Acceleration and Re-start of the Alternating Least Squares Algorithm for Non-linear Principal Components Analysis

Masahiro Kuroda[1], Yuichi Mori[1], Masaya Iizuka[2] and Michio Sakakihara[1]

[1]Okayama University of Science, Okayama, JAPAN

[2]Okayama University, Okayama, JAPAN

Corresponding author: Masahiro Kuroda, email: kuroda@soci.ous.ac.jp

Abstract

In principal components analysis (PCA) of mixture of quantitative and qualitative data, we require to quantify qualitative data. The alternating least squares (ALS) algorithm can be used for PCA including such quantification. However, the ALS algorithm is linear convergence and its speed is very slow in the application of PCA to very large mixed data. In order to circumvent the problem of its slow convergence, Kuroda et al. (2011) provided an acceleration of the ALS algorithm using the vector ε (vε) algorithm of Wynn (1962). In this paper, we try to further increase the speed of convergence of the vε acceleration of the ALS algorithm using a re-starting procedure. Numerical experiments examine how the re-starting procedure works effectively to reduce the number of iterations and computation time of the vε acceleration of the ALS algorithm.

Keywords: acceleration of convergence, vector ε algorithm, re-starting procedure

1 Introduction

In principal components analysis (PCA) of mixture of quantitative and qualitative data, we require to quantify qualitative data. The PCA including such quantification is called non-linear PCA (Gifi, 1990). The alternating least squares (ALS) algorithm is used as the quantification method and alternates between quantification of qualitative data and computation of ordinary PCA of optimal scaling data. PRINCIPALS of Young et al. (1978) and PRINCALS of Gifi (1990) are popularly used as the ALS algorithm for non-linear PCA. However, the drawback of these algorithms is linear convergence and is very slow in the application of non-linear PCA to very large data. In order to circumvent such slow convergence problem, Kuroda et al. (2011) provided an acceleration method using the vector ε (vε) algorithm of Wynn (1962). Numerical experiments demonstrated that the vε acceleration of the ALS algorithm works well to accelerate the convergence of the sequence generated by the ALS algorithm.

In this paper, we try to re-start the ALS iterations using the vε accelerated sequence and generate a new ALS sequence that increases its speed of convergence. The re-starting procedure can reduce the number of iterations and computation time of the vε acceleration of the ALS algorithm.

The paper is organized as follows. In Section 2, we present PRINCIPALS used as the ALS algorithm for non-linear PCA and, in Section 3, show the vε acceleration of PRINCIPALS (vε-PRINCIPALS) that accelerates the convergence of PRINCIPALS. Section 4 proposes a re-starting procedure for speeding up the convergence of PRINCIPALS and describes the procedure of vε-PRINCIPALS with the re-starting. Numerical experiments in Section 5 examine the effect of the re-starting for vε-PRINCIPALS. In Section 6, we present our concluding remarks.

2 The ALS algorithm for non-linear PCA: PRINCIPALS

Let $\mathbf{X} = (\mathbf{X}_1\ \mathbf{X}_2\ \cdots\ \mathbf{X}_p)$ be an $n \times p$ matrix of n observations on p variables and be columnwise standardized. In PCA, we postulate that $\mathbf{X}$ is approximated by the following bilinear form:

$$\hat{\mathbf{X}} = \mathbf{Z}\mathbf{A}^\top, \tag{1}$$

where $\mathbf{Z} = (\mathbf{Z}_1\ \mathbf{Z}_2\ \cdots\ \mathbf{Z}_r)$ is an $n \times r$ matrix of n component scores on r $(1 \leq r \leq p)$ components, and $\mathbf{A} = (\mathbf{A}_1\ \mathbf{A}_2\ \cdots\ \mathbf{A}_r)$ is a $p \times r$ matrix consisting of the eigenvectors of $\mathbf{X}^\top\mathbf{X}/n$ and $\mathbf{A}^\top\mathbf{A} = \mathbf{I}_r$. Then we determine model parameters $\mathbf{Z}$ and $\mathbf{A}$ such that

$$\theta = \mathrm{tr}(\mathbf{X} - \hat{\mathbf{X}})^\top(\mathbf{X} - \hat{\mathbf{X}}) = \mathrm{tr}(\mathbf{X} - \mathbf{Z}\mathbf{A}^\top)^\top(\mathbf{X} - \mathbf{Z}\mathbf{A}^\top) \tag{2}$$

is minimized for the prescribed r components.

For observed data being a mixture of quantitative and qualitative data, ordinary PCA cannot be directly applied to such data. Optimal scaling is used to quantify the observed qualitative data and then ordinary PCA can be applied. Let $\mathbf{X}^* = (\mathbf{X}_1^*\ \mathbf{X}_2^*\ \cdots\ \mathbf{X}_p^*)$ be an $n \times p$ matrix of optimally scaled observations to satisfy restrictions

$$\mathbf{X}^{*\top}\mathbf{1}_n = \mathbf{0}_p \qquad \text{and} \qquad \mathrm{diag}\left[\frac{\mathbf{X}^{*\top}\mathbf{X}^*}{n}\right] = \mathbf{I}_p, \tag{3}$$

where $\mathbf{1}_n$ and $\mathbf{0}_p$ are vectors of ones and zeros of length n and p, respectively. In the presence of qualitative data, the optimization criterion (2) is replaced by

$$\theta^* = \mathrm{tr}(\mathbf{X}^* - \hat{\mathbf{X}})^\top(\mathbf{X}^* - \hat{\mathbf{X}}) = \mathrm{tr}(\mathbf{X}^* - \mathbf{Z}\mathbf{A}^\top)^\top(\mathbf{X}^* - \mathbf{Z}\mathbf{A}^\top). \tag{4}$$

In non-linear PCA, we determine the optimal scaling parameter $\mathbf{X}^*$, in addition to estimating $\mathbf{Z}$ and $\mathbf{A}$.

PRINCIPALS

Young et al. (1978) developed PRINCIPALS as the ALS algorithm for non-linear PCA with mixed measurement levels of single nominal, ordinal and numerical variables. PRINCIPALS alternates between ordinary PCA and optimal scaling and finds $\mathbf{X}^*$, $\mathbf{Z}$ and $\mathbf{A}$ by minimizing θ^* defined by Equation (4) under the restriction (3).

For the initialization of PRINCIPALS, we determine initial data $\mathbf{X}^{*(0)}$. The observed data $\mathbf{X}$ may be used as $\mathbf{X}^{*(0)}$ after it is standardized to satisfy the restriction (3). For given initial data $\mathbf{X}^{*(0)}$ with the restriction (3), PRINCIPALS updates each of the parameters $\mathbf{Z}$, $\mathbf{A}$ and $\mathbf{X}^*$ in turn, keeping the others fixed.

- *Model parameter estimation step*: Obtain $\mathbf{A}^{(t)}$ by solving

$$\left[\frac{\mathbf{X}^{*(t)\top}\mathbf{X}^{*(t)}}{n}\right]\mathbf{A} = \mathbf{A}\mathbf{D}_r, \tag{5}$$

where $\mathbf{A}^\top\mathbf{A} = \mathbf{I}_r$ and $\mathbf{D}_r$ is an $r \times r$ diagonal matrix of eigenvalues, and the superscript (t) indicates the t-th iteration. Compute $\mathbf{Z}^{(t)} = \mathbf{X}^{*(t)}\mathbf{A}^{(t)}$.

- *Optimal scaling step*: Calculate $\hat{\mathbf{X}}^{(t+1)} = \mathbf{Z}^{(t)}\mathbf{A}^{(t)\top}$ from Equation (1). Find $\mathbf{X}^{*(t+1)}$ such that

  $$\mathbf{X}^{*(t+1)} = \arg\min_{\mathbf{X}^*} \mathrm{tr}(\mathbf{X}^* - \hat{\mathbf{X}}^{(t+1)})^\top(\mathbf{X}^* - \hat{\mathbf{X}}^{(t+1)})$$

  for fixed $\hat{\mathbf{X}}^{(t+1)}$ under measurement restrictions on each of the variables. Scale $\mathbf{X}^{*(t+1)}$ by columnwise centering and normalizing.

3 The vε acceleration of PRINCIPALS: vε-PRINCIPALS

We briefly introduce the vε algorithm of Wynn (1962). The vε algorithm is utilized to speed up the convergence of a slowly convergent vector sequence and is very effective for linearly converging sequences.

Let $\{\mathbf{Y}^{(t)}\}_{t\geq 0} = \{\mathbf{Y}^{(0)}, \mathbf{Y}^{(1)}, \mathbf{Y}^{(2)}, \ldots\}$ be a linear convergent sequence generated by an iterative computational procedure. Then the vε algorithm transforms $\{\mathbf{Y}^{(t)}\}_{t\geq 0}$ into another vector sequence $\{\dot{\mathbf{Y}}^{(t)}\}_{t\geq 0}$ by using

$$\dot{\mathbf{Y}}^{(t-1)} = \mathbf{Y}^{(t)} + \left[\left[\mathbf{Y}^{(t-1)} - \mathbf{Y}^{(t)}\right]^{-1} + \left[\mathbf{Y}^{(t+1)} - \mathbf{Y}^{(t)}\right]^{-1}\right]^{-1}, \tag{6}$$

where $[\mathbf{Y}]^{-1} = \mathbf{Y}/||\mathbf{Y}||^2$ and $||\mathbf{Y}||$ is the Euclidean norm of $\mathbf{Y}$. When $\{\mathbf{Y}^{(t)}\}_{t\geq 0}$ converges to a limit point $\mathbf{Y}^{(\infty)}$ of $\{\mathbf{Y}^{(t)}\}_{t\geq 0}$, it is known that, in many cases, $\{\dot{\mathbf{Y}}^{(t)}\}_{t\geq 0}$ generated by the vε algorithm converges to $\mathbf{Y}^{(\infty)}$ faster than $\{\mathbf{Y}^{(t)}\}_{t\geq 0}$.

We assume that $\{\mathbf{X}^{*(t)}\}_{t\geq 0}$ generated by PRINCIPALS converges to a limit point $\mathbf{X}^{*(\infty)}$. Then vε-PRINCIPALS produces a faster convergent sequence $\{\dot{\mathbf{X}}^{*(t)}\}_{t\geq 0}$ of $\{\mathbf{X}^{*(t)}\}_{t\geq 0}$. Given $\mathbf{X}^{*(0)}$ and set a desired accuracy δ for convergence, vε-PRINCIPALS iterates the following steps:

- *PRINCIPALS step*: Compute $\mathbf{A}^{(t)}$ and $\mathbf{Z}^{(t)}$ in the *Model parameter estimation step* and determine $\mathbf{X}^{*(t+1)}$ in the *Optimal scaling step*.

- *Acceleration step*: Calculate $\dot{\mathbf{X}}^{*(t-1)}$ using $\{\mathbf{X}^{*(t-1)}, \mathbf{X}^{*(t)}, \mathbf{X}^{*(t+1)}\}$ from

  $$\mathrm{vec}\dot{\mathbf{X}}^{*(t-1)} = \mathrm{vec}\mathbf{X}^{*(t)} + \left[\left[\mathrm{vec}(\mathbf{X}^{*(t-1)} - \mathbf{X}^{*(t)})\right]^{-1} + \left[\mathrm{vec}(\mathbf{X}^{*(t+1)} - \mathbf{X}^{*(t)})\right]^{-1}\right]^{-1},$$

  where $\mathrm{vec}\mathbf{X}^* = (\mathbf{X}_1^{*\top}\ \mathbf{X}_2^{*\top} \cdots\ \mathbf{X}_p^{*\top})^\top$. If $||\mathrm{vec}(\dot{\mathbf{X}}^{*(t-1)} - \dot{\mathbf{X}}^{*(t-2)})||^2 < \delta$ then stop the algorithm.

Before starting the iteration, we determine initial data $\mathbf{X}^{*(0)}$ satisfying the restriction (3) and execute the *PRINCIPALS step* twice to generate $\{\mathbf{X}^{*(0)}, \mathbf{X}^{*(1)}, \mathbf{X}^{*(2)}\}$.

The vε acceleration is designed to generate $\{\dot{\mathbf{X}}^{*(t)}\}_{t\geq 0}$ converging to $\mathbf{X}^{*(\infty)}$. Thus the final value of $\{\dot{\mathbf{X}}^{*(t)}\}_{t\geq 0}$ is the estimate of $\mathbf{X}^*$ when vε-PRINCIPALS terminates. The estimates of $\mathbf{Z}$ and $\mathbf{A}$ can be calculated immediately from the estimate of $\mathbf{X}^*$ in the *Model parameter estimation step* of PRINCIPALS.

Note that $\dot{\mathbf{X}}^{*(t-1)}$ obtained at the t-th iteration of the *Acceleration step* is not used as the estimate $\mathbf{X}^{*(t+1)}$ at the $(t+1)$-th iteration of the *PRINCIPALS step*. Thus vε-PRINCIPALS speeds up the convergence of $\{\mathbf{X}^{*(t)}\}_{t\geq 0}$ without affecting the convergence of ordinary PRINCIPALS.

4 The re-starting procedure for vε-PRINCIPALS

During the iteration of vε-PRINCIPALS, two steps generate two parallel sequences, $\{\mathbf{X}^{*(t)}\}_{t\geq 0}$ in the *PRINCIPALS step* and $\{\dot{\mathbf{X}}^{*(t)}\}_{t\geq 0}$ in the *Acceleration step*. We find that $\{\dot{\mathbf{X}}^{*(t)}\}_{t\geq 0}$ enters rapidly in the neighborhood of $\mathbf{X}^{(\infty)}$. Then, we re-start the iterations of the *PRINCIPALS step* using a value $\dot{\mathbf{X}}^{*(s)}$ in $\{\dot{\mathbf{X}}^{*(t)}\}_{t\geq 0}$ and generate a new sequence $\{\mathbf{X}^{*(t)}\}_{t\geq s}$ that increases its speed of convergence. The rule for re-starting the *PRINCIPALS step* is given by

$$||\text{vec}(\dot{\mathbf{X}}^{*(t-1)} - \dot{\mathbf{X}}^{*(t-2)})||^2 \leq \delta_{Re}(> \delta), \tag{7}$$

and we reset $\delta_{Re} = \delta_{Re}/K$ at each re-starting, where K is an integer, such as 10^2. By using the rule (7), we can control the re-starting frequency. For example, let $\delta = 10^{-12}$, and the initial value of δ_{Re} set to 1 and $K = 10^2$. Then the re-starting procedure is performed at most six times.

When we find a value $\dot{\mathbf{X}}^{*(t-1)}$ satisfying the condition (7), the re-starting procedure treats $\dot{\mathbf{X}}^{*(t-1)}$ as the initial value instead of the current estimate $\mathbf{X}^{*(t+1)}$ and the *PRINCIPALS step* is re-started using $\dot{\mathbf{X}}^{*(t-1)}$. This re-starting algorithm proposed here is called vεR-PRINCIPALS.

For specified δ_{Re} and K in addition to $\mathbf{X}^{*(0)}$ and δ, vεR-PRINCIPALS performs the following steps:

- *PRINCIPALS step*: Compute $\mathbf{A}^{(t)}$ and $\mathbf{Z}^{(t)}$ in the *Model parameter estimation step* and determine $\mathbf{X}^{*(t+1)}$ in the *Optimal scaling step*.

- *Acceleration step*: Calculate $\dot{\mathbf{X}}^{*(t-1)}$ using $\{\mathbf{X}^{*(t-1)}, \mathbf{X}^{*(t)}, \mathbf{X}^{*(t+1)}\}$ from

$$\text{vec}\dot{\mathbf{X}}^{*(t-1)} = \text{vec}\mathbf{X}^{*(t)} + \left[\left[\text{vec}(\mathbf{X}^{*(t-1)} - \mathbf{X}^{*(t)})\right]^{-1} + \left[\text{vec}(\mathbf{X}^{*(t+1)} - \mathbf{X}^{*(t)})\right]^{-1}\right]^{-1}.$$

If $||\text{vec}(\dot{\mathbf{X}}^{*(t-1)} - \dot{\mathbf{X}}^{*(t-2)})||^2 < \delta$ then stop the algorithm.

re-starting: If $||\text{vec}(\dot{\mathbf{X}}^{*(t-1)} - \dot{\mathbf{X}}^{*(t-2)})||^2 < \delta_{Re}$ then update $\mathbf{X}^{*(t+1)} = \dot{\mathbf{X}}^{*(t-1)}$ and reset $\delta_{Re} = \delta_{Re}/K$.

5 Numerical experiments

We study how much faster vεR-PRINCIPALS converges than PRINCIPALS and vε-PRINCIPALS. For all experiments, we set $\delta = 10^{-12}$ for convergence of vε-PRINCIPALS and vεR-PRINCIPALS, and terminate PRINCIPALS when $|\theta^{(t+1)} - \theta^{(t)}| < 10^{-12}$, where $\theta^{(t)}$ is the t-th update of θ calculated from Equation (4). The maximum number of iterations is set to 10,000. We also specify the initial value of δ_{Re} being 1 and $K = 10^2$ for vεR-PRINCIPALS. In this setting, the re-starting procedure performs at most six times. We apply these algorithms to a random data matrix of 100 observations on 15 variables with 10 levels and measure the number of iterations and CPU time taken for $r = 2$. The procedure is replicated 100 times. All computations are performed with the statistical package `R` (R Development Core Team, 2008). CPU times (in seconds) are measured by the function `proc.time`[1].

[1] Times are typically available to 10 msec.

Table 1 is summary statistics of the numbers of iterations and CPU times of PRINCIPALS, vε-PRINCIPALS and vεR-PRINCIPALS from 100 simulated data. The table indicates that vε-PRINCIPALS and vεR-PRINCIPALS greatly reduce the number of iterations and CPU time of PRINCIPALS. We also find that, in the case where PRINCIPALS does not converge within 10,0000 iterations, two acceleration algorithms still converge. The advantage of these acceleration algorithms is very obvious.

Table 1: Summary of statistics of the numbers of iterations and CPU times of PRINCIPALS (ALS), vε-PRINCIPALS (vε) and vεR-PRINCIPALS (vεR) algorithms from 100 simulated data.

	The number of iterations			CPU time		
	ALS	vε	vεR	ALS	vε	vεR
Min.	136.0	44.00	31.0	1.640	0.630	0.510
1st Qu.	286.2	94.75	60.0	3.328	1.230	0.825
Median	403.0	122.50	78.0	4.660	1.545	1.045
Mean	598.6	181.76	110.7	6.841	2.224	1.414
3rd Qu.	657.0	183.00	123.0	7.468	2.245	1.562
Max.	10000.0	3326.00	1402.0	111.820	38.280	16.200

Table 2 is summary statistics of the iteration and CPU time speed-ups for comparing the speed of convergence of PRINCIPALS with that of vε-PRINCIPALS and vεR-PRINCIPALS. The iteration speed-ups are calculated by dividing the number of iterations required for PRINCIPALS by the number of iterations required for vε-PRINCIPALS and vεR-PRINCIPALS, respectively. The CPU time speed-ups are calculated similarly to the iteration speed-up. The table shows that, in mean, vεR-PRINCIPALS is smaller 5.3 times of iterations and 4.5 times of CPU time than those of PRINCIPALS. We also see that vεR-PRINCIPALS requires about 2/3 times of the number of iterations and CPU time of vε-PRINCIPALS.

Table 2: Summary of statistics of the iteration and CPU time speed-ups of vε-PRINCIPALS (vε) and vεR-PRINCIPALS (vεR) algorithms from 100 simulated data.

	Iteration speed-up		CPU time speed-up	
	vε	vεR	vε	vεR
Min.	1.832	2.358	1.771	2.159
1st Qu.	2.945	4.514	2.639	3.677
Median	3.260	5.187	2.981	4.384
Mean	3.364	5.281	3.073	4.576
3rd Qu.	3.828	5.904	3.424	5.062
Max.	5.415	10.842	4.969	9.722

6 Concluding remarks

In this paper, we provided a re-starting procedure for speeding up the convergence of the vε acceleration of the ALS algorithm. The rule of the re-starting is a very simple and it requires much less computational cost.

Numerical experiments demonstrated that vεR-PRINCIPALS works effectively to reduce the number of iterations and CPU time of PRINCIPALS and the re-starting procedure works very well to improve the speed of convergence of vε-PRINCIPALS. We note that the vε acceleration and the re-starting procedure are applicable to not only PRINCIPALS but also PRINCALS.

Acknowledgement

This research is supported by the Japan Society for the Promotion of Science (JSPS), Grant-in-Aid for Scientific Research (C), No 24500353.

References

[1] Gifi, A. (1990). *Nonlinear multivariate analysis*. John Wiley & Sons, Ltd., Chichester.

[2] Kuroda, M., Mori, Y., Iizuka, M. and Sakakihara, M. (2011). Accelerating the convergence of the EM algorithm using the vector epsilon algorithm. *Computational Statistics and Data Analysis,* 55, 143-153.

[3] R Development Core Team (2008). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL `http://www.R-project.org`.

[4] Young, F.W., Takane, Y. and de Leeuw, J. (1978). Principal components of mixed measurement level multivariate data: An alternating least squares method with optimal scaling features. *Psychometrika*, 43, 279-281.

[5] Wynn, P. (1962). Acceleration techniques for iterated vector and matrix problems. *Mathematics of Computation*, 16, 301-322.