RENATO LEONI

Another Approach to Simple Principal Component Analysis

(*version January, 2014*)

Florence, Italy

Università di Firenze
Dipartimento di Statistica, Informatica, Applicazioni
"Giuseppe Parenti" (DiSIA)
Italia

leoni@disia.unifi.it

0 INTRODUCTION

Researchers are frequently faced with the task of analysing a data collection concerning a large number of quantitative variables measured on many individuals (units) and usually displayed in matrix form.

The aim of the analysis is often to find out patterns of relationships which can exist among variables.

The problem is that, given the data volume, this aim is not readily achieved.

A way to solve this problem is to perform a *principal component analysis* (PCA) of the data at hand.

Yet, since principal components are linear combinations of the variables with coefficients generally all different from zero, they can sometimes be difficult to interpret.

Attempts to overcome this drawback – which can be collected under the label *simple principal component analysis* (S_PCA) – are numerous.

In this paper, we will present another approach to S_PCA.

Essentially, the method we propose lies in forcing the first few principal vectors to be *sparse* and *orthonormal* and the corresponding sparse principal components to be *uncorrelated*.

The contents of the paper can be summarized as follows.

In Section 1, some preliminary concepts and notation are introduced. In Section 2, the approach to S_PCA we suggest is outlined. In Section 3, rules for building up and interpreting graphical representation of variables are given. In Section 4, a further remark is set out. Finally, in Appendix, the Matlab code for performing the necessary calculations and graphics is presented and results of a numerical example are shown.

1 SOME PRELIMINARY CONCEPTS AND NOTATION

Consider the *raw data matrix* (1)

(1) Uppercase and lowercase boldface letters represent, respectively, matrices and column vectors. A prime denotes transposition.

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \cdots & & \cdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix}$$

where x_{ij} $(j = 1, \ldots, p;\, i = 1, \ldots, n)$ denotes the value of the jth quantitative variable observed on the ith individual.

Notice that, setting $(j = 1, \ldots, p)$

$$\mathbf{x}_j = \begin{bmatrix} x_{1j} \\ \vdots \\ x_{nj} \end{bmatrix}$$

and $(i = 1, \ldots, n)$

$$\boldsymbol{x}_i = \begin{bmatrix} x_{i1} \\ \vdots \\ x_{ip} \end{bmatrix},$$

we can write

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \cdots & \mathbf{x}_p \end{bmatrix}$$

and

$$\mathbf{X}' = \begin{bmatrix} \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n \end{bmatrix}.$$

Considering the notation just introduced, we say that $\mathbf{x}_1, \ldots, \mathbf{x}_p$ and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ represent, respectively, the p variables and the n individuals.

Regarding $\mathbf{x}_1, \ldots, \mathbf{x}_p$ and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ as elements of R^n and R^p, respectively, R^n (*variable space*) and R^p (*individual space*) are equipped with a Euclidean metric.

In R^n the matrix (symmetric and positive definite) of the Euclidean metric – with respect to the basis formed by the n canonical vectors $\mathbf{u}_1, \ldots, \mathbf{u}_n$ – is

$$\mathbf{M} = \text{diag}\left(\frac{1}{n}, \ldots, \frac{1}{n}\right).$$

In R^p the matrix (symmetric and positive definite) of the Euclidean metric

– with respect to the basis formed by the p canonical vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_p$ – is

$$\mathbf{Q} = \mathrm{diag}\left(\frac{1}{\sigma_1^2}, \ldots, \frac{1}{\sigma_p^2}\right)$$

where σ_j^2 $(j = 1, \ldots, p)$ represents the *variance* of the jth variable.

Now, let

$$\mathbf{g}' = \left[\bar{x}_1 \cdots \bar{x}_p\right]$$

where $\bar{x}_j = \Sigma_i m_i x_{ij}$ is the (weighted) arithmetic mean of the variable $\mathbf{x}_j$.

The vector $\mathbf{g}$ is called the *mean vector* of the p variables $\mathbf{x}_1, \ldots, \mathbf{x}_p$ or the *barycentre* (*centroid*) of the n individuals $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$.

Next, assuming that $\mathbf{u}$ is a column vector of order n with elements all equal to 1, consider the *column centred data matrix*

$$\mathbf{Y} = \mathbf{X} - \mathbf{u}\,\mathbf{g}' = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix} - \begin{bmatrix} \bar{x}_1 & \cdots & \bar{x}_p \\ \cdots & \cdots & \cdots \\ \bar{x}_1 & \cdots & \bar{x}_p \end{bmatrix} = \begin{bmatrix} x_{11} - \bar{x}_1 & \cdots & x_{1p} - \bar{x}_p \\ \cdots & \cdots & \cdots \\ x_{n1} - \bar{x}_1 & \cdots & x_{n1} - \bar{x}_p \end{bmatrix}.$$

Then, setting $(j = 1, \ldots, p)$

$$\mathbf{y}_j = \begin{bmatrix} x_{1j} - \bar{x}_j \\ \vdots \\ x_{nj} - \bar{x}_j \end{bmatrix}$$

and $(i = 1, \ldots, n)$

$$\boldsymbol{y}_i = \begin{bmatrix} x_{i1} - \bar{x}_1 \\ \vdots \\ x_{ip} - \bar{x}_p \end{bmatrix}$$

we can write

$$\mathbf{Y} = \left[\mathbf{y}_1 \cdots \mathbf{y}_p\right]$$

and

$$\mathbf{Y}' = \left[\boldsymbol{y}_1 \cdots \boldsymbol{y}_n\right].$$

Taking into account the notation just introduced, we say that $\mathbf{y}_1, \dots, \mathbf{y}_p$ and $\boldsymbol{y}_1, \dots, \boldsymbol{y}_n$ represent, respectively, the p variables and the n individuals (measured in terms of deviations from the means).

Of course, the (weighted) arthmetic mean of $\mathbf{y}_j$ $(j = 1, \dots, p)$ is zero.

2 AN APPROACH TO S_PCA

Assuming that $rank(\mathbf{Y}) > 2$, suppose we have performed a PCA of the matrix $\mathbf{Y}$ obtaining the (first two) *principal components* (2)

$$(1) \qquad \tilde{\mathbf{y}}_1 = \mathbf{Y}\mathbf{Q}\,\tilde{\mathbf{c}}_1 = \mathbf{Y}\,\tilde{\mathbf{w}}_1 \quad , \quad \tilde{\mathbf{y}}_2 = \mathbf{Y}\mathbf{Q}\,\tilde{\mathbf{c}}_2 = \mathbf{Y}\,\tilde{\mathbf{w}}_2$$

where

- $\tilde{\mathbf{c}}_1$ and $\tilde{\mathbf{c}}_2$ are the (first two) *principal vectors*;
- $\tilde{\mathbf{w}}_1 = \mathbf{Q}\,\tilde{\mathbf{c}}_1$ and $\tilde{\mathbf{w}}_2 = \mathbf{Q}\,\tilde{\mathbf{c}}_2$ are the (first two) *principal factors* or *loadings*.

It should be remembered that

- $\tilde{\mathbf{c}}_1$ and $\tilde{\mathbf{c}}_2$ are orthonormal with respect to the metric represented by the matrix $\mathbf{Q}$;
- $\tilde{\mathbf{w}}_1$ and $\tilde{\mathbf{w}}_2$ are orthonormal with respect to the metric represented by the matrix $\mathbf{Q}^{-1}$;
- $\tilde{\mathbf{y}}_1$ and $\tilde{\mathbf{y}}_2$ are uncorrelated with variances given by the (first two) eigen-values $\tilde{\lambda}_1$ and $\tilde{\lambda}_2$.

The method we propose lies in forcing principal vectors to be *sparse* and *orthonormal* (which implies that principal factors are *sparse* and *ortho-normal* too), and the corresponding principal components to be *sparse* and *uncorrelated*.

---

(2) We limit ourselves to consider the first two principal components, but the approach we propose can be exended to three or more proncipal components.

The procedure we suggest consists of two steps.

Step 1. Notice that the first relationship in (1) can be rewritten as

(2) $$\tilde{\mathbf{y}}_1 = \mathbf{YQ}\,\tilde{\tilde{\mathbf{C}}}_{(1)}\,\mathbf{u}$$

where

$$\tilde{\tilde{\mathbf{C}}}_{(1)} = \mathrm{diag}(\tilde{\mathbf{c}}_1) \ .$$

Notice also that if we replace $\mathbf{u}$ with a *sparsity vector* $\mathbf{s}_1$ the relatonship (2) is generally no more exact but becomes

(3) $$\tilde{\mathbf{y}}_1 = \mathbf{YQ}\,\tilde{\tilde{\mathbf{C}}}_{(1)}\,\mathbf{s}_1 + \mathbf{e}_1$$

where $\mathbf{e}_1$ is a vector of residuals.

Suppose that we choose

(4) $$\bar{\mathbf{s}}_1 = \underset{\mathbf{s}_1}{argmin}\ (\tilde{\mathbf{y}}_1 - \mathbf{YQ}\,\tilde{\tilde{\mathbf{C}}}_{(1)}\,\mathbf{s}_1)'\mathbf{M}(\tilde{\mathbf{y}}_1 - \mathbf{YQ}\,\tilde{\tilde{\mathbf{C}}}_{(1)}\,\mathbf{s}_1)$$

under the constraints

(5) $$\mathbf{s}_1 \geq \mathbf{0} \quad , \quad \mathbf{u}'\mathbf{s}_1 \leq \mathrm{t} < \mathrm{p} \ .$$

As the *tuning parameter* t decreasis, some elements of $\bar{\mathbf{s}}_1$ are forced to zero, while the remaining elements are shrunken.

Hence, setting $\bar{\mathbf{c}}_1 = \tilde{\tilde{\mathbf{C}}}_{(1)}\,\bar{\mathbf{s}}_1$,

(6) $$\bar{\bar{\mathbf{c}}}_1 = \bar{\mathbf{c}}_1/(\bar{\mathbf{c}}_1'\mathbf{Q}\bar{\mathbf{c}}_1)^{1/2} \quad , \quad \bar{\bar{\mathbf{y}}}_1 = \mathbf{YQ}\,\bar{\bar{\mathbf{c}}}_1$$

represent, respectively, the *first normalized sparse principal vector* and the *first sparse principal component*.

Step 2. This step is similar to the previous one but, in order to achieve orthogonality of sparse principal vectors and lack of correlation of sparse principal components, more constraints are taken into account.

In details, notice that the second relationship in (1) can be rewritten as

$$(7) \qquad \tilde{\mathbf{y}}_2 = \mathbf{Y}\mathbf{Q}\,\tilde{\tilde{\mathbf{C}}}_{(2)}\,\mathbf{u}$$

where

$$\tilde{\tilde{\mathbf{C}}}_{(2)} = \mathrm{diag}(\tilde{\mathbf{c}}_2) \ .$$

Notice also that if we replace $\mathbf{u}$ with a *sparsity vector* $\mathbf{s}_2$ the relatonship (7) is generally no more exact but becomes

$$(8) \qquad \tilde{\mathbf{y}}_2 = \mathbf{Y}\mathbf{Q}\,\tilde{\tilde{\mathbf{C}}}_{(2)}\,\mathbf{s}_2 + \mathbf{e}_2$$

where $\mathbf{e}_2$ is a vector of residuals.

Suppose we choose

$$(9) \qquad \bar{\mathbf{s}}_2 = \underset{\mathbf{s}_2}{argmin}\ (\tilde{\mathbf{y}}_2 - \mathbf{Y}\mathbf{Q}\,\tilde{\tilde{\mathbf{C}}}_{(2)}\,\mathbf{s}_2)'\mathbf{M}(\tilde{\mathbf{y}}_2 - \mathbf{Y}\mathbf{Q}\,\tilde{\tilde{\mathbf{C}}}_{(2)}\,\mathbf{s}_2)$$

under the constraints

$$(10) \qquad \mathbf{s}_2 \geq \mathbf{0}\ ,\ \mathbf{u}'\mathbf{s}_2 \leq t < p\ ,\ \bar{\mathbf{c}}_1'\,\mathbf{Q}\,\tilde{\tilde{\mathbf{C}}}_{(2)}\,\mathbf{s}_2 = 0\ ,\ \tilde{\tilde{\mathbf{y}}}_1'\,\mathbf{M}\mathbf{Y}\mathbf{Q}\,\tilde{\tilde{\mathbf{C}}}_{(2)}\,\mathbf{s}_2 = 0\ .$$

Again, as the *tuning parameter* t decreasis, some elements of $\bar{\mathbf{s}}_2$ are forced to zero, while the remaining elements are shrunken.

Hence, setting $\bar{\mathbf{c}}_2 = \tilde{\tilde{\mathbf{C}}}_{(2)}\,\bar{\mathbf{s}}_2$,

$$(11) \qquad \tilde{\tilde{\mathbf{c}}}_2 = \bar{\mathbf{c}}_2/(\bar{\mathbf{c}}_2'\mathbf{Q}\bar{\mathbf{c}}_2)^{1/2} \quad , \quad \tilde{\tilde{\mathbf{y}}}_2 = \mathbf{Y}\mathbf{Q}\,\tilde{\tilde{\mathbf{c}}}_2$$

represent, respectively, the *second normalized sparse principal vector*, orthogonal to $\tilde{\tilde{\mathbf{c}}}_1$, and the *second sparse principal component*, uncorrelated with $\tilde{\tilde{\mathbf{y}}}_1$.

REMARK 1. The problems (4)-(5) and (9)-(10) are classical least squares problems under linear constraints for which exist efficient numerical algorithms to solve them.

REMARK 2. We have assumed that the tuning parameter t is the same in Step 1 and Step 2, but this is not necessary.

REMARK 3. Like any other approach to S_PCA, our approach is sub-optimal with respect to PCA, in the sense that sparse principal components have variances (depending on the tuning parameter t) less than those of ordinary principal components.

Besides orthonormality of sparse principal vectors and lack of correlation between sparse principal components, two more properties should be noted.

1. It results $(h = 1, 2)$

$$\mathbf{u}' \mathbf{M} \tilde{\tilde{\mathbf{y}}}_h = \mathbf{u}' \mathbf{M} \mathbf{Y} \mathbf{Q} \tilde{\tilde{\mathbf{c}}}_h = 0 \tag{12}$$

namely the (weighted) arthmetic mean of $\tilde{\tilde{\mathbf{y}}}_h$ is zero.

2. The orthogonal projection of $\mathbf{y}_j$ $(j = 1, \dots, p)$ on the subspace spanned by $\tilde{\tilde{\mathbf{y}}}_h$ $(h = 1, 2)$ is

$$\tilde{\tilde{\mathbf{y}}}_h (\tilde{\tilde{\mathbf{y}}}'_h \mathbf{M} \tilde{\tilde{\mathbf{y}}}_h)^{-1} \tilde{\tilde{\mathbf{y}}}'_h \mathbf{M} \mathbf{y}_j = \frac{\tilde{\tilde{\mathbf{y}}}_h}{(\tilde{\tilde{\mathbf{y}}}'_h \mathbf{M} \tilde{\tilde{\mathbf{y}}}_h)^{-1/2}} \sigma_j \frac{\tilde{\tilde{\mathbf{y}}}'_h \mathbf{M} \mathbf{y}_j}{(\tilde{\tilde{\mathbf{y}}}'_h \mathbf{M} \tilde{\tilde{\mathbf{y}}}_h)^{-1/2} \sigma_j} = \tilde{\tilde{\mathbf{y}}}^*_h \sigma_j r_{jh} \tag{13}$$

where $\tilde{\tilde{\mathbf{y}}}^*_h$ denotes the hth standardized sparse principal component and r_{jh} is the linear correlation coefficient between $\mathbf{y}_j$ and $\tilde{\tilde{\mathbf{y}}}_h$.

3 GRAPHICAL REPRESENTATION OF VARIABLES

A graphical representation of the p variables $\mathbf{y}_1, \dots, \mathbf{y}_p$ is usually obtained by their orthogonal projections on the subspace $S(\tilde{\tilde{\mathbf{y}}}^*_1, \tilde{\tilde{\mathbf{y}}}^*_2)$ spanned by the first two standardized sparse principal components $\tilde{\tilde{\mathbf{y}}}^*_1, \tilde{\tilde{\mathbf{y}}}^*_2$.

Taking into account of (13) and denoting by $\hat{\mathbf{y}}_j$ the orthogonal projection of $\mathbf{y}_j$ $(j = 1, \dots, p)$ on $S(\tilde{\tilde{\mathbf{y}}}^*_1, \tilde{\tilde{\mathbf{y}}}^*_2)$, we have

$$\hat{\mathbf{y}}_j = \tilde{\tilde{\mathbf{y}}}^*_1 \sigma_j r_{j1} + \tilde{\tilde{\mathbf{y}}}^*_2 \sigma_j r_{j2} .$$

Thus, the co-ordinates of $\hat{\mathbf{y}}_j$ relative to $\tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*$ are $(\sigma_j r_{j1}, \sigma_j r_{j2})$.

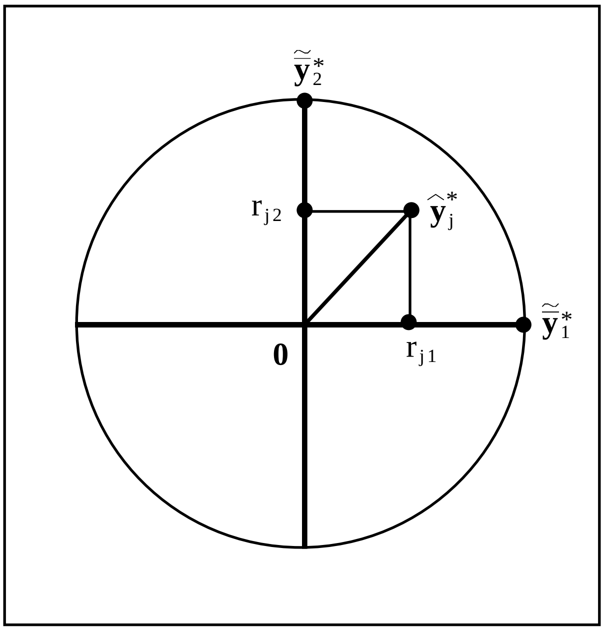
However, since we are mainly interested in representing linear correlations between pairs of variables or between a variable and a sparse principal component, and linear correlations are invariant if each variable is scaled by its standard deviation, it is more suitable to work with standardized variables.

In that case, the orthogonal projection $\hat{\mathbf{y}}_j^*$ of the standardized variable $\mathbf{y}_j^* = \mathbf{y}_j / \sigma_j$ $(j = 1, \ldots, p)$ on $S(\tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*)$ is given by

$$\hat{\mathbf{y}}_j^* = \tilde{\tilde{\mathbf{y}}}_1^* \, r_{j1} + \tilde{\tilde{\mathbf{y}}}_2^* \, r_{j2}$$

so that the co-ordinates of $\hat{\mathbf{y}}_j^*$ relative to $\tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*$ are (r_{j1}, r_{j2}) (see Fig. 1), and hence it is very easy to distinguish those variables which are the most correlated with a standardized sparse principal component and which play a significant role in its interpretation.

Fig. 1



Of course, each $\hat{\mathbf{y}}_j^*$ $(j = 1, \ldots, p)$ is inside a circle of centre **0** and radius 1 (the so-called *correlation circle*).

Moreover, the quality of representation of each variable on $S(\tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*)$ can be judged by means of the square cosine of the angle formed by $\mathbf{y}_j^*$ and $\hat{\mathbf{y}}_j^*$ which is given by $((\mathbf{y}_j^*)'\mathbf{M}(\mathbf{y}_j^*) = 1)$

$$QR(j\,; \tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*) = \frac{[(\mathbf{y}_j^*)'\mathbf{M}(\hat{\mathbf{y}}_j^*)]^2}{[(\mathbf{y}_j^*)'\mathbf{M}(\mathbf{y}_j^*)]\,[(\hat{\mathbf{y}}_j^*)'\mathbf{M}(\hat{\mathbf{y}}_j^*)]} = \frac{[(\mathbf{y}_j^*)'\mathbf{M}(\hat{\mathbf{y}}_j^*)]^2}{(\hat{\mathbf{y}}_j^*)'\mathbf{M}(\hat{\mathbf{y}}_j^*)}\,.$$

A high $QR(j\,; \tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*)$ – for example, $QR(j\,; \tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*) \geq 0.7$ – means that $\mathbf{y}_j^*$ is well represented by $\hat{\mathbf{y}}_j^*$; on the contrary, a low $QR(j\,; \tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*)$ means that the representation of $\mathbf{y}_j^*$ by $\hat{\mathbf{y}}_j^*$ is poor.

Notice that a more explicit expression of $QR(j\,; \tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*)$ can be obtained taking into account that ($(\tilde{\tilde{\mathbf{y}}}_h^*)'\mathbf{M}(\tilde{\tilde{\mathbf{y}}}_h^*) = 1$; $(\tilde{\tilde{\mathbf{y}}}_h^*)'\mathbf{M}(\tilde{\tilde{\mathbf{y}}}_{h^*}^*) = 0$ for $h \neq h^*$)

$$(\hat{\mathbf{y}}_j^*)'\mathbf{M}(\hat{\mathbf{y}}_j^*) = (\tilde{\tilde{\mathbf{y}}}_1^*\, r_{j1} + \tilde{\tilde{\mathbf{y}}}_2^*\, r_{j2})'\mathbf{M}(\tilde{\tilde{\mathbf{y}}}_1^*\, r_{j1} + \tilde{\tilde{\mathbf{y}}}_2^*\, r_{j2}) = r_{j1}^2 + r_{j2}^2$$

and

$$(\mathbf{y}_j^*)'\mathbf{M}(\hat{\mathbf{y}}_j^*) = (\mathbf{y}_j^*)'\mathbf{M}(\tilde{\tilde{\mathbf{y}}}_1^*\, r_{j1} + \tilde{\tilde{\mathbf{y}}}_2^*\, r_{j2}) = r_{j1}^2 + r_{j2}^2\,.$$

Thus,

$$QR(j\,; \tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*) = r_{j1}^2 + r_{j2}^2\,.$$

On the other hand, since $QR(j\,; \tilde{\tilde{\mathbf{y}}}_1^*, \tilde{\tilde{\mathbf{y}}}_2^*)$ also denotes the square distance of $\hat{\mathbf{y}}_j^*$ from the correlation circle centre, we can see that well-represented points lie near the circumference of the correlation circle.

Concluding, for well-represented variables we can visualize on the correlation circle:

- which variables are correlated among themselves and with each principal component;
- which variables are uncorrelated (orthogonal) among themselves and with each principal component.

4 A FURTHER REMARK

As mentioned in the Introduction, the literature on S_PCA is somewhat extensive (3). It can be added that such literature is rather heterogeneous for what concerns the formulation of the problem and/or the way to solve it.

Any how, suppose that the $a$ht ($a = 1, \ldots, A$) approach to S_PCA has produced the principal components $\tilde{\tilde{\mathbf{y}}}_1^{(a)}, \tilde{\tilde{\mathbf{y}}}_2^{(a)}$ (not necessarily uncorrelated) spanning the subspace $S(\tilde{\tilde{\mathbf{y}}}_1^{(a)}, \tilde{\tilde{\mathbf{y}}}_2^{(a)})$.

On the other hand, an ordinary PCA also produces the principal components $\tilde{\mathbf{y}}_1, \tilde{\mathbf{y}}_2$ (uncorrelated) spanning the subspace $S(\tilde{\mathbf{y}}_1, \tilde{\mathbf{y}}_2)$.

It seems quite natural to evaluate the performance of each approach to S_PCA comparing the subspaces $S(\tilde{\tilde{\mathbf{y}}}_1^{(a)}, \tilde{\tilde{\mathbf{y}}}_2^{(a)})$ and $S(\tilde{\mathbf{y}}_1, \tilde{\mathbf{y}}_2)$.

Setting

$$\tilde{\tilde{\mathbf{Y}}}^{(a)} = \left[\tilde{\tilde{\mathbf{y}}}_1^{(a)} \quad \tilde{\tilde{\mathbf{y}}}_2^{(a)} \right] \quad , \quad \tilde{\mathbf{Y}} = \left[\tilde{\mathbf{y}}_1 \quad \tilde{\mathbf{y}}_2 \right]$$

an index (among many others) which can serve the purpose − lying in the interval [0 , 1) − is

$$corr^2(\tilde{\tilde{\mathbf{Y}}}^{(a)}, \tilde{\mathbf{Y}}) = \frac{\left[trace(\tilde{\tilde{\mathbf{Y}}}^{(a)\prime} \mathbf{M} \tilde{\mathbf{Y}}) \right]^2}{\left[trace(\tilde{\tilde{\mathbf{Y}}}^{(a)\prime} \mathbf{M} \tilde{\tilde{\mathbf{Y}}}^{(a)}) \right] \left[trace(\tilde{\mathbf{Y}}' \mathbf{M} \tilde{\mathbf{Y}}) \right]} .$$

It should be noted that, when applyed to our approach, this index depends on the tuning parameter t and, as t increases, the index increases too.

(3) A partial list of contibutions to the theme is given below.

BIBLIOGRAPHY

Anaya-Izquierdo, K., Critchley, F. and Vines, K. (2008): *Orthogonal simple component analysis*. The Open University.

Cadima, J. F. C. L. and Jolliffe, I. T. (1995): *Loading and correlations in the interpretation of principle components*. Journal of Applied Statistics.

Cadima, J. F. C. L. and Jolliffe I. T. (2001): *Variable selection and the interpretation of principal subspaces*. Journal of Agricultural, Biological, and Environmental Statistics.

Chipman, H. A. and Gu, H. (2005): *Interpretable dimension reduction*. Journal of Applied Statistics.

Choulakian, V., D'Ambra, L. and Simonetti, B. (2006): *Hausman principal component analysis*. From Data and Information Analysis to Knowledge Engineering.

d'Aspremont, A., Ghaoui, L. E., Jordan, M. I. and Lanckriet, G. R. (2004): *A direct formulation for sparse PCA using semidefinite programming*. SIAM Review.

DeSarbo, W. S. and Hausman, R. E. (2005): *An efficient branch and bound procedure for restricted principal components analysis*. Data Analysis and Decision Support.

Gervini, D. and Rousson, V. (2004): *Criteria for evaluating dimension-reducing components for multivariate data*. The American Statistician.

Gragn, D. and Trendafilov, N. T. (2010): *Sparse principal components based on clustering*. The Open University.

Hausman, R. (1982): *Constrained multivariate analysis*. Studies in the Management Sciences.

Johnstone, I. M. and Lu, A. Y. (2009): *On consistency and sparsity for principal components analysis in high dimensions (with discussion)*. Journal of the American Statistical Association.

Jolliffe, I. T. (1995): *Rotation of principal components: choice of normalization constraints*. Journal of Applied Statistics.

Jolliffe, I. T. (2002). *Principal component analysis*. Springer Verlag.

Jolliffe, I. T., Trendafilov, N. T. and Uddin, M. (2003): *A modified principal component technique based on the LASSO*. Journal of Computational and Graphical Statistics.

Jolliffe, I. T. and Uddin, M. (2000): *The simplified component technique: An* *alternative to rotated principal components*. Journal of Computational and Graphical Statistics.

Journèe, M., Nesterov, Y., Richtàrik, P. and Sepulchre, R. (2010): *Generalized power method for sparse principal component analysis*, Journal of Machine Learning Research.

Rousson, V. and Gasser, T. (2004): *Simple component analysis*. Journal of the Royal Statistical Society, Series C (Applied Statistics).

Vines, S. K. (2000): *Simple principal components*. Applied Statistics.

Witten, D. M., Hastie, T. and Tibshirani, R. (2009): *A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis*. Biostatistics.

Zou, H., Hastie, T. and Tibshirani, R. (2006): *Sparse principal component analysis*. Journal of Computational and Graphical Statistics.

Appendix

- Matlab Code
- A numerical example: 1992 Olimpic Games (decathlon)

MATLAB CODE

```
%
%
%
function [SPV1,SPC1,SPV2,SPC2] = S_PCA(Y,PV,PC,labu,labv,labc);
%
% SPARSE PRINCIPAL COMPONENT ANALYSIS (S_PCA)
%
% The program requires the following inputs:
%
%     Y:    mean centred data matrix of order (n,p) where
%           n = number of units (individuals)
%           p = number of variables
%
%     PV:   matrix of order (p,2) of the first two principal
%           vectors resulting from an ordinary PCA
%
%     PC:   matrix of order (n,2) of the first two principal
%           components resulting from an ordinary PCA
%
%     labu: column vector of n unit (individuals) labels
%
%     labv: column vector of p variable labels
%
%     labc: column vector of p dummy labels (1),(2),...,(p)
%
% The matrix of order (n,n) representing the metric in Rn
% is M = (1/n)*In.
%
% The matrix of order (p,p) representing the metric in Rp
% is Q = diag(1/var(x(j).
%
% The program calls some utility functions (DISPTAB, RPS, RP).
% The code is given after the main program.
%
% WARNING! The program assumes that the rank r of the mean
% centred data matrix Y is at least two.
% Position, ticks, and so on, of each plot must be adjusted by
% user.
%
% Leoni, R., Department of Statistics,Informatics,Applications
% "Giuseppe Parenti"(DiSIA)
% University of Florence, Italy
% leoni@disia.unifi.it
%
%
%
```

MAIN PROGRAM

```
disp(' ')
disp('Y: Mean centred data matrix')
pause
disp(' ')
```

```
disptab(Y,labu,labv,'8.4')
pause
disp(' ')
disp('r: Rank of Y')
r = rank(Y);
pause
disp(' ')
disptab(r,[],[],'2.0')
pause
disp(' ')
[n,p] = size(Y);
V     = (1/n)*Y'*Y;
Q = diag(diag(eye(p))./diag(V));
disp('PV: Matrix of the first two principal vectors')
pause
disp(' ')
labc1  = ['PV1';'PV2'];
disptab(PV,labc,labc1,'8.4')
pause
disp(' ')
disp('PF: Matrix of the first two principal factors')
pause
disp(' ')
PF = Q*PV;
labc2  = ['PF1';'PF2'];
disptab(PF,labc,labc2,'8.4')
pause
disp(' ')
disp('PC: Matrix of the first two principal components')
pause
disp(' ')
labv1 = ['PC1';'PC2'];
disptab(PC,labu,labv1,'8.4')
pause
disp(' ')
disp('FIRST STEP')
pause
disp(' ')
disp('Input of the tuning parameter t')
disp(' ')
t = input('t = ')
pause
disp(' ')
disp('s1: First sparsity vector')
u = ones(p,1);
A = [-eye(p);u'];
pause
disp(' ')
a  = [zeros(p,1);t];
H1 = Y*Q*diag(PV(:,1));
options = optimset;
options = optimset(options,'Display', 'final');
options = optimset(options,'LargeScale', 'off');
```

```
s1 = lsqlin(H1,PC(:,1),A,a,[],[],[],[],[],options)
pause
disp(' ')
disp('SPV1: First sparse principal vector')
disp(' ')
SPV1  = diag(PV(:,1))*s1;
SPV1  = SPV1./sqrt(SPV1'*Q*SPV1);
labc3 = ['SPV1'];
disptab(SPV1,labc,labc3,'8.4')
pause
disp(' ')
disp('SPF1: First sparse principal factor')
disp(' ')
SPF1 = Q*SPV1;
labc4 = ['SPF1'];
disptab(SPF1,labc,labc4,'8.4')
pause
disp(' ')
disp('SPC1: First sparse principal component')
SPC1  = Y*Q*SPV1;
labv2 = ['SPC1'];
disp(' ')
pause
disptab(SPC1,labu,labv2,'8.4')
pause
disp(' ')
disp('SECOND STEP')
pause
disp(' ')
disp('s2: Second sparsity vector')
H2 = Y*Q*diag(PV(:,2));
Aeq  = [SPV1'*Q*diag(PV(:,2));(1/n)*SPC1'*Y*Q*diag(PV(:,2))];
beq = [0;0];
s2 = lsqlin(H2,PC(:,2),A,a,Aeq,beq,[],[],[],options)
pause
disp(' ')
disp('SPV2: Second sparse principal vector')
disp(' ')
pause
SPV2  = diag(PV(:,2))*s2;
SPV2  = SPV2./sqrt(SPV2'*Q*SPV2);
labc5 = ['SPV2'];
disptab(SPV2,labc,labc5,'8.4')
pause
disp(' ')
disp('SPF2: Second sparse principal factor')
disp(' ')
SPF2 = Q*SPV2;
labc6 = ['SPF2'];
disptab(SPF2,labc,labc6,'8.4')
pause
disp(' ')
disp('SPC2: Second sparse principal component')
```

```
disp(' ')
pause
SPC2  = Y*Q*SPV2;
labv2 = ['SPC2'];
disptab(SPC2,labu,labv2,'11.4')
pause
disp(' ')
disp('CheckOrthon: Are SPV1 and SPV2 orthonormal?')
SPV   = [SPV1 SPV2];
CheckOrthon = SPV'*Q*SPV
SPC   = [SPC1 SPC2];
pause
disp('CheckUncorr: Are SPC1 and SPC2 uncorrelated?')
pause
CheckUncorr = (1/n)*SPC'*SPC
pause
disp(' ')
disp('GRAPHICS')
pause
disp(' ')
disp('rj1,rj2: Correlations between variables and first two
SPC')
pause
disp(' ')
Ys = Y*pinv(diag(sqrt(diag(V))));
SPCs1 = SPC(:,1)./sqrt((1/n)*SPC(:,1)'*SPC(:,1));
SPCs2 = SPC(:,2)./sqrt((1/n)*SPC(:,2)'*SPC(:,2));
SPCs = [SPCs1 SPCs2];
corr = (1/n)*Ys'*SPCs;
labccz = ['rj1';'rj2'];
disptab(corr,labv,labccz,'8.4')
pause
disp(' ')
disp('QR(j;SPC1,SPC2): Quality of representation of variables
on correlation circle')
pause
disp(' ')
CORRs  = corr.^2;
U      = ones(2);
U      = triu(U);
CORRss = CORRs*U;
CORR = CORRss(:,2);
labaz = ['QR(j;SPC1,SPC2)'];
disptab(CORR,labv,labaz,'8.4')
pause
disp(' ')
disp('Representation of variables on correlation circle')
pause
figure
z1 = corr(:,1);
z2 = corr(:,2);
c = -pi:pi/8:pi;
plot(sin(c),cos(c),'-',z1,z2,'w')
```

```
text(z1,z2,labv)
title('Variables on Correlation Circle')
hold
a=([-1.5,1.5,-1.5,1.5]);
axis(a)
xlabel('SPC1 Axis')
ylabel('SPC2 Axis')
plot([a(1) 0 a(2)],[0 0 0],'-','LineWidth',2)
plot([0 0 0],[a(3) 0 a(4)],'-','LineWidth',2)
axis('equal')
grid
%
%
```

UTILITY FUNCTIONS

```
function disptab(mat,rowlab,collab,a)
%DISPTAB dispays a table with labels.
%Syntax: DISPTAB (mat,rowlab,collab,fmt)
%       The labels may be missing specifying [].
%       An optional format fmt may be indicated (ex: '12.4').
c = size(mat);
r = c(1); % number of rows
c = c(2); % number of cols
if ~isempty(collab)
    collab = [blanks(c)' collab];
    % Blank added to the left (better if collab are right
adjusted)
    d = size(collab);
    d = d(2);
else
    d = 1;
end
if nargin == 3
   a = '10.4';
end
b = eval(a);
m = max(round(b),d);
g = round(10*(b-round(b)));
a = ['%' int2str(m) '.' int2str(g) 'f'];
% Formatting the matrix
a = [rps(a, c) '\n'];
a = rps(a,r);
fmt = reshape(a,length(a)/r,r);
% Adding the format for the row labels (if any)
if ~isempty(rowlab)
   rowlab = [rowlab blanks(r)']; % blank added to be sure
   fmt = [rowlab  fmt']';
end
% Format for the header (if any)
if ~isempty(collab)
  b = reshape(blanks(c*(m-d)),c,m-d);
   f = [b collab]' ;
```

```
   f = f(:)';
   f = [f '\n'];
   a = size(rowlab);
   a = a(2);
f = [blanks(a) f];
   fprintf(1,f)
end
fprintf(1,fmt,mat')
function s = rps(s,n)
%RPS Replicates a string n times.
p = length(s);
s = s(rp(1:p,1,n));
function m = rp(x,n,p);
% Replicates a matrix x, n times across the rows
% and p times across the columns.
m=[];
for i = 1:p
    m = [m x];
end
x = m;
m = [];
for i = 1:n
    m = [m;x];
end
```

1992 OLIMPIC GAMES (decathon)

Y: Mean centred data matrix

	100	ljump	shot	hjump	400	110h	disc
1	-0.3889	0.7225	0.7554	0.0932	-1.0511	-1.0814	2.5721
2	-0.0789	0.3925	2.7254	0.0932	-0.0411	-0.4514	7.2521
3	-0.0089	0.1825	1.5054	0.0332	0.0589	-0.2714	6.6921
4	-0.0789	0.2725	-0.0446	0.0032	-1.4611	-0.1714	-3.2079
5	0.1911	-0.2075	2.2454	0.0332	0.2989	-0.2814	8.3121
6	-0.4189	0.3925	1.5654	0.1832	-1.3711	0.1886	-0.2879
7	0.2511	-0.0175	1.6654	0.1532	0.7389	-0.0114	8.1521
8	-0.1989	0.3425	0.5754	0.1532	-0.4011	-0.2514	2.6521
9	-0.0589	-0.1275	0.5454	0.0632	-0.7911	-0.0614	3.3321
10	-0.0789	-0.1275	1.7054	0.0332	-0.4611	-0.1414	5.9121
11	-0.6789	0.3825	0.7854	-0.0568	-1.7911	-0.3914	2.6321
12	0.1311	0.1825	1.4354	0.0032	0.9189	-0.2514	2.4121
13	-0.2189	0.1025	-0.2246	0.0032	-1.7211	-0.5214	0.1321
14	-0.1389	-0.0175	-0.6646	0.1232	-0.6711	-0.4314	2.1521
15	0.1411	0.1625	-0.2146	0.0632	-0.5111	-0.5314	-4.6479
16	-0.2189	0.4125	0.6354	0.0632	-1.5611	0.1486	1.0921
17	-0.3689	0.1525	1.5154	0.0032	-1.3711	-0.2714	2.2721
18	-0.0289	0.1125	-1.2146	-0.0268	-1.9711	-0.0914	-5.0079
19	-0.1989	0.2825	-0.1046	-0.0568	0.3589	0.2586	0.1121
20	-0.0389	0.1325	-1.2146	-0.0568	-1.0111	-0.2214	-1.2079
21	0.3911	-0.0775	-0.8546	-0.0868	0.3789	0.1386	0.8321
22	0.2411	-0.2075	-0.7846	-0.0568	-0.1011	-0.1714	-2.3279
23	0.2311	0.0025	-0.3746	-0.0568	1.8389	0.3886	-2.6279
24	0.2311	-0.6475	-0.5646	-0.1468	1.0589	0.0486	-2.5279
25	0.3811	-0.8175	-0.2046	-0.0868	2.0789	1.6486	-2.9079
26	-0.0289	-0.2375	-1.8046	-0.1468	1.7489	0.6686	-1.7479
27	0.4611	-0.7775	-5.1546	-0.0568	1.6989	0.9486	-17.8479
28	0.5811	-0.9675	-4.2346	-0.2668	5.1089	1.1686	-12.1679

	pole	jav	1500
1	0.4750	2.5214	-11.0718
2	0.2750	2.1014	-0.2618
3	0.4750	6.3214	-1.6518
4	0.6750	2.6014	-18.3218
5	0.2750	5.1014	-6.7618
6	-0.0250	-1.0986	-0.0718
7	0.0750	10.9614	10.6082
8	0.2750	-1.8386	3.9182
9	0.2750	5.6814	-9.1718
10	0.1750	4.8214	3.1082
11	0.1750	-2.7986	14.2782
12	0.2750	6.0214	-4.7918
13	-0.0250	5.3014	-9.0918
14	0.1750	-1.9786	-12.0918
15	0.4750	1.4614	-14.4618
16	-0.1250	-10.8786	-15.8618

```
17  -0.1250  -3.5986   6.6282
18  -0.1250   3.9814 -10.8618
19  -0.1250   3.9814   8.5482
20  -0.2250  -3.6386 -19.6618
21  -0.0250   2.2814 -26.2118
22  -0.1250   3.1414   4.2982
23  -0.1250  -8.6386   7.4882
24  -0.8250   1.2614  10.5282
25  -0.4250  -4.2986   8.3582
26  -0.6250  -4.2186  39.2882
27  -0.5250 -10.4786  17.4782
28  -0.6250 -14.0786  25.8182

r: Rank of Y

10

PV: Matrix of the first two principal vectors

           PV1     PV2
( 1)   -0.0819 -0.1233
( 2)    0.1408  0.1110
( 3)    0.5742 -0.5900
( 4)    0.0323 -0.0022
( 5)   -0.5094 -0.5681
( 6)   -0.1922 -0.0526
( 7)    1.7575 -2.3958
( 8)    0.1219 -0.0271
( 9)    1.5237 -2.9742
(10)   -3.5080 -1.9816

PF: Matrix of the first two principal factors

           PF1     PF2
( 1)   -1.0005 -1.5054
( 2)    0.8994  0.7087
( 3)    0.1948 -0.2001
( 4)    3.2510 -0.2237
( 5)   -0.2199 -0.2452
( 6)   -0.6339 -0.1736
( 7)    0.0564 -0.0769
( 8)    0.9296 -0.2065
( 9)    0.0447 -0.0873
(10)   -0.0172 -0.0097

PC: Matrix of the first two principal components

         PC1     PC2
 1    3.2952  0.9624
 2    2.3242 -0.8761
 3    1.8635 -1.2818
 4    1.6330  0.7660
 5    1.3489 -1.9917
```

```
 6   1.7672  0.9820
 7   1.2381 -2.6393
 8   1.6206  0.3966
 9   1.3240 -0.6404
10   1.2539 -1.1244
11   1.5746  1.5233
12   1.0236 -1.2592
13   1.3642  0.5793
14   1.2185  0.6300
15   1.1109  0.4211
16   0.8994  1.8830
17   1.0227  0.8439
18   0.2636  1.0402
19  -0.0732 -0.0129
20  -0.0030  1.3423
21  -0.5052 -0.5741
22  -0.8151 -0.3968
23  -1.9165  0.1326
24  -2.6981 -0.7765
25  -3.8355 -1.2831
26  -3.3641 -0.0249
27  -5.5874  1.4440
28  -7.3478 -0.0656

FIRST STEP

Input of the tuning parameter t

t = 5

s1: First sparsity vector

s1 =
  1.0818e-016
       1.4219
        1.234
     0.035278
      0.56484
       1.1699
  1.1358e-017
      0.57408
  1.2238e-017
  1.3465e-017

SPV1: First sparse principal vector

         SPV1
( 1)  -0.0000
( 2)   0.2453
( 3)   0.8680
( 4)   0.0014
( 5)  -0.3524
( 6)  -0.2754
```

```
( 7)   0.0000
( 8)   0.0857
( 9)   0.0000
(10)  -0.0000
```

SPF1: First sparse principal factor

```
         SPF1
( 1)  -0.0000
( 2)   1.5667
( 3)   0.2944
( 4)   0.1405
( 5)  -0.1521
( 6)  -0.9085
( 7)   0.0000
( 8)   0.6537
( 9)   0.0000
(10)  -0.0000
```

SPC1: First sparse principal component

```
       SPC1
 1   2.8204
 2   2.0266
 3   1.2819
 4   1.2335
 5   0.7306
 6   1.1225
 7   0.4314
 8   1.1967
 9   0.3256
10   0.6200
11   1.5650
12   0.9774
13   0.8141
14   0.4027
15   1.0714
16   0.8630
17   1.0590
18   0.1161
19   0.0326
20   0.0499
21  -0.5851
22  -0.4747
23  -0.8289
24  -1.9459
25  -3.4451
26  -2.2061
27  -4.2072
28  -5.047
```

```
SECOND STEP

s2: Second sparsity vector

s2 =

      0.96725
      0.17357
        0.828
  5.2248e-018
       1.3148
 -2.1784e-017
      0.77071
  1.2327e-016
      0.94567
 -8.7064e-017

SPV2: Second sparse principal vector

          SPV2
( 1)  -0.1302
( 2)   0.0210
( 3)  -0.5332
( 4)  -0.0000
( 5)  -0.8153
( 6)   0.0000
( 7)  -2.0153
( 8)  -0.0000
( 9)  -3.0697
(10)   0.0000

SPF2: Second sparse principal factor

          SPF2
( 1)  -1.5892
( 2)   0.1343
( 3)  -0.1808
( 4)  -0.0000
( 5)  -0.3519
( 6)   0.0000
( 7)  -0.0647
( 8)  -0.0000
( 9)  -0.0901
(10)   0.0000

SPC2: Second sparse principal component

           SPC2
 1       0.5548
 2      -0.9587
 3      -1.2568
 4       0.6574
 5      -1.8401
```

```
 6      1.0355
 7     -2.4776
 8      0.3934
 9     -0.4712
10     -0.8547
11      1.7005
12     -1.4654
13      0.5217
14      0.6139
15      0.1852
16      1.7474
17      0.9926
18      0.9396
19     -0.1194
20      1.0612
21     -0.8701
22     -0.3660
23      0.0021
24     -0.6749
25     -0.8346
26      0.2181
27      1.5958
28     -0.0298

CheckOrthon: Are SPV1 and SPV2 orthonormal?

CheckOrthon =

            1  4.8353e-017
  4.8353e-017            1

CheckUncorr: Are SPC1 and SPC2 uncorrelated?

CheckUncorr =

       3.2623   1.249e-016
  1.2403e-016       1.1202

GRAPHICS

rj1,rj2: Correlations between variables and first two SPC

           rj1     rj2
  100  -0.7330 -0.4804
ljump   0.9216  0.2259
 shot   0.8003 -0.3987
hjump   0.7389 -0.0510
  400  -0.8016 -0.3862
 110h  -0.8832 -0.0172
 disc   0.7334 -0.4959
 pole   0.8131 -0.1515
  jav   0.5674 -0.6049
 1500  -0.5597 -0.0483
```

QR(j;SPC1,SPC2): Quality of representation of variables on correlation circle

	QR(j;SPC1,SPC2)
100	0.7681
ljump	0.9004
shot	0.7995
hjump	0.5485
400	0.7918
110h	0.7803
disc	0.7838
pole	0.6840
jav	0.6879
1500	0.3156

Representation of variables on correlation circle