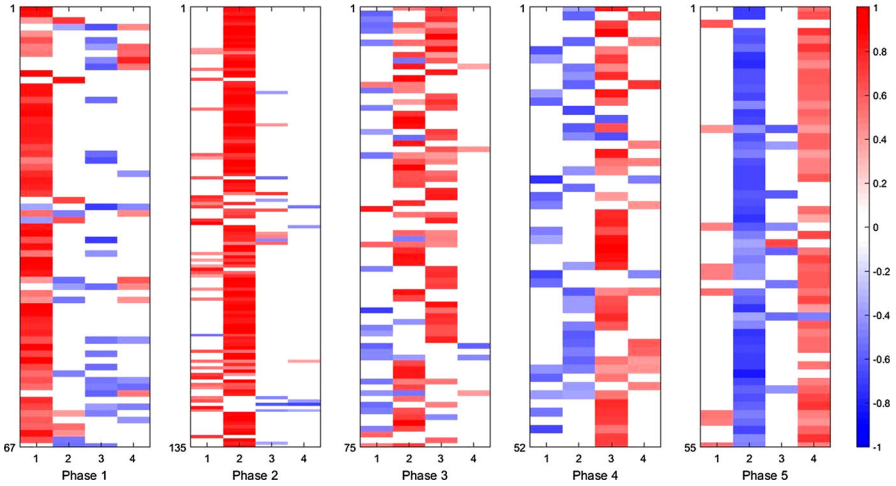

Fig. 4 Heat map of the loadings for gene expression data, which was produced by polarmap.m from MATLAB file exchange

In USLPCA, the loss function of PCA is minimized subject to the direct cardinality constraint on the loadings and the orthonormality constraint on the component score matrix. The latter condition makes it possible to decompose the loss function as a sum of two terms, one of which is irrelevant to the loadings, and the other one is a function easily minimized subject to the cardinality constraint. Using this decomposition, we formed the alternate least squares algorithm for USLPCA. This algorithm has two important features. First, the updating of the PC score matrix $\mathbf{F}$ can be avoided, and second, the algorithm can operate with either a data or covariance matrix as an input.

We also presented the cardinality selection procedure using AIC and BIC. This selection can be easily attained by considering all possible values of cardinality, owing to the cardinality being prespecified to be an integer in USLPCA. The simulation study shows that the true cardinality can be identified fairly well by BIC, though AIC overestimates the cardinality considerably.

In USLPCA, the total percentage of explained variances (PEV), the PEV index for components, and the one for variables are defined in the same form as the standard PCA. It facilitates comparing USLPCA solutions with the PCA one in the goodness-of-fit for a data set. As the total PEV of PCA is the upper limit of those of USLPCA, we can ascertain that a USLPCA solution is satisfactory if its total PEV is not very lower than that of PCA, as illustrated in the examples. There, it was also demonstrated that the variables can exist that are well explained by sparse USLPCA components than the PCA ones being not sparse.

One of the most critical differences of USLPCA to the existing sparse PCA is that the loadings in $\mathbf{A}$ are to be interpreted in the former, while the weights in $\mathbf{W}$ is of interest in the latter. As the weights must be inspected for finding how the PC scores in $\mathbf{F} = \mathbf{XW}$ are defined, the existing procedures are suitable in particular when PC scores are of interest. On the other hand, the loadings express how the original variables are regressed onto components. Thus, USLPCA is to be used when components are

treated as the factors explaining the variables. To what extent they are explained by components are assessed by the PEV indices discussed above.

One may be interested in modifying USLPCA for obtaining sparse $\mathbf{W}$, that is, minimizing the PCA loss function (1.1) over $\mathbf{W}$ and $\mathbf{A}$ subject to $\mathrm{Card}(\mathbf{W}) = c$. Unfortunately, this modification is not easy, as the key point of USLPCA is using the decomposition of (1.1) into (2.2), but this strategy is difficult to use for $\mathbf{W}$. That is, (2.2) reduces the cardinality constrained minimization of (1.1) over $\mathbf{A}$ into that of a simple function $g(\mathbf{A}) = \|\mathbf{B} - \mathbf{A}\|^2$. Unfortunately, such a simple function for $\mathbf{W}$ does not seem to exist.

In this paper, we did not present a systematic procedure for selecting dimensionality, i.e. the number of components (m). For this selection, we can perform the cardinality selection over different dimensionality to find the best combination of cardinality and dimensionality. For example, the combination with the least BIC can be chosen as the best one. Assessing such a procedure and considering other approaches remains for future studies.

Acknowledgments This work is supported by a Grant RPG-2013-211 from the Leverhulme Trust, UK and Grant (C)-26330039 from the Japan Society for the Promotion of Science.

9 Appendices

9.1 Multiple-runs procedure

To choose the number of the runs of the USLPCA algorithm, we assume that the value of the objective function (3.12) should decrease monotonically with the increase of $\mathrm{Card}(\mathbf{A}) = c$ in the constraint (1.3):

$$LS_{\mathrm{N}}(\mathbf{A}_c) \leq LS_{\mathrm{N}}(\mathbf{A}_{c-1}) \tag{9.1}$$

with $\mathbf{A}_c$ and $\mathbf{A}_{c-1}$ being the optimal solutions with $\mathrm{Card}(\mathbf{A}) = c$ and $\mathrm{Card}(\mathbf{A}) = c-1$ respectively. That is, we run the algorithm until $\mathbf{A}_c$ satisfying (9.1) is found. Let $\mathbf{A}_{ck}$ denote the solution of $\mathbf{A}$ resulting in the kth run. Then, our multiple-runs procedure for $\mathrm{Card}(\mathbf{A}) = c$ is described as follows:

1. Run the algorithm $K_c = 50$ times and set $\mathbf{A}_c = \mathrm{argmin}_{1 \leq k \leq K_c} LS_{\mathrm{N}}(\mathbf{A}_{ck})$.
2. Finish if $\mathbf{A}_c$ satisfies (9.1); otherwise go to 3.
3. Increase K_c by one, run the algorithm, and set $\mathbf{A}_c = \mathrm{argmin}_{1 \leq k \leq K_c} LS_{\mathrm{N}}(\mathbf{A}_{ck})$.
4. Finish if $\mathbf{A}_c$ satisfies (9.1) or $K_c = 1000$; otherwise back to 3.

Here, the number of runs is denoted by K_c with subscript c, as it cannot be the same among different c values.

When $K_c = 1$, the initial loading matrix $\mathbf{A}$ is taken to be the matrix of the standard PCA loadings. For $K_c > 1$, each element $\mathbf{A}$ is set to $a_{\max} \times u_{[-1,1]}$, with $u_{[-1,1]}$ a variable following the uniform distribution over the range $[-1, 1]$ and $a_{\max}$ being the maximum absolute value of the elements in the initial $\mathbf{A}$ for the first run.

The value of $LS_{\mathrm{N}}(\mathbf{A}_{c-1})$ must be known before the above multiple-runs procedure is carried out with $\mathrm{Card}(\mathbf{A}) = c$. Thus, the procedure (2.5) should be applied for

increasing sequence of values $c = c_{\min}, \ldots, c_{\max}$. We thus evaluate $I(c)$ with increasing c from $c_{\min}$ to $c_{\max}$ one by one. Only when $c = c_{\min}$, the multiple-runs procedure only consist of 1 and 2.

9.2 No correlation for components and errors

Here, we prove that no correlation exists between the two terms in the right-side hand of (4.4), i.e. the columns of $\mathbf{f}_j\mathbf{a}_j'$ is uncorrelated with those of $\mathbf{H}_{[j]} = \Sigma_{k\neq j}\mathbf{f}_k\mathbf{a}_k' + \mathbf{E}$.

The proof can be attained by showing that

[1] $\mathbf{f}_j\mathbf{a}_j'$ and $\mathbf{H}_{[j]}$ are column-centered;
[2] $(\mathbf{f}_j\mathbf{a}_j')'\mathbf{H}_{[j]} = \mathbf{O}$.

First, [1] follows from that $\mathbf{X}$ and $\mathbf{F}$ being column-centered implies $\mathbf{f}_j\mathbf{a}_j'$, $\Sigma_{k\neq j}\mathbf{f}_k\mathbf{a}_k'$, and $\mathbf{E} = \mathbf{X} - \mathbf{F}\mathbf{A}'$ being also column-centered.

Next, [2] can be proved by showing that $(\mathbf{f}_j\mathbf{a}_j')'\Sigma_{k\neq j}\mathbf{f}_k\mathbf{a}_k' = \mathbf{O}$ and $(\mathbf{f}_j\mathbf{a}_j')'\mathbf{E} = \mathbf{O}$. The former equality follows from that (1.2) implies $\mathbf{f}_j'\mathbf{f}_k = 0 (k \neq j)$. The left-side hand of the latter equality is expanded as

$$(\mathbf{f}_j\mathbf{a}_j')'\mathbf{E} = (\mathbf{f}_j\mathbf{a}_j')'(\mathbf{X} - \mathbf{F}\mathbf{A}') = \mathbf{a}_j\mathbf{f}_j'\mathbf{X} - \mathbf{a}_j\mathbf{f}_j'\mathbf{F}\mathbf{A}'. \tag{9.2}$$

Here, $\mathbf{a}_j\mathbf{f}_j'\mathbf{X}$ can be rewritten into

$$\mathbf{a}_j\mathbf{f}_j'\mathbf{X} = n\mathbf{a}_j\mathbf{b}_j' \tag{9.3}$$

with $\mathbf{b}_j = \mathbf{X}'\mathbf{f}_j$ the jth column of $\mathbf{B}$ defined in (2.3), while $\mathbf{a}_j\mathbf{f}_j'\mathbf{F}\mathbf{A}'$ can be expressed as

$$\mathbf{a}_j\mathbf{f}_j'\mathbf{F}\mathbf{A}' = n\mathbf{a}_j\mathbf{a}_j' \tag{9.4}$$

using (1.2). The Eq. (3.10) implies the equality between (9.2) and (9.3), which leads to that (9.2) equals $\mathbf{O}$. It completes the proof.

9.3 Likelihood for PCA model

Normality assumption (5.1) implies that the log likelihood for $\mathbf{X}$ is expressed as

$$l(\mathbf{F}, \mathbf{A}, \sigma^2) = -\frac{np}{2}\log 2\pi - \frac{np}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\|\mathbf{X} - \mathbf{F}\mathbf{A}'\|^2 \tag{9.5}$$

By solving the equation $dl(\mathbf{F}, \mathbf{A}, \sigma^2)/d\sigma^2 = 0$, we can find the ML estimate of σ^2 must satisfy $\sigma^2 = (np)^{-1}\|\mathbf{X} - \mathbf{F}\mathbf{A}'\|^2$. Substituting this into (9.5) lead to

$$l(\mathbf{F}, \mathbf{A}) = -\frac{np}{2}\log 2\pi + \frac{np}{2}\log np - \frac{np}{2}\log\|\mathbf{X} - \mathbf{F}\mathbf{A}'\|^2 - \frac{np}{2}, \tag{9.6}$$

whose part relevant to $\mathbf{F}$ and $\mathbf{A}$ is expressed as (5.2).

9.4 Component-wise constrained USLPCA

In the preliminary analysis of Yeung and Ruzzo's (2001) data, we used a version of USLPCA in which (1.3) is replaced by

$$\text{card}(\mathbf{a}_j) = c_j \tag{9.7}$$

for $j = 1, \ldots, m$, with $\text{card}(\mathbf{a}_j)$ the cardinality of the jth column of $\mathbf{A}$ and c_j an integer. The algorithm for this version is the same as in Sect. 3, except that (3.9) is replaced by setting

$$a_{ij} = \begin{cases} 0 & iff\ b_{ij}^2 \leq b_{[q]j}^2 \\ b_{ij} & \text{otherwise} \end{cases} \tag{9.8}$$

over $j = 1, \ldots, m$. Here, $b_{[q]j}^2$ denotes the q_jth smallest value among the squares of the elements in $\mathbf{b}_j$.

As described in Sect. 7.2, the 384 variables in the data are classified into the five clusters (genes). We performed the above version of USLPCA with $m = 5$ by setting the cardinality c_j in (9.6) to the number of the variables belonging to each cluster. If the five clusters correspond to the components columns, the nonzero loadings would be obtained that stand for the cluster memberships of the variables. However, the resulting solution did not have such a feature, with including a trivial component whose PEV was very low (3.2 %).

References

Akaike H (1974) A new look at the statistical model identification. IEEE Trans Automat Contr 19:716–723
d'Aspremont A, Bach F, Ghaoui LE (2008) Optimal solutions for sparse principal component analysis. J Mach Learn Res 9:1269–1294
Eckart C, Young G (1936) The approximation of one matrix by another of lower rank. Psychometrika 1:211–218
Enki DG, Trendafilov NT (2012) Sparse principal components by semi-partition clustering. Comput Stat 27:605–626
Izenman AJ (2008) Modern multivariate statistical techniques: regression, classification, and manifold learning. Springer, New York
Jeffers JNR (1967) Two case studies in the application of principal component analysis. Appl Stat 16:225–236
Jolliffe IT (2002) Principal component analysis, 2nd edn. Springer, New York
Jolliffe IT, Trendafilov NT, Uddin M (2003) A modified principal component technique based on the LASSO. J Comput Graph Stat 12:531–547
Journée M, Nesterov Y, Richtárik P, Sepulchre R (2010) Generalized power method for sparse principal component analysis. J Mach Learn Res 11:517–553
Schwarz G (1978) Estimating the dimension of a model. Ann Stat 6:461–464
Seber GAF (2008) A matrix handbook for statisticians. Wiley, Hoboken
Shen H, Huang JZ (2008) Sparse principal component analysis via regularized low rank matrix approximation. J Multivar Anal 99:1015–1034
SPSS Inc (1997) SPSS 7.5 statistical algorithms. SPSS Inc, Chicago
Takane Y (2014) Constrained principal component analysis and related techniques. CRC Press, Boca Raton
Trendafilov NT (2014) From simple structure to sparse components: a review. Comput Stat 29:431–454

Trendafilov NT, Adachi K (2015) Sparse versus simple structure loadings. Psychometrika. doi:10.1007/s11336-014-9416-y

Van Deun K, Wilderjans TF, van den Berg RA, Antoiadis A, Van Mechelen I (2011) A flexible framework for sparse simultaneous component based data integration. BMC Bioinformatics 12:448–464

Witten DM, Tibshirani R, Hastie T (2009) A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. Biostatistics 10:515–534

Yeung KY, Ruzzo WL (2001) Principal component analysis for clustering gene expression data. Bioinformatics 17:763–774

Zou DM, Hastie T, Tibshirani R (2006) Sparse principal component analysis. J Comput Graph Stat 15:265–286