Penalty-free sparse PCA

Kohei Adachi, *Osaka University*, adachi@hus.osaka-u.ac.jp
Nickolay Trendafilov, *Open University*, Nickolay.Trendafilov@open.ac.uk

Abstract. A drawback of the sparse principal component analysis (PCA) procedures using penalty functions is that the number of zeros in the matrix of component loadings as a whole cannot be specified in advance. We thus propose a new sparse PCA procedure in which the least squares PCA loss function is minimized subject to a pre-specified number of zeros in the loading matrix. The procedure is called unpenalized sparse matrix PCA (USMPCA), as it does not use a penalty function and obtains component loadings matrix-wise, i.e., simultaneously rather than sequentially. The key point of USMPCA is to use the fact that the PCA loss function can be decomposed into sum of two terms, one of them irrelevant to loadings, and another one being a function easily minimized under the considered cardinality constraint. This decomposition makes it possible to construct an efficient alternate least squares algorithm for USMPCA. Another useful feature is that the PC score matrix is column-orthonormal, which helps to define naturally the percentage of explained variance by the sparse PCs. USMPCA is illustrated with real data examples.

Keywords. Sparse component loadings, loss function decomposition, constrained matrix complexity.

1 Introduction

For an $n-$observations $\times$ $p-$variables column-centered data matrix X, principal component analysis (PCA) can be formulated as minimizing

$$f(F, A) = \|X - FA^\top\|^2 \tag{1}$$

over an $n \times m$ PC score matrix F and a $p \times m$ component loading matrix A, with $\| \|^2$ indicating the squared Frobenius norm and the number of components $m \leq \min(n, p)$. The resulting solution is interpreted by noting the loadings in A which quantify the relationships between the p variables and m components. It is desired for A to be sparse, i.e., to have a 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 A 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 [8]. Such procedures are called sparse PCA.

Almost all existing sparse PCA procedures are using penalized approaches: they are formulated by combining a PCA objective function with penalty functions that penalize A to have nonzero elements. Such examples are SCoTLASS [3], SPCA [10], and sPCA-rSVD [6], where the relative importance of penalty functions is controlled by tuning parameters. That is, they control the number of nonzero elements, which is called cardinality. Though a number of other penalized procedures have been developed for improving the preceding ones [4, 7, 8], they are formulated by the same format.

A common drawback of the penalized sparse PCA is that the appropriate value of the tuning parameter which corresponds to the desired cardinality is not obvious. Thus, the penalized sparse PCA is not convenient for users who wish to have a loading matrix with a specified number of zero elements. The procedures studied in [1] and [5] avoid such a difficulty. Their authors presented efficient heuristic algorithms called "greedy" search to find component loadings sequentially with direct cardinality constraint. In this paper, we also propose a directly constrained cardinality procedure without using a penalty function. However, our proposed procedure differs from the "greedy" search approaches in that all component are extracted simultaneously (not sequentially), i.e., F and A are obtained matrix-wise (not column-wise). We, thus, refer to our proposed procedure as unpenalized sparse matrix PCA (USMPCA). Moreover, the resulting PC scores are uncorrelated, which helps to define naturally the percentage of explained variance as described in Section 4.

2 Unpenalized Sparse Matrix PCA

In USMPCA, the PCA loss function (1) is minimized subject to the column-orthnormality condition for $n^{-1/2}F$ and the constraint on card(A) which denotes the cardinality of A. That is, USMPCA is formulated as

$$\min_{F,A} f(F,A) = \|X - FA^\top\|^2 \ , \text{ subject to } \frac{1}{n}F^\top F = I_m \text{ and } \mathrm{card}(A) = c \tag{2}$$

with I_m denoting the $m \times m$ identity matrix and c being a specified integer.

The key point of USMPCA is to use the fact that the orthonormality $\frac{1}{n}F^\top F = I_m$ allows the loss function (1) to be decomposed as

$$\|X - FA^\top\|^2 = \|X - FB^\top + FB^\top - FA^\top\|^2 = \|X - FB^\top\|^2 + n\|B - A\|^2 \ , \tag{3}$$

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

$$B = \frac{1}{n}X^\top F \ . \tag{4}$$

The decomposition (3), which is derived from $(X - FB^\top)(FB^\top - FA^\top)$ being the zero matrix, shows that a simple function $\|B - A\|^2$ is only relevant to A, which allows us to easily attain the cardinality constrained minimization of (1) as found in the next section.

3 Algorithm

The USMPCA problem (4) can be solved by alternately performing the two steps:

A-step minimizing (1) over A subject to $\mathrm{card}(A) = c$ with F being kept fixed;

F-step minimizing (1) over F subject to $\frac{1}{n}F^\top F = I_m$ with A kept fixed.

First, let us consider the A-step, which is equivalent to minimizing $g(A) = \|B - A\|^2$ under $\mathrm{card}(A) = c$, since of (3). Using $A = (a_{ij})$ and $B = (b_{ij})$, we can rewrite $g(A)$ as

$$g(A) = \|B - A\|^2 = \sum_{(i,j)\in O} b_{ij}^2 + \sum_{(i,j)\in O^\perp} (a_{ij} - b_{ij})^2 \geq \sum_{(i,j)\in O} b_{ij}^2 \ . \tag{5}$$

Here, O denotes the set of the $q = pm - c$ indexes (i, j)'s indicating the locations of the loadings a_{ij} to be zero, while the complement set $O^\perp$ contains the c (i, j)'s of nonzero a_{ij}. The inequality in (5) shows that $g(A)$ attains its lower limit $\sum_{(i,j)\in O} b_{ij}^2$ when the non-zero loadings a_{ij} with $(i, j) \in O^\perp$ are taken equal to the corresponding b_{ij}. Moreover, the limit $\sum_{(i,j)\in O} b_{ij}^2$ is minimal, when O contains the indexes for the q smallest b_{ij}^2 among all squared elements of B. Thus, $g(A)$ is minimized for $A = (a_{ij})$ being

$$a_{ij} = \begin{cases} 0 \text{ if } b_{ij}^2 \leq b_{[q]}^2 \\ b_{ij} \text{ otherwise} \end{cases} \ , \tag{6}$$

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

Next, let us consider the minimization in F-step. It is attained for

$$F = \sqrt{n}KL^\top = XAL\Lambda^{-1}L^\top, \tag{7}$$

where K and L are given by the singular value decomposition (SVD) of XA defined as

$$\frac{1}{\sqrt{n}}XA = K\Lambda L^\top \tag{8}$$

with $K^\top K = L^\top L = I_p$ and Λ a diagonal matrix. However, it is shown in the next paragraph that the update of F by (7) can be skipped.

Using $\frac{1}{n}F^\top F = I_m$ and (4), the loss function (1) can be expanded as

$$f(F, A) = \mathrm{tr}X^\top X + \mathrm{tr}AF^\top FA^\top - 2\mathrm{tr}X^\top FA = n\mathrm{tr}S + n\mathrm{tr}A^\top A - 2n\mathrm{tr}B^\top A \ , \tag{9}$$

with $S = \frac{1}{n}X^\top X$. Noting that (9) is a function of B and the use of (7) in (4) leads to

$$B = \frac{1}{n}X^\top XAL\Lambda^{-1}L^\top = SAL\Lambda^{-1}L^\top \ , \tag{10}$$

we can find that (1) or (9) is minimized for B given by (10) and this B is also used for (6): F may not be obtained in F-step. Moreover, the original data matrix X may not be available and only the sample covariance matrix S suffices for minimizing (1), since $L\Lambda^{-1}L^\top$ in (10) can be obtained through the eigenvalue decomposition (EVD)

$$A^\top SA = L\Lambda^2 L^\top \ , \tag{11}$$

following from (8): X is found to vanish in (9), (10), and (11).

It should be noted that the A resulting in (6) satisfies $\mathrm{tr}A^\top A = \mathrm{tr}B^\top A$. We can use it in (9) to find that the value of loss function (1) after the update (6) is expressed as

$$f(A) = n\mathrm{tr}S - n\mathrm{tr}A^\top A = n\mathrm{tr}S \times f_N(A) \ . \tag{12}$$

Here, $f_N(A) = 1 - \mathrm{tr}A^\top A/\mathrm{tr}S$ is normalized so as to take a value within $[0, 1]$, thus convenient for checking convergence. Thus, the USMPCA algorithm can be formed as follows:

1. Initialize A.
2. Perform EVD (11) to obtain B with (10).
3. Obtain A with (6).
4. Finish if $f_N(A) \leq \varepsilon$; otherwise go back to 2.

Here, $f_N(A)$ denotes the change in $f_N(A)$ from the previous round. In this paper, $\varepsilon = 0.1^7$ and the algorithm is repeated fifty times with random initialization. Among the resulting solutions, we select the one with the lowest $f_N(A)$ value as the optimal solution, in order to avoid local minimizers. After those procedures, F can be obtained using (7).

4 Percentages of Explained Variances

The loss function value (12) allows us to define the goodness of the resulting A as

$$\mathrm{PEV} = 100\mathrm{tr}A^\top A/\mathrm{tr}S \ , \tag{13}$$

with $\mathrm{tr}A^\top A = \frac{1}{n}\|FA^\top\|^2$ following from $\frac{1}{n}F^\top F = I_m$. The statistic (13) can be called total percentage of explained variance (PEV), since $\mathrm{tr}S$ in (13) is the total variance of the variables, while $\mathrm{tr}A^\top A = \frac{1}{n}\|FA^\top\|^2$ is the total variance of $FA^\top$, since (7) shows that F is column-centered as X is so.

The total PEV (13) can be decomposed as the sum of

$$\mathrm{PEV}(j) = 100a_j^\top a_j/\mathrm{tr}S \ , \tag{14}$$

over $j = 1, ..., m$. It serves as the PEV index for each component. On the other hand, the PEV for each variable is derived from the fact that (12) can be rewritten as $n\sum_{i=1}^p(s_{ii} - \|\widetilde{a}_i\|^2) = n\sum_{i=1}^p s_{ii}(1 - \|\widetilde{a}_i\|^2/s_{ii}) \geq 0$, with $\widetilde{a}_i^\top$ the ith row of A and s_{ii} the variance of variable i. It gives the percentage of $\|\widetilde{a}_i\|^2 = \frac{1}{n}\|F\widetilde{a}_i^\top\|^2$ to s_{ii},

$$\mathrm{PEV}[i] = 100\|\widetilde{a}_i\|^2/s_{ii} \ . \tag{15}$$

In the same forms as (13), (14), and (15), PEV indices are defined for the standard PCA, which is formulated as minimizing (1) with $\frac{1}{n}F^\top F = I_m$ and $A^\top A$ being a diagonal matrix. The same forms of definitions facilitate the comparison of solutions between USMPCA and PCA in goodness-of-fit. Since PCA is the best rank m approximation of X, the value of the total PEV (13) for USMPCS cannot exceed the one for PCA. However, if the former value is not substantially less than the latter, the USMPCS solution can be considered to be acceptable. It should be noted that USMPCS can be superior to PCA in (14) and (15), as illustrated in Section 6.1.

5 Nonzero Loadings as Covariances

The matrix B defined in (4) contains the covariances of p variables to m components, since X and F are column-centered. By taking this fact into account in (6), the nonzero loadings in A are found to equal the corresponding covariances in B: nonzero a_{ij} equals the covariance between variable i and component j. It implies that the nonzero loadings equal the correlation coefficients of variables to components, when the columns of X have unit variances or S is a correlation matrix, since of $\frac{1}{n}F^\top F = I_m$.

6 Two Examples

The first example is the Pitprop data set [2] given as the correlation matrix obtained from a 180×13 data matrix. We set $m = 6$ following the previous studies to perform USMPCA. The solution subject to $\text{card}(A) = pm/2 = 39$ is shown left in Table 1 with blank indicating zero loadings. There, the total PEV 86.7 is found to be almost equivalent to the PEV 87.0 for PCA: USMPCA approximated the data as well as PCA with a half of loadings vanishing in the former solution. We further performed USMPCA with decreasing card(A) one by one, to find that the PEV for the solution with $\text{card}(A) = 17$ nearly exceeded 80, a benchmark percentage not being very lower than 87.0 for PCA. That solution is shown right in Table 1. Bold font is used for the PEV for variables and components which exceed the corresponding ones for PCA. One notes that the USMPCA components with $j = 4, 5, 6$ explain more variance than the PCA ones.

Vars	USMPCA: card$(A) = 39$							USMPCA: card$(A) = 17$							PCA
	1	2	3	4	5	6	PEV	1	2	3	4	5	6	PEV	PEV
topdiam	.86	.40					90.4	.89						79.2	90.9
length	.90	.33					**92.0**	.91						82.9	92.5
moist		.98	-.10				97.5		.96					92.4	97.8
testsg		.90		-.40			97.5		.94					88.6	97.5
ovensg		-.17		-.93			88.7			.81				65.0	86.8
ringtop	.32	.19	.59	-.55	0.29		**87.1**	.37		.79				76.6	86.4
ringbut	.61		.61	-.41		-.14	**93.1**	.67		.62				83.4	92.7
bowmax	.54		.15		-0.60		67.8	.61				.51		63.4	68.4
bowdist	.75	.15			-0.20		62.9	.80						**63.5**	64.0
whoris	.66		.33		-0.38	-.42	86.8	.75			.44			75.1	87.2
clear		.15				.97	**96.9**				-.98			95.3	95.9
knots	-.11	.25		.25	0.80		77.5					-.92		**85.5**	80.4
diaknot	.15	.00	-.87	.10	0.31		88.6						-.96	**91.6**	90.7
PEV	25.8	17.1	**12.5**	**12.0**	**10.5**	**8.8**	86.7	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	32.5	18.3	14.5	8.5	7.0	6.3	87.0	

Table 1. USMPCA solutions for Pitprop data with PCA's PEV in the final row and column.

The variables are well clustered with every variable loading only one or two components. It makes sense to compare the USMPCA solutions with the classic (subjective) interpretation of the Pitprop component loadings [2], which is summarized in Table 2. The adopted notations mean that the first component is determined by topdiam, length, ringbut, bowmax, bowdist and whoris, the second – by moist and testsg, and etc. The ringbut value for component four in [2, Table 4, p.229] seems incorrect, by inspecting the corresponding eigenvalue. The corrected "classic" interpretation is given in [8], where ringbut is dropped off the fourth component.

Clearly, the USMPCA solution with $\text{card}(A) = 17$ suggests identical interpretation of the first three components as the one given in [2, p.230]. The fourth component is, indeed, a contrast, but between clear and whoris. The fifth component is also a contrast between knots

and bowmax, and the sixth component is a direct measure of diaknot (the average diameter of the knots in inches).

Vars	1	2	3	4	5	6
topdiam	x					
length	x					
moist		x				
testsg		x	x			
ovensg			x			x
ringtop	x		x			
ringbut	x			x		
bowmax	x					
bowdist	x					
whoris	x					
clear				x		
knots					x	
diaknot						x

Table 2. Classic interpretation of the Pitprop component loadings [2, p.229-30].

The second example concerns the gene expression data matrix of $n = 17$ time points by $p = 384$ genes presented by [9] and available at `http://faculty.washington.edu/kayee/pca`. The 384 genes are categorized into five phases of cell cycles, with each phase containing 67, 135, 75, 52, and 55 genes, respectively. It suggests $m = 5$, but this choice yielded one trivial component in preliminary trials. We thus reduced m to 4. For card(A), we first used the integer nearest to the one-third of pm, then increase card(A) one by one to find that the total PEV of the solution with card(A) $= 538 \cong 0.35pm$ nearly exceeds a benchmark 70, which is not considerably lower than the PEV 81.2 for PCA with $m = 4$. The resulting A with card(A) $= 538$ are presented block-wise in Figure 1. There, the blocks correspond to the five phases, with the block for the second one divided into two, and positive/negative nonzero loadings represented as filled squares/triangles, respectively. The solution is considered to be reasonable, as each phase has a unique feature of loadings: [a] Phases 1, 2, and 4 are characterized by positive loadings for 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.

7 Final Remarks

In this paper, we proposed the penalty-free sparse PCA procedure USMSPCA and presented its alternate least squares algorithm. An advantage of USMPCA over the penalized sparse PCA is that the cardinality of loadings can be set to a specified integer in advance. For that integer we can use the one conceived easily such as a half or the one-third of the number of loadings, which can be flexibly changed for finding a better solution, as illustrated in the examples. There, it was also illustrated that a solution obtained can be validated by comparing the PEV value with the corresponding one for the standard PCA. The reasonableness of this PEV comparison follows from that the PEV indices for USMPCA are defined in the same manner as in PCA.

Acknowledgement

This work is supported by a grant RPG-2013-211 from The Leverhulme Trust, UK.

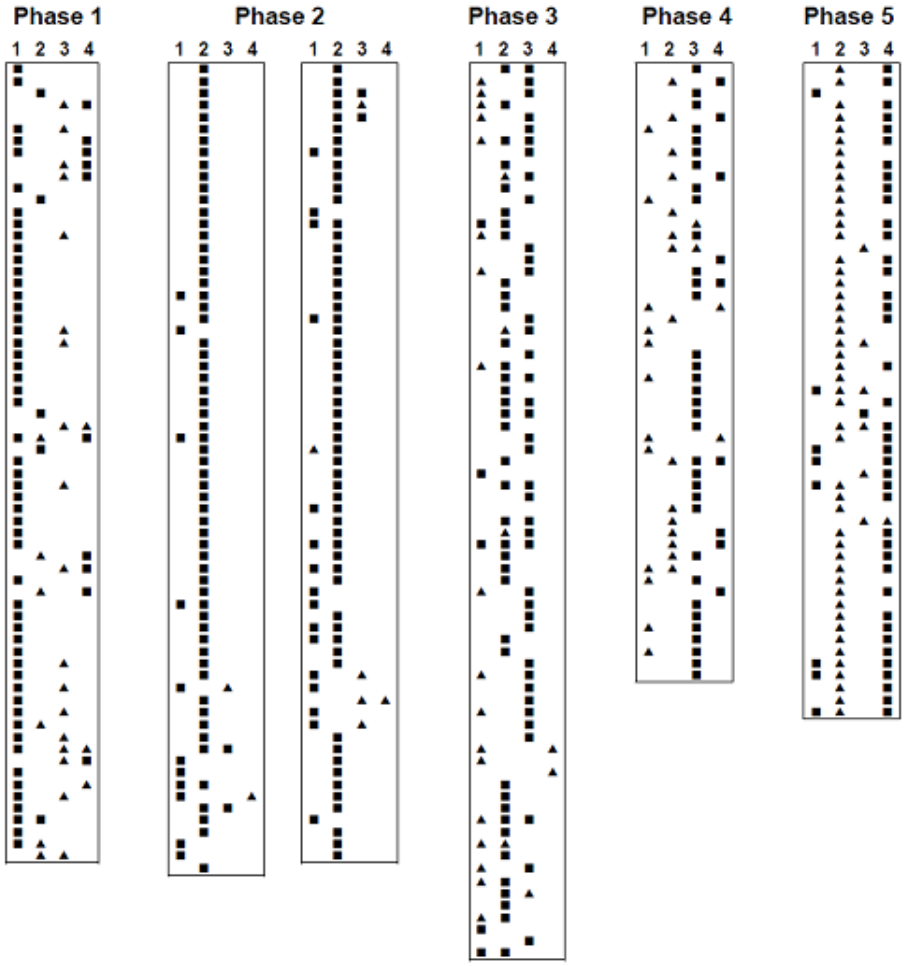

Figure 1. USMPCA solution for gene expression data with blank indicating zero

Bibliography

[1] d'Aspremont, A., Bach, F., & Ghaoui, L. E. (2008) Optimal solutions for sparse principal component analysis, *Journal of Machine Learning Research*, 9, 1269-1294.

[2] Jeffers, J. N. R. (1967). Two case studies in the application of principal component analysis. *Applied Statistics*, 16, 225-236.

[3] Jolliffe, I. T., Trendafilov, N. T., & Uddin, M. (2003). A modified principal component technique based on the LASSO. *Journal of Computational and Graphical Statistics*, 12, 531-547.

[4] Journe M, Nesterov Y, Richtrik P, & Sepulchre R (2010) Generalized power method for sparse principal component analysis, *Journal of Machine Learning Research*, 11, 517-553.

[5] Moghaddam, B., Weiss, Y., & Avidan, S. (2006). Spectral bounds for sparse PCA: Exact and greedy algorithms. *Advances in Neural Information Processing System*, 18, 915-922.

[6] Shen, H. & Huang, J. Z. (2008). Sparse principal component analysis via regularized low rank matrix approximation. *Journal of Multivariate Analysis*, 99, 1015 - 1034.

[7] Qi, X., Luo, R., & Zhao, H. (2013). Sparse principal component analysis by choice of norm. *Journal of Multivariate Analysis*, 114, 127-160.

[8] Trendafilov, N. T. (2013). From simple structure to sparse components: a review. *Computational Statistics*, published on line DOI:10.1007/s00180-013-0434-5.

[9] Yeung, K. Y., & Ruzzo, W. L. (2001). Principal component analysis for clustering gene expression data. *Bioinformatics*, 17, 763-774.

[10] Zou, D. M., Hastie, T., & Tibshirani, R. (2006). Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15, 265-286.