

Convergence Analysis of a Kind of Deterministic Discrete-Time PCA Algorithm

Ze Zhu, Wanzhou Ye, Haijun Kuang

Department of Mathematics, College of Science, Shanghai University, Shanghai, China

Email: zhuze_96@163.com, wzhy@shu.edu.cn

How to cite this paper: Zhu, Z., Ye, W.Z. and Kuang, H.J. (2021) Convergence Analysis of a Kind of Deterministic Discrete-Time PCA Algorithm. *Advances in Pure Mathematics*, 11, 408-426.
<https://doi.org/10.4236/apm.2021.115028>

Received: April 12, 2021

Accepted: May 21, 2021

Published: May 24, 2021

Copyright © 2021 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

We proposed a generalized adaptive learning rate (GALR) PCA algorithm, which could be guaranteed that the algorithm's convergence process would not be affected by the selection of the initial value. Using the deterministic discrete time (DDT) method, we gave the upper and lower bounds of the algorithm and proved the global convergence. Numerical experiments had also verified our theory, and the algorithm is effective for both online and offline data. We found that choosing different initial vectors will affect the convergence speed, and the initial vector could converge to the second or third eigenvectors by satisfying some exceptional conditions.

Keywords

GALR PCA Algorithm, DDT Method, Global Convergence, Online Data

1. Introduction

At present, Principal Component Analysis is widely used in data processing, image recognition and even video recognition [1]. In general, QR decomposition and SVD decomposition are most commonly used when calculating principal components. However, when the data dimension is large, the calculation cost of the traditional decomposition matrix method is very high. And the approach is not suitable for real applications where data come incrementally or in the online way. The PCA algorithm can overcome these problems.

Regarding the PCA algorithm, the Hebbian learning rules were first proposed by Hebbian, but the Hebbian rules are unstable and easy to diverge. Oja [2] added a delayed item under the Hebbian learning rule premise, called Oja's learning rule. The Oja's rule could always converge to the principal component of the covariance matrix.

Many optimization methods could also obtain the principal components. Xu

[3] proposed the LMSE PCA algorithm to extract the principal components based on the least mean square error reconstruction. Chatterjee [4] proposed an unconstrained objective function and obtained various new PCA adaptive algorithms using gradient descent method, steepest descent method, conjugate gradient method, and Newton method. The results showed that other methods converge faster than the gradient descent method. When there are outliers in the data, Xu [5] used a Gibbs distribution derivation, added a coefficient to the learning rate to detect outliers, and adjusted their weights called the robust PCA algorithm. Then Song [6] used a Cauchy distribution to redefine the weights. Yang [7] proposed the FRPCA algorithm, which automatically selects relevant parameters compared to Xu [5].

All of these PCA learning algorithms are described by stochastic discrete time systems, and convergence is crucial for these algorithms to be useful in applications. It is not easy to study the behavior of these PCA algorithms directly. In order to indirectly prove the convergence of the Oja's learning algorithm, traditionally, the algorithms are first transformed into some deterministic continuous time (DCT) systems [8] [9] [10]. This transformation is based on a fundamental theorem for the study of the stochastic approximation algorithm [10], relating the discrete time stochastic model to a DCT formulation. The stochastic approximation theory requires strict assumptions, including the algorithm's learning rate to converge to 0. However, these conditions are difficult to achieve in practice and affect convergence speed [11] [12]. This makes it challenging to apply the DCT method to analyze the PCA algorithm's convergence in practice. Subsequently, Zufiria [12] proposed to study Oja's algorithm through a deterministic discrete time (DDT) system. This DDT system is obtained by applying conditional expectation to Oja's learning algorithm. It preserves the original learning algorithm's discrete time nature and can shed some light on the characteristics of the constant learning rate. Zhang [13] proposed that when $\eta = 0.618\lambda$, the convergence speed is the fastest. Based on this, Lv [14] proposed an adaptive non-zero learning rate algorithm and proved its convergence. [15] summarized this.

Although Lv [14] proposed an adaptive non-zero learning rate algorithm and proved its convergence. When we choose an inappropriate initial value will cause the algorithm to fail to iterate. This paper we proposed a generalized adaptive learning rate (GALR) PCA algorithm which can ensure the convergence of the algorithm.

2. Oja's Algorithm, the DCT Method and the DDT Method

2.1. Oja's Algorithm

Consider a linear single neuron network with input-output relation

$$y(k) = w^T(k)x(k), (k = 0, 1, 2, \dots) \quad (1)$$

where $y(k)$ is the network output, the input sequence

$\{x(k) | x(k) \in R^n (k = 0, 1, 2, \dots)\}$ is a zero mean stationary stochastic process, each $w(k) \in R^n (k = 0, 1, 2, \dots)$ is a weight vector. The basic Oja's PCA learning algorithm for weight updating is described as

$$w(k+1) = w(k) + \eta y(k) [x(k) - y(k)w(k)] \quad (2)$$

for $k \geq 0$, where $\eta(k) > 0$ is the learning rate. It is very difficult to study the convergence of (2) directly.

2.2. DCT Method

To indirectly interpret the convergence of (2), traditionally, the DCT method is used. This method can be described as follows: applying a fundamental stochastic approximation theorem [8] to (2), a DCT system can be obtained

$$\frac{dw(t)}{dt} = Cw(t) - w^T Cw(t) \cdot w(t), t \geq 0 \quad (3)$$

where $C = E[x(k)x^T(k)]$ denotes the correlation matrix of $\{x(k) | x(k) \in R^n (k = 0, 1, 2, \dots)\}$. The convergence of this DCT system is then used to interpret the convergence of (2). To transform (2) into the previous DCT system, the following restrictive conditions are required:

- 1) $\sum_{k=1}^{+\infty} \eta^2(k) < +\infty$;
- 2) $\sup_k E\left\{ \left| y(k) [x(k) - y(k)w(k)] \right|^2 \right\} < +\infty$.

By the condition 1), clearly, it implies that $\eta(k) \rightarrow 0$ as $k \rightarrow +\infty$. However, in many practical applications, $\eta(k)$ is often set to a small constant due to the round off limitation and speed requirement [11] [12]. The condition 2) is also difficult to be satisfied [12]. Thus, these conditions are unrealistic in practical applications and the previous DCT analysis is not directly applicable [11] [12].

2.3. DDT Method

To overcome the problem of learning rate converging toward zero, a DDT method is proposed in [12]. Unlike the DCT method to transform (2) into a continuous time deterministic system, the DDT method transform (2) into a discrete time deterministic system. The DDT system can be formulated as follows. Applying the conditional expectation operator $E\{w(k+1)/w(0), x(i), i < n\}$ to (2) and identifying the conditional expected value as the next iterate in the system [11], a DDT system can be obtained and given as

$$w(k+1) = w(k) + \eta [Cw(k) - w^T(k)Cw(k)w(k)] \quad (4)$$

for $k \geq 0$, where $C = E[x(k)x^T(k)]$ is the correlation matrix. The DDT method has some obvious advantages. First, it preserves the discrete time nature as that of the original learning algorithm. Second, it is not necessary to satisfy some unrealistic conditions of the DCT approach. Thus, it allows the learning rate to be constant. The main propose of this paper is to study the convergence of (4) subject to the learning rate η being some constant.

[13] proved the convergence of (4) when $\eta, \lambda_1, w(0)$ satisfied certain conditions. And it is proved that when $\eta = 0.618\lambda_1$, the algorithm has the fastest convergence speed. According to the proof of [13] [14] proposed an adaptive learning rate PCA algorithm and gave a theoretical proof:

$$w(k+1) = w(k) + \frac{\xi}{w^T(k)Cw(k)}(Cw(k) - w^T(k)Cw(k)w(k)) \quad (5)$$

3. The Generalized Adaptive Learning Rate (GALR) PCA Algorithm

In this section, details associated with a convergence analysis of the proposed learning algorithm will be provided systematically.

3.1. GALR PCA Algorithm Formulation

Although [14] has given the convergence proof of (5) under certain conditions, when the data is online, we cannot obtain the covariance matrix C in advance, which may lead to $w^T(0)Cw(0) = 0$ and affect the convergence process of the algorithm. We proposed a generalized adaptive learning rate (GALA) PCA algorithm

$$w(k+1) = w(k) + \frac{\xi}{w^T(k)Aw(k)}(Cw(k) - w^T(k)Aw(k)w(k)) \quad (6)$$

for $k \geq 0$, where $C = E[x(k)x^T(k)]$, $0 < \xi < 0.8$ is the correlation matrix and $A = aC + bI$, a, b are the coefficients and I is an identity matrix.

For convenience of analysis, next, some preliminaries are given. It is known that the correlation matrix C is a symmetric nonnegative definite matrix. There exists an orthonormal basis of R^n composed by eigenvectors of C . Let $\lambda_1, \dots, \lambda_n$ be all the eigenvalues of C order by $\lambda_1 \geq \dots \geq \lambda_n \geq 0$, Suppose that λ_p is the smallest nonzero eigenvalue, i.e., $\lambda_p > 0$ but $\lambda_j = 0 (j = p+1, \dots, n)$. Denote by σ , the largest eigenvalue of C . Suppose that the multiplicity of σ is $m (1 \leq m \leq p \leq n)$, then $\sigma = \lambda_1 = \dots = \lambda_m$. Suppose that $\{v_i | i = 1, \dots, n\}$ is an orthonormal basis in R^n such that each v_i is a unit eigenvector of C associated with the eigenvalue λ_i . Denote by V_σ the eigensubspace of the largest eigenvalue σ , i.e. $V_\sigma = \text{span}\{v_1, \dots, v_m\}$. Denoting by V_σ^T the subspace which is perpendicular to V_σ , clearly $V_\sigma^T = \text{span}\{v_{m+1}, \dots, v_n\}$.

Since the vector set $\{v_1, \dots, v_n\}$ is an orthonormal basis of R^n , for each $k \geq 0$, $w(k) \in R^n$ can be represented as

$$w(k) = \sum_{i=1}^n z_i(k)v_i \quad (7)$$

where $z_i(k) (i = 1, \dots, n)$ are some constants, and

$$Cw(k) = \sum_{i=1}^n \lambda_i z_i(k)v_i \quad (8)$$

$$w^T(k)Aw(k) = a \sum_{i=1}^n \lambda_i z_i^2(k) + b \sum_{i=1}^n z_i^2(k) = \sum_{i=1}^n (a\lambda_i + b) z_i^2(k) \tag{9}$$

Substitute (7) to (6), we have

$$z_i(k+1) = \left[1 + \frac{\xi}{w^T(k)Aw(k)} (\lambda_i - w^T(k)Aw(k)) \right] z_i(k), (i = 1, \dots, n) \tag{10}$$

for $k \geq 0$, where $0 < \xi < 0.8$.

Next, a lemma is presented that will be used for the subsequent algorithm analysis.

Lemma 1. It holds that

$$\begin{cases} \left[1 + \xi \left(\frac{\sigma}{s} - 1 \right) \right]^2 \geq 4(1 - \xi) \xi \sigma, & s > 0 \\ \left[1 + \xi \left(\frac{\sigma}{s} - 1 \right) \right]^2 \leq \max \left\{ \sigma, \left[1 + \xi \left(\frac{\sigma}{s_*} - 1 \right) \right]^2 s_* \right\}, & s \in [s_*, \sigma] \end{cases}$$

where $0 < \xi < 0.8$ and $s_* > 0$ is a constant.

Its proof can be found in [14].

3.2. Boundedness

We will analyze the boundedness of (6). The lower and upper bounds will be given in the following theorems.

Lemma 2. Suppose that $0 < \xi < 0.8$, if $0 < w^T(k)Aw(k) < \lambda_p$ then $w^T(k)Aw(k)$ is increasing.

Proof. If $w^T(k)Aw(k) < \lambda_p$ it can be checked that

$$\left[1 + \xi \left(\frac{\lambda_i}{w^T(k)Aw(k)} - 1 \right) \right]^2 \geq \left[1 + \xi \left(\frac{\lambda_p}{w^T(k)Aw(k)} - 1 \right) \right]^2 \geq 1 \tag{11}$$

for $k \geq 0$. Then, from (6), (8), (10), we have

$$\begin{aligned} w^T(k+1)Aw(k+1) &= \sum_{i=1}^n (a\lambda_i + b) z_i^2(k+1) \\ &= \sum_{i=1}^n \left[1 + \xi \left(\frac{\lambda_i}{w^T(k)Aw(k)} - 1 \right) \right]^2 (a\lambda_i + b) z_i^2(k) \\ &= \sum_{i=1}^p \left[1 + \xi \left(\frac{\lambda_i}{w^T(k)Aw(k)} - 1 \right) \right]^2 (a\lambda_i + b) z_i^2(k) \\ &\geq \sum_{i=1}^p \left[1 + \xi \left(\frac{\lambda_p}{w^T(k)Aw(k)} - 1 \right) \right]^2 (a\lambda_i + b) z_i^2(k) \\ &\geq \left[1 + \xi \left(\frac{\lambda_p}{w^T(k)Aw(k)} - 1 \right) \right]^2 w^T(k)Aw(k) \\ &\geq w^T(k)Aw(k) \end{aligned} \tag{12}$$

□

Theorem 1. If $w^T(0)Aw(0) > 0$, it holds that

$$w^T(k+1)A w(k+1) \geq 4(1-\xi)\xi\lambda_p > 0 \tag{13}$$

for all $k \geq 0$, where $0 < \xi < 0.8$.

Proof. It can be checked that

$$\left[1 + \xi \left(\frac{\lambda_i}{w^T(k)A w(k)} - 1 \right)\right]^2 \geq \left[1 + \xi \left(\frac{\lambda_p}{w^T(k)A w(k)} - 1 \right)\right]^2 \geq 0 \tag{14}$$

Similar to the proof of Lemma 2, we have

$$\begin{aligned} w^T(k+1)A w(k+1) &\geq \left[1 + \xi \left(\frac{\lambda_p}{w^T(k)A w(k)} - 1 \right)\right]^2 w^T(k)A w(k) \\ &\geq \min_{s>0} \left\{ \left[1 + \xi \left(\frac{\lambda_p}{s} - 1 \right)\right]^2 s \right\} \end{aligned} \tag{15}$$

for $k \geq 0$. By Lemma 1, it holds that

$$w^T(k+1)A w(k+1) \geq 4(1-\xi)\xi\lambda_p > 0 \quad \square$$

The above theorem shows that if $w^T(0)A w(0) \neq 0$, the trajectory starting from $w(0)$ is lower bounded, then it will never converge to zero.

Theorem 2. For any $w(0)$, there is a constant $H(w(0))$. The specific expression is

$$H(w(0)) = \left\{ \sigma, \left[1 + \xi \left(\frac{\sigma}{w^T(0)A w(0)} - 1 \right)\right]^2 w^T(0)A w(0) \right\} \tag{16}$$

for all $k \geq 0$. Then $w^T(k)A w(k) \leq H(w(0))$ is always true.

Proof. Let's proved it by mathematical induction. For all $k \geq 0$,

$0 < w^T(k)A w(k) \leq H(w(0))$ holds. Next, we proved

$0 < w^T(k+1)A w(k+1) \leq H(w(0))$ holds too. Discuss in two situations:

Case 1: $\sigma \leq w^T(k)A w(k) \leq H(w(0))$. Then

$$\left[1 + \xi \left(\frac{\lambda_i}{w^T(k)A w(k)} - 1 \right)\right]^2 \leq 1, (i = 1, \dots, p) \tag{17}$$

It follows that

$$\begin{aligned} w^T(k+1)A w(k+1) &= \sum_{i=1}^p \left[1 + \xi \left(\frac{\lambda_i}{w^T(k)A w(k)} - 1 \right)\right]^2 (a\lambda_i + b) z_i^2(k) \\ &\leq \sum_{i=1}^p (a\lambda_i + b) z_i^2(k) = w^T(k)A w(k) \leq H(w(0)) \end{aligned} \tag{18}$$

Case 2: $0 < w^T(k)A w(k) \leq \sigma$. Then

$$\left[1 + \xi \left(\frac{\lambda_i}{w^T(k)A w(k)} - 1 \right)\right]^2 \leq \left[1 + \xi \left(\frac{\sigma}{w^T(k)A w(k)} - 1 \right)\right]^2 \tag{19}$$

It follows that

$$\begin{aligned}
 w^T(k+1)Aw(k+1) &= \sum_{i=1}^p \left[1 + \xi \left(\frac{\lambda_i}{w^T(k)Aw(k)} - 1 \right) \right]^2 (a\lambda_i + b)z_i^2(k) \\
 &\leq \sum_{i=1}^p \left[1 + \xi \left(\frac{\sigma}{w^T(k)Aw(k)} - 1 \right) \right]^2 (a\lambda_i + b)z_i^2(k) \\
 &= \left[1 + \xi \left(\frac{\sigma}{w^T(k)Aw(k)} - 1 \right) \right]^2 \cdot w^T(k)Aw(k) \\
 &\leq \max_{0 < s \leq \sigma} \left\{ \left[1 + \xi \left(\frac{\sigma}{s} - 1 \right) \right]^2 s \right\}
 \end{aligned} \tag{20}$$

By lemma 1 and lemma 2, denote $s^* = \frac{\xi\sigma}{1-\xi}$

When $s^* > \sigma$

$$w^T(k+1)Aw(k+1) \leq \left[1 + \xi \left(\frac{\sigma}{w^T(0)Aw(0)} - 1 \right) \right]^2 w^T(0)Aw(0) \tag{21}$$

When $s^* < \sigma$

$$\begin{aligned}
 &w^T(k+1)Aw(k+1) \\
 &\leq \begin{cases} \sigma, & s^* < s < \sigma \\ \left[1 + \xi \left(\frac{\sigma}{w^T(0)Aw(0)} - 1 \right) \right]^2 \cdot w^T(0)Aw(0), & 0 < s < s^* \end{cases}
 \end{aligned} \tag{22}$$

So $w^T(k+1)Aw(k+1) \leq H(w(0))$ is always true. □

3.3. Global Convergence

Lemma 3. If $w(0) \notin V_\sigma^\perp$ there exists constants $\theta_1 > 0$ and $\Pi_1 \geq 0$ such that

$$\sum_{j=m+1}^n z_j^2(k) \leq \Pi_1 \cdot e^{-\theta_1 k} \tag{23}$$

for all $k \geq 0$, where

$$\theta_1 = \ln \left(\frac{\xi\sigma + (1-\xi)H(w(0))}{\xi\lambda_{m+1} + (1-\xi)H(w(0))} \right) > 0 \tag{24}$$

Proof. Since $w(0) \notin V_\sigma^\perp$ there must exist some $i(1 \leq i \leq m)$ such that $z_i(0) \neq 0$. Without loss of generality, assume that $z_1(0) \neq 0$.

From (10), it follows that

$$z_i(k+1) = \left[1 + \xi \left(\frac{\sigma}{w^T(k)Aw(k)} - 1 \right) \right] z_i(k), (i = 1, \dots, m) \tag{25}$$

$$z_j(k+1) = \left[1 + \xi \left(\frac{\lambda_j}{w^T(k)Aw(k)} - 1 \right) \right] z_j(k), (j = m+1, \dots, n) \tag{26}$$

By Theorem 2, $w^T(k)Aw(k)$ has an upper bound for all $k \geq 0$. Given

any $i, (i = 1, \dots, n)$, it holds that

$$\left[1 + \xi \left(\frac{\lambda_i}{w^T(k) A w(k)} - 1 \right) \right] = 1 - \xi + \xi \left(\frac{\lambda_i}{w^T(k) A w(k)} \right) \geq (1 - \xi) > 0 \quad (27)$$

for $k \geq 1$. Then from (25) and (26), for each $j (j = m + 1, \dots, n)$, it follows that

$$\begin{aligned} \left[\frac{z_j(k+1)}{z_1(k+1)} \right]^2 &= \left[\frac{1 + \xi \left(\frac{\lambda_j}{w^T(k) A w(k)} - 1 \right)}{1 + \xi \left(\frac{\sigma}{w^T(k) A w(k)} - 1 \right)} \right]^2 \cdot \left[\frac{z_j(k)}{z_1(k)} \right]^2 \\ &= \left[\frac{w^T(k) A w(k) + \xi (\lambda_j - w^T(k) A w(k))}{w^T(k) A w(k) + \xi (\sigma - w^T(k) A w(k))} \right]^2 \cdot \left[\frac{z_j(k)}{z_1(k)} \right]^2 \\ &\leq \left[\frac{\xi \lambda_j + (1 - \xi) H(w(0))}{\xi \sigma + (1 - \xi) H(w(0))} \right]^2 \cdot \left[\frac{z_j(k)}{z_1(k)} \right]^2 \\ &\leq \left[\frac{\xi \lambda_{m+1} + (1 - \xi) H(w(0))}{\xi \sigma + (1 - \xi) H(w(0))} \right]^2 \cdot \left[\frac{z_j(k)}{z_1(k)} \right]^2 \\ &= \left[\frac{z_j(0)}{z_1(0)} \right]^2 \cdot e^{-\theta_1(k+1)} \end{aligned} \quad (28)$$

for all $k \geq 1$.

Since $w^T(k) A w(k)$ is bounded, there exists a constant $d > 0$ such that $z_1^2(k) \leq d$ for all $k \geq 0$. Then

$$\sum_{j=m+1}^n z_j^2(k) = \sum_{j=m+1}^n \left[\frac{z_j(k)}{z_1(k)} \right]^2 \cdot z_1^2(k) \leq \Pi_1 e^{-\theta_1 k} \quad (29)$$

for $k \geq 1$, where $\Pi_1 = d \sum_{j=m+1}^n \left[\frac{z_j(0)}{z_1(0)} \right]^2 \geq 0$ □

Next, for convenience, denote

$$P(k) = \left[1 + \xi \left(\frac{\sigma}{w^T(k) A w(k)} - 1 \right) \right]^2 w^T(k) A w(k) \quad (30)$$

$$Q(k) = \sum_{i=m+1}^n (a \lambda_i + b) \left[2(1 - \xi) + \frac{\xi (\lambda_i + \sigma)}{w^T(k) A w(k)} \right] \left[\frac{\xi (\sigma - \lambda_i)}{w^T(k) A w(k)} \right] z_i^2(k) \quad (31)$$

for all $k \geq 0$. Clearly, $P(k) \geq 0$ and $Q(k) \geq 0$ for all $k \geq 0$.

Lemma 4. It holds that

$$w^T(k+1) A w(k+1) = P(k) - Q(k) \quad (32)$$

for $k \geq 0$.

Proof. From (9) and (10), it follows that

$$\begin{aligned}
 w^T(k+1)Aw(k+1) &= \sum_{i=1}^n \left[1 + \xi \left(\frac{\lambda_i}{w^T(k)Aw(k)} - 1 \right) \right]^2 (a\lambda_i + b) z_i^2(k) \\
 &= \sum_{i=1}^n \left[1 + \xi \left(\frac{\sigma}{w^T(k)Aw(k)} - 1 \right) \right]^2 (a\lambda_i + b) z_i^2(k) \\
 &\quad - \sum_{i=m+1}^n (a\lambda_i + b) \left[2(1-\xi) + \frac{\xi(\lambda_i + \sigma)}{w^T(k)Aw(k)} \right] \left[\frac{\xi(\sigma - \lambda_i)}{w^T(k)Aw(k)} \right] z_i^2(k) \quad (33) \\
 &= \left[1 + \xi \left(\frac{\sigma}{w^T(k)Aw(k)} - 1 \right) \right]^2 w^T(k)Aw(k) \\
 &\quad - \sum_{i=m+1}^n (a\lambda_i + b) \left[2(1-\xi) + \frac{\xi(\lambda_i + \sigma)}{w^T(k)Aw(k)} \right] \left[\frac{\xi(\sigma - \lambda_i)}{w^T(k)Aw(k)} \right] z_i^2(k) \\
 &= P(k) - Q(k)
 \end{aligned}$$

□

Lemma 5. If $w(0) \notin V_\sigma^\perp$, it holds that

$$P(k-1) \geq 4(1-\xi)\xi\sigma \quad (34)$$

Proof. From (30), it follows that

$$\begin{aligned}
 P(k-1) &= \left[1 + \xi \left(\frac{\sigma}{w^T(k-1)Aw(k-1)} - 1 \right) \right]^2 w^T(k-1)Aw(k-1) \\
 &\geq \min_{s>0} \left\{ \left[1 + \xi \left(\frac{\sigma}{s} - 1 \right) \right]^2 s \right\} \quad (35)
 \end{aligned}$$

for $k \geq 1$. By Lemma 1, it holds that

$$P(k-1) \geq 4(1-\xi)\xi\sigma, k \geq 1 \quad \square$$

Lemma 6. There exists a positive constant $\Pi_2 > 0$ such that

$$Q(k) \leq \Pi_2 \cdot e^{-\theta k} \quad (36)$$

for $k \geq 0$.

Proof. From (31)

$$\begin{aligned}
 Q(k) &= \sum_{i=m+1}^n (a\lambda_i + b) \left[2(1-\xi) + \frac{\xi(\lambda_i + \sigma)}{w^T(k)Aw(k)} \right] \left[\frac{\xi(\sigma - \lambda_i)}{w^T(k)Aw(k)} \right] z_i^2(k) \\
 &\leq \sum_{i=m+1}^n (a\lambda_i + b) \left[2 + \frac{2\xi\sigma}{w^T(k)Aw(k)} \right] \left[\frac{\xi\sigma}{w^T(k)Aw(k)} \right] z_i^2(k) \quad (37) \\
 &\leq 2 \left[1 + \frac{\xi\sigma}{w^T(k)Aw(k)} \right] \left[\frac{\xi\sigma}{w^T(k)Aw(k)} \right] \sum_{i=m+1}^n (a\lambda_i + b) z_i^2(k)
 \end{aligned}$$

for $k \geq 0$. By Theorem 1 and Lemma 3,

$$\begin{aligned}
 Q(k) &\leq 2 \left[1 + \frac{\xi\sigma}{4(1-\xi)\xi\lambda_p} \right] \left[\frac{\xi\sigma}{4(1-\xi)\xi\lambda_p} \right] \sum_{i=m+1}^n (a\lambda_i + b) z_i^2(k) \\
 &\leq 2\Pi_1 \left[1 + \frac{\sigma}{4(1-\xi)\lambda_p} \right] \left[\frac{\sigma}{4(1-\xi)\lambda_p} \right] e^{-\theta k} \leq \Pi_2 \cdot e^{-\theta k} \quad (38)
 \end{aligned}$$

for $k \geq 0$, where

$$\Pi_2 = 2\Pi_1 \left[1 + \frac{\sigma}{4(1-\xi)\lambda_p} \right] \left[\frac{\sigma}{4(1-\xi)\lambda_p} \right] \quad \square$$

Lemma 7. If $w(0) \notin V_\sigma^\perp$, then there exist constants $\theta_2 > 0$ and $\Pi_5 > 0$ such that

$$\left| \sigma - w^T(k+1)A w(k+1) \right| \leq k\Pi_5 \left[\left\{ e^{-\theta_2(k+1)} + \max \left\{ e^{-\theta_2 k}, e^{-\theta_1 k} \right\} \right\} \right] \quad (39)$$

for all $k \geq 1$, where

$$\begin{cases} \theta_2 = -\ln \delta \\ \delta = \max \left\{ (1-\xi)^2, \frac{\xi}{4(1-\xi)} \right\} \end{cases} \quad (40)$$

and $0 < \xi < 0.8, 0 < \delta < 1$.

Proof. From (30), (31) and Lemma 4, it follows that

$$\begin{aligned} \sigma - w^T(k+1)A w(k+1) &= \sigma - P(k) + Q(k) \\ &= \sigma - \left[1 + \xi \left(\frac{\sigma}{w^T(k)A w(k)} - 1 \right) \right]^2 w^T(k)A w(k) + Q(k) \\ &= (\sigma - w^T(k)A w(k)) \left[(1-\xi)^2 - \frac{\xi^2 \sigma}{w^T(k)A w(k)} \right] + Q(k) \\ &= (\sigma - w^T(k)A w(k)) \left[(1-\xi)^2 - \frac{\xi^2 \sigma}{P(k-1) - Q(k-1)} \right] + Q(k) \\ &= (\sigma - w^T(k)A w(k)) \left[(1-\xi)^2 - \frac{\xi^2 \sigma}{P(k-1)} \right] \\ &\quad - \frac{(\sigma - w^T(k)A w(k)) \xi^2 \sigma Q(k-1)}{P(k-1)(P(k-1) - Q(k-1))} + Q(k) \\ &= (\sigma - w^T(k)A w(k)) \left[(1-\xi)^2 - \frac{\xi^2 \sigma}{P(k-1)} \right] \\ &\quad - \frac{(\sigma - w^T(k)A w(k)) \xi^2 \sigma}{P(k-1)w^T(k)A w(k)} Q(k-1) + Q(k) \\ &= (\sigma - w^T(k)A w(k)) \left[(1-\xi)^2 - \frac{\xi^2 \sigma}{P(k-1)} \right] \\ &\quad - \xi^2 \sigma \left[\frac{\sigma}{w^T(k)A w(k)} - 1 \right] \frac{Q(k-1)}{P(k-1)} + Q(k) \end{aligned} \quad (41)$$

for $k \geq 1$.

Denote

$$V(k) = \left| \sigma - w^T(k)A w(k) \right| \quad (42)$$

for $k \geq 1$. It follows that,

$$\begin{aligned}
 V(k+1) &\leq V(k) \cdot \left| (1-\xi)^2 - \frac{\xi^2 \sigma}{P(k-1)} \right| \\
 &\quad + \xi^2 \sigma \left[\frac{\sigma}{w^T(k) A w(k)} + 1 \right] \frac{Q(k-1)}{P(k-1)} + Q(k) \\
 &\leq \max \left\{ (1-\xi)^2, \frac{\xi^2 \sigma}{P(k-1)} \right\} V(k) \\
 &\quad + \xi^2 \sigma \left[\frac{\sigma}{w^T(k) A w(k)} + 1 \right] \frac{Q(k-1)}{P(k-1)} + Q(k)
 \end{aligned} \tag{43}$$

for $k \geq 1$. By Lemma 5, it holds that

$$\begin{aligned}
 V(k+1) &\leq \max \left\{ (1-\xi)^2, \frac{\xi}{4(1-\xi)} \right\} V(k) \\
 &\quad + \frac{\xi}{4(1-\xi)} \left[\frac{\sigma}{w^T(k) A w(k)} + 1 \right] Q(k-1) + Q(k)
 \end{aligned} \tag{44}$$

for $k \geq 1$. Denote

$$\delta = \max \left\{ (1-\xi)^2, \frac{\xi}{4(1-\xi)} \right\} \tag{45}$$

Clearly, if $0 < \xi < 0.8, 0 < \delta < 1$. Then

$$V(k+1) \leq \delta \cdot V(k) + \frac{\xi}{4(1-\xi)} \left[\frac{\sigma}{w^T(k) A w(k)} + 1 \right] Q(k-1) + Q(k) \tag{46}$$

Denote

$$\Pi_3 = \frac{\xi}{4(1-\xi)} \left[\frac{\sigma}{4(1-\xi)\xi\lambda_p} + 1 \right] \tag{47}$$

By Theorem 1 and Lemma 6,

$$\begin{aligned}
 V(k+1) &\leq \delta \cdot V(k) + \Pi_3 \cdot Q(k-1) + Q(k) \\
 &\leq \delta \cdot V(k) + \Pi_4 \cdot e^{-\theta_1 k}
 \end{aligned} \tag{48}$$

where

$$\Pi_4 = \Pi_2 \cdot (\Pi_3 e^{\theta_1} + 1) \tag{49}$$

Denote.

$$\begin{cases} \theta_2 = -\ln \delta \\ \delta = \max \left\{ (1-\xi)^2, \frac{\xi}{4(1-\xi)} \right\} \end{cases} \tag{50}$$

Then,

$$\begin{aligned}
 V(k+1) &\leq \delta^{k+1} V(0) + \Pi_4 \sum_{r=0}^k \delta^r e^{-\theta_1(k-r)} \\
 &\leq \delta^{k+1} V(0) + k \Pi_4 \cdot \max \{ \delta^k, e^{-\theta_1 k} \} \\
 &\leq k \Pi_5 \left[e^{-\theta_2(k+1)} + \max \{ e^{-\theta_2 k}, e^{-\theta_1 k} \} \right]
 \end{aligned} \tag{51}$$

for all $k \geq 1$. □

Lemma 8. Suppose there exists constants $\theta > 0$ and $\Pi > 0$ such that

$$\frac{\xi}{w^T(k)Aw(k)} \left| (\sigma - w^T(k)Aw(k))z_i(k) \right| \leq k \cdot \Pi e^{-\theta k}, \quad (i = 1, \dots, m) \tag{52}$$

for $k \geq 1$. Then,

$$\lim_{k \rightarrow \infty} z_i(k) = z_i^*, \quad (i = 1, \dots, m) \tag{53}$$

where $z_i^* (i = 1, \dots, m)$ are constants.

Proof. When $z_i(k) \leq 1$

$$\begin{aligned} & \frac{\xi}{w^T(k)Aw(k)} \left| (\sigma - w^T(k)Aw(k))z_i(k) \right| \\ & \leq \frac{\xi}{w^T(k)Aw(k)} \left| (\sigma - w^T(k)Aw(k)) \right| \\ & \leq k \cdot \Pi e^{-\theta k} \end{aligned} \tag{54}$$

when $z_i(k) > 1$

$$\begin{aligned} & \frac{\xi}{w^T(k)Aw(k)} \left| (\sigma - w^T(k)Aw(k))z_i(k) \right| \\ & \leq \xi \cdot \frac{z_i(k)}{\sum_{i=1}^m (a\sigma + b)z_i(k)^2} \left| (\sigma - w^T(k)Aw(k)) \right| \\ & \leq \frac{\xi}{(a\sigma + b)} \left| (\sigma - w^T(k)Aw(k)) \right| \\ & \leq k \cdot \Pi e^{-\theta k} \end{aligned} \tag{55}$$

Next, we can prove that each sequence $\{z_i(k)\}$ is a Cauchy sequence. See [14] for the proof. By Cauchy Convergence Principle, there exist constants $z_i^* (i = 1, \dots, m)$ such that

$$\lim_{k \rightarrow \infty} z_i(k) = z_i^*, \quad (i = 1, \dots, m) \tag{56}$$

Theorem 3. Suppose that $0 < \xi < 0.8$, if $w(0) \notin V_\sigma^\perp$ then the trajectory of (4) starting from $w(0)$ will converge to a linear combination of σ and coefficients a, b . This linear combination is the eigenvector corresponding to the largest eigenvalue of its covariance matrix.

Proof. By Lemma 3, there exists constants $\theta_1 > 0$ and $\Pi_1 \geq 0$ such that

$$\sum_{j=m+1}^n z_j^2(k) \leq \Pi_1 \cdot e^{-\theta_1 k} \tag{57}$$

for all $k \geq 0$. By Lemma 7, there exists constants $\theta_2 > 0$ and $\Pi_2 > 0$ such that

$$\left| \sigma - w^T(k+1)Aw(k+1) \right| \leq k \cdot \Pi_2 \cdot \left[e^{-\theta_2(k+1)} + \max \{ e^{-\theta_2 k}, e^{-\theta_1 k} \} \right] \tag{57}$$

for all $k \geq 0$.

Obviously, there exists constants $\theta > 0$ and $\Pi > 0$ such that

$$\frac{\xi}{w^T(k)Aw(k)} \left| (\sigma - w^T(k)Aw(k))z_i(k) \right| \leq k \cdot \Pi e^{-\theta k} \tag{58}$$

for $k \geq 0$ and $i(i = 1, \dots, m)$

Using Lemmas 3 and 8, it follows that

$$\begin{cases} \lim_{k \rightarrow \infty} z_i(k) = z_i^*, (i = 1, \dots, m) \\ \lim_{k \rightarrow \infty} z_i(k) = 0, (i = m + 1, \dots, n) \end{cases} \tag{59}$$

Then,

$$\lim_{k \rightarrow \infty} w(k) = \sum_{i=1}^m z_i^* v_i \in V_\sigma \tag{60}$$

When the algorithm converges, we have

$$\lim_{k \rightarrow \infty} C w(k) = \lim_{k \rightarrow \infty} w^T(k) A w(k) w(k) \tag{61}$$

$$\sum_{i=1}^m \sigma z_i^* v_i = \sum_{i=1}^m (a\sigma + b)(z_i^*)^2 \cdot \sum_{i=1}^m z_i^* v_i \tag{62}$$

$$\sum_{i=1}^m (z_i^*)^2 = \frac{\sigma}{a\sigma + b} \tag{63}$$

Consider the most common case where the largest eigenvalue has only one. After the algorithm converges, $(z_1^*)^2 = \sigma / (a\sigma + b)$. We have $w = [\sigma / (a\sigma + b)]^{1/2} v$. Finally, we can get the largest eigenvector

$$v = w / [\sigma / (a\sigma + b)]^{1/2} = w / [w^T A w / (a w^T A w + b)]^{1/2} \quad \square$$

4. Numerical Experiment

Some examples will be provided in this section to illustrate the proposed theory.

4.1. Offline Data Experiment

The numerical experiment is mainly to observe the convergence of the algorithm, and we randomly generate a covariance matrix

$$C = \begin{bmatrix} 1.090719 & 0.154061 & 0.109432 & 0.089424 & 0.05406 & 0.125653 \\ 0.154061 & 1.261628 & 0.185839 & 0.151862 & 0.091805 & 0.213386 \\ 0.109432 & 0.185839 & 1.132004 & 0.10787 & 0.065211 & 0.151571 \\ 0.089424 & 0.151862 & 0.10787 & 1.088148 & 0.053288 & 0.12386 \\ 0.05406 & 0.091805 & 0.065211 & 0.053288 & 1.032214 & 0.074877 \\ 0.125653 & 0.213386 & 0.151571 & 0.12386 & 0.074877 & 1.174038 \end{bmatrix}$$

According to the existing related methods, we can directly find C 's maximum eigenvalue is 1.778753, and the maximum eigenvector is [0.341311, 0.579619, 0.411713 0.336439, 0.203388, 0.472741]. Next, we select three different initial vectors, take $\xi = 0.5, a = 0.5, b = 0.5$ and use $\varepsilon = 0.0001$ as the stopping criterion (That is, in the iterative process of the algorithm, when the absolute value of each element of the i -th eigenvector minus each element of the $i-1$ -th step is less than ε , the algorithm is considered to have converged and the iteration is stopped) to observe the convergence of the algorithm.

The first initial vector we randomly generated is [0.5488, 0.7152, 0.6028, 0.5449, 0.4237, 0.6459]. The trajectory of the eigenvector iteration and the

learning rate iteration trajectory is shown in **Figure 1**. The algorithm converged after 23 iterations, $w^T A w$ converged to the maximum eigenvalue 1.778753, and the maximum eigenvector from the final iteration was [0.341436, 0.579398, 0.411745, 0.336571, 0.203655, 0.472685], and the learning rate converged to 0.281096. It can be seen that the error from the true maximum eigenvector is less than 0.0001.

The second initial vector we randomly generated is [0.0055, 0.0072, 0.006, 0.0054, 0.0042, 0.0065]. The trajectory of the eigenvector iteration and the learning rate iteration trajectory are shown in **Figure 2**. The algorithm converges after 27 iterations, $w^T A w$ converges to the maximum eigenvalue 1.778753, and the maximum eigenvector from the final iteration is [0.341423, 0.579428, 0.411735, 0.336547, 0.203620, 0.472697], and the learning rate converges to 0.281096. It can be seen that the error with the true maximum eigenvector is also less than 0.0001.

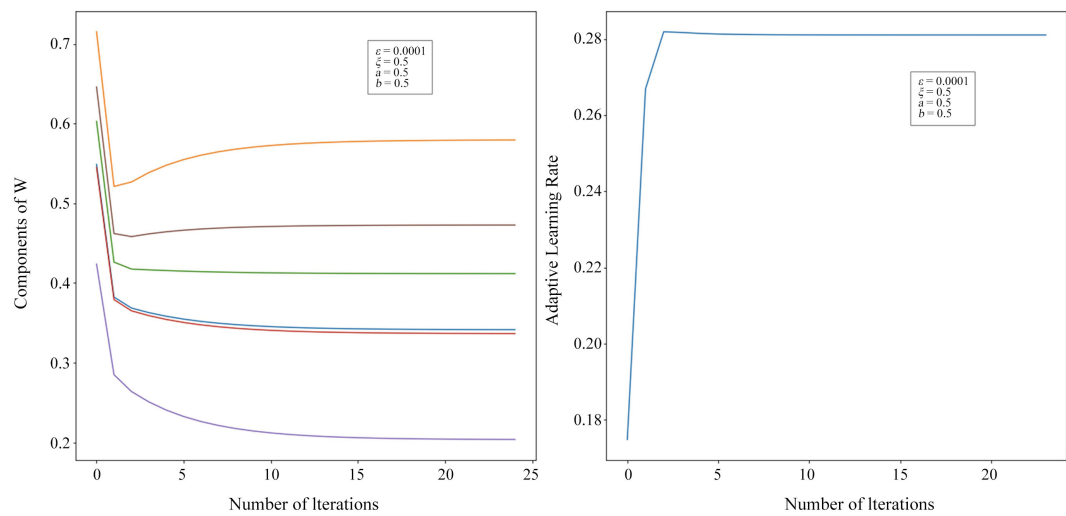


Figure 1. Convergence process of (6) with median initial vector.

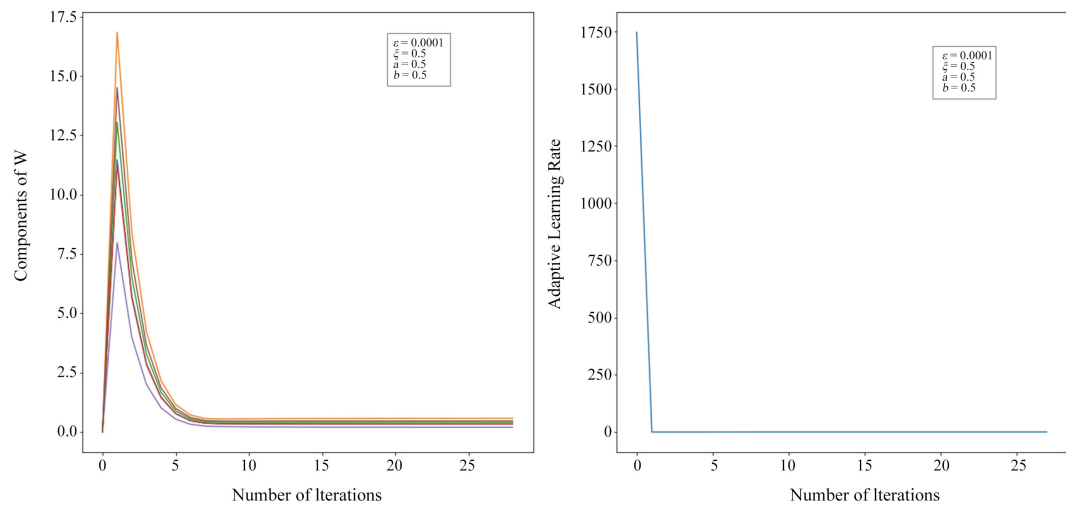


Figure 2. Convergence process of (6) with smaller initial vector.

The third initial vector we randomly generated is [1142.75, 1458.86, 1245.25, 1135.28, 904.94, 1327.2]. The trajectory of the eigenvector iteration and the learning rate iteration are shown in **Figure 3**. The algorithm converges after 35 iterations, $w^T A w$ converges to the maximum eigenvalue 1.778753, and the maximum eigenvector from the final iteration is [0.341414, 0.579436, 0.411738, 0.336548, 0.203616, 0.472693], and the learning rate converges to 0.281096. It can be seen that the error with the real maximum eigenvector is still less than 0.0001.

We can see that the algorithm can converge to the maximum eigenvector faster from three different initial value experiments. Of course, the closer the initial vector is to the maximum eigenvector, the faster the convergence will be. When the initial value is vast, the initial learning rate is close to 0, and the vector will gradually come down to convergence. When the initial value is minimal, since the initial learning rate will be huge, the vector will have a more significant jump and gradually converge. When ξ in the algorithm is determined, no matter how the initial vector is taken, the learning rate after the final algorithm converges is a fixed value, which is consistent with our theoretical proof.

Although in the theoretical proof, $w(0) \notin V_\sigma^\perp$ will converge, but after many numerical experiments, we found that an initial vector $w(0)$ is chosen arbitrarily, the results will converge. This is because the probability that $w(0) \in V_\sigma^\perp$ is almost 0. Then we select a low-dimensional covariance matrix with integer eigenvectors for experiment. We found that when $w(0) \in V_\sigma^\perp$, $w(k)$ is not divergent, it will converge to other eigenvectors. Suppose the multiplicity of the largest eigenvalue (denoted by σ) of C is $m(1 \leq m \leq n)$, i.e.,

$\lambda_1 = \lambda_2 = \dots = \lambda_m = \sigma$; Suppose the multiplicity of the second largest eigenvalue (denoted by τ) of C is $t(m < t \leq n)$, i.e., $\lambda_{m+1} = \lambda_{m+2} = \dots = \lambda_{m+t} = \tau$. Denote $V_\sigma^\perp = span\{v_{m+1}, \dots, v_n\}$, $V_\tau^\perp = span\{v_{m+\tau+1}, \dots, v_n\}$ when $w(0) \notin V_\sigma^\perp$, $w(k)$ will converge to the largest eigenvector of C ; when $w(0) \in V_\sigma^\perp$ and $w(0) \notin V_\tau^\perp$, $w(k)$ will converge to the second largest eigenvector of C .

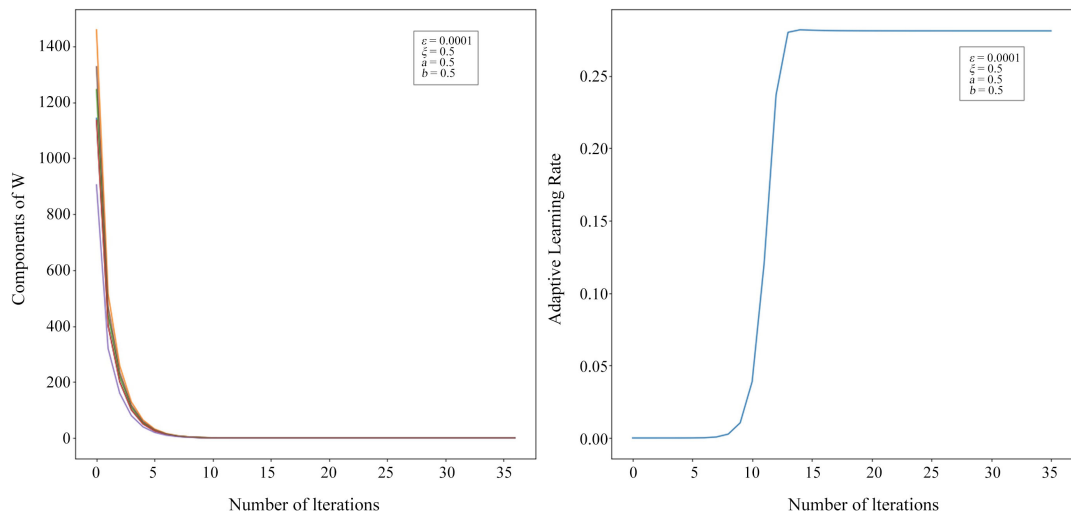


Figure 3. Convergence process of (6) with larger initial vector.

4.2. Comparison Algorithm (5) and (6)

In order to compare the difference between (5) and (6), we did some experiments. We randomly generated an initial vector as $[571.37, 729.43, 622.63, 567.64, 452.47, 663.6]$. The two algorithms use the same $\zeta = 0.2$. For (6) we use $a = 0.001, b = 0.001$. We still use $\varepsilon = 0.0001$ as the stopping criterion. The results are shown in **Figure 4** and **Figure 5**.

From the results, we can see that Algorithm (5) converges after 83 iterations, and Algorithm (6) converges after 69 iterations. Algorithm (6) can quickly converge to near the true value within the first ten iterations, while Algorithm (5) requires more iterations. We found that when we have a larger initial value, choosing smaller coefficients a, b can help the algorithm converge more quickly.

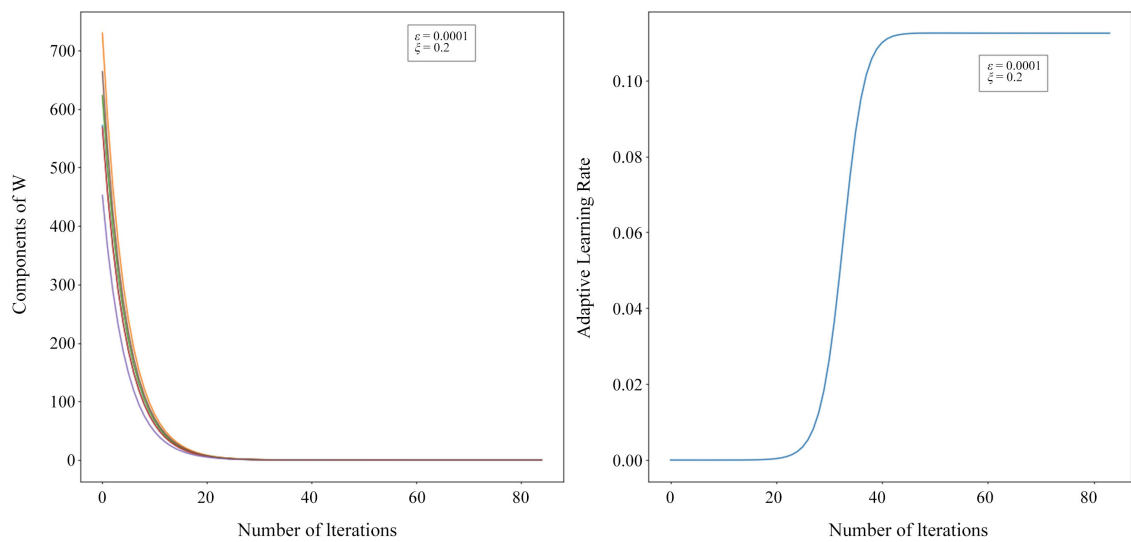


Figure 4. Convergence of algorithm (5) with larger initial vector.

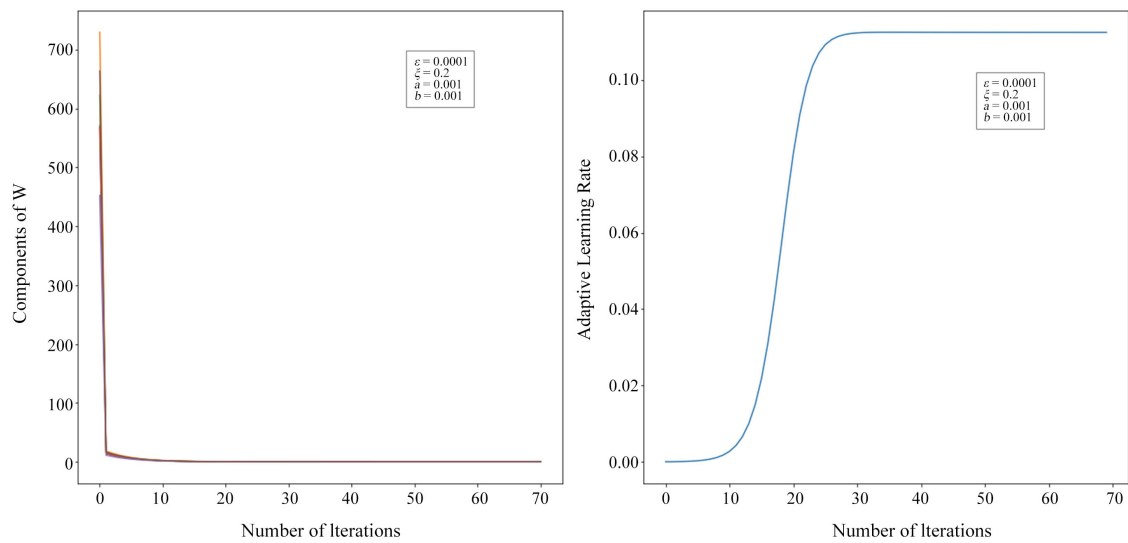


Figure 5. Convergence of algorithm (6) with larger initial vector and smaller a, b .

4.3. Selection of Parameters a, b

The most significant difference between Algorithm (6) and Algorithm (5) is the newly introduced two parameters a and b . Next, we used numerical experiments to discuss the influence of parameters a and b on the algorithm's convergence process.

We take $[0.5488, 0.7152, 0.6028, 0.5449, 0.4237, 0.6459]$ as the initial vector, choose $\xi = 0.2$, and change a, b to observe the convergence of the algorithm (6). The number of iterations required by the algorithm (6) under different parameters is shown in **Table 1**.

We can see that when one parameter is much larger than the other, the smaller parameter will lose its effect. We must keep the two parameters in the same order of magnitude. This is because the covariance matrix C and the identity matrix I are in the same order of magnitude in the experiment. If one coefficient is larger, the effect of the other coefficient will be reduced.

From **Table 1**, we can see that when $a, b \in [0.01, 0.1]$ the convergence speed is the fastest. Next, we take a value in this interval, and the convergence rate hardly changes. This shows that the algorithm is not sensitive to changes of a and b in the same order of magnitude.

We use $[0.0055, 0.0072, 0.006, 0.0054, 0.0042, 0.0065]$ as the initial vector, we found when $a, b \in [100, 1000]$ the convergence speed is the fastest. Then we use $[59.3932, 74.367, 64.2487, 59.0395, 48.1289, 68.1305]$ as the initial vector, we found when $a, b \in [10^{-6}, 10^{-5}]$ the convergence speed is the fastest. This shows that when we choose a larger initial vector, we should choose smaller a, b and vice versa. The above three situations are all satisfied $w(0)^T A w(0) \in (0.01, 1)$. It means when we choose $w(0)$ a and b satisfied $w(0)^T A w(0) \in (0.01, 1)$, the convergence speed of the algorithm (6) will reach the fastest.

4.4. Online Data Experiment

Next, we consider the case of online data. Suppose the input sequence $\{x(k) | x(k) \in R^n (k = 0, 1, 2, \dots)\}$ is a zero mean stationary stochastic process. Because the data come in one by one online, we cannot get the covariance matrix C in advance. According to [4] C can be expressed as

$$C_k = \beta C_{k-1} + \frac{(x(k)x(k)^T - \beta C_{k-1})}{k} \quad (64)$$

where β is a coefficient, when $x(k)$ comes from a stationary process, then $\beta = 1$, else $\beta \in (0, 1)$. According to The Law of Large Numbers,

$P(\|C - C_k\|_F \leq \varepsilon)$ converges with probability 1.

We randomly generate data and add a new data every iteration. We still use $\varepsilon = 0.0001$ as the stopping criterion, and find that different initial vectors and different ξ will affect the convergence speed of the algorithm. Experiments with different initial values are just like offline data.

We studied the influence of parameter ξ on the convergence of the algorithm and selected the same initial value. The result is shown in **Figure 6**. It is found that the smaller the ξ is, the faster the algorithm can converge.

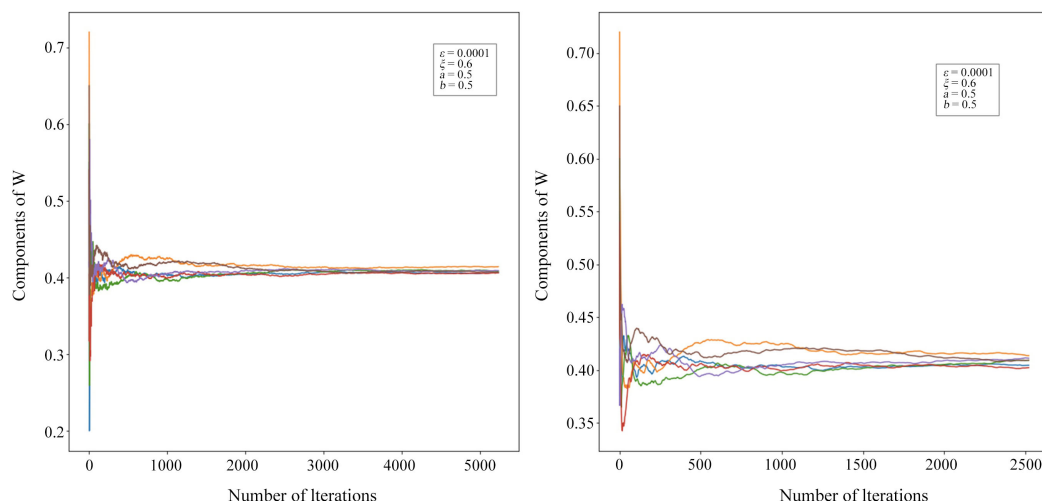


Figure 6. Convergence of algorithms with different ξ .

Table 1. Number of iterations with different a, b .

	$a = 0.0001$	$a = 0.001$	$a = 0.01$	$a = 0.1$	$a = 1$	$a = 10$	$a = 100$	$a = 1000$
$b = 0.0001$	55	51	47	47	51	56	61	67
$b = 0.001$	52	50	47	47	51	56	61	67
$b = 0.01$	48	47	46	47	51	56	61	67
$b = 0.1$	46	46	46	47	51	56	61	67
$b = 1$	50	50	50	50	52	56	61	67
$b = 10$	55	55	55	55	55	57	62	67
$b = 100$	60	60	60	60	60	60	62	67
$b = 1000$	65	65	65	65	65	65	66	68

5. Conclusion

In this paper, we proposed a Generalized Adaptive Learning Rate (GALR) PCA Algorithm, which could be guaranteed that the algorithm’s convergence process will not be affected by the selection of the initial value. Using the DDT method, we gave the upper and lower bounds of the algorithm and proved the global convergence. We no longer need the learning rate to tend to zero. Numerical experiments have also verified our theory. We discussed the relationship between the initial vector and the parameters a, b . Our algorithm is effective for both online and offline data.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Bouwmans, T., Javes, S., et al. (2018) On the Applications of Robust PCA in Image

- and Video Processing. *Proceedings of the IEEE*, **106**, 1427-1457.
<https://doi.org/10.1109/JPROC.2018.2853589>
- [2] Oja, E. (1982) Simplified Neuron Model as a Principal Component Analyzer. *Journal of Mathematical Biology*, **15**, 267-273. <https://doi.org/10.1007/BF00275687>
- [3] Xu, L. (1993) Least Mean Square Error Reconstruction Principle for Self-Organizing Neural-Nets. *Neural Networks*, **6**, 627-648.
[https://doi.org/10.1016/S0893-6080\(05\)80107-8](https://doi.org/10.1016/S0893-6080(05)80107-8)
- [4] Chatterjee, C., Kang, Z. and Roychowdhury, V.P. (2000) Algorithms for Accelerated Convergence of Adaptive PCA. *IEEE Transactions on Neural Networks*, **11**, 338-355. <https://doi.org/10.1109/72.839005>
- [5] Xu, L. and Yuille, A.L. (1995) Robust Principal Component Analysis by Self-Organizing Rules Based on Statistical Physics Approach. *IEEE Transactions on Neural Networks*, **6**, 131-143. <https://doi.org/10.1109/72.363442>
- [6] Wang, S., Liang, Y.L. and Ma, F. (1998) An Adaptive Robust PCA Neural Network. *The 1998 IEEE International Joint Conference on Neural Networks Proceedings*, Vol. 3, 2288-2293. <https://doi.org/10.1109/IJCNN.1998.687218>
- [7] Yang, T.-N. and Wang, S.D. (1999) Robust Algorithms for Principal Component Analysis. *Pattern Recognition Letters*, **20**, 927-933.
[https://doi.org/10.1016/S0167-8655\(99\)00060-4](https://doi.org/10.1016/S0167-8655(99)00060-4)
- [8] Chen, T.P., Hua, Y.B. and Yan, W.-Y. (1998) Global Convergence of Oja's Subspace Algorithm for Principal Component Extraction. *IEEE Transactions on Neural Networks*, **9**, 58-67. <https://doi.org/10.1109/72.655030>
- [9] Zhang, Q. and Bao, Z. (1995) Dynamical Systems for Computing the Eigenvectors Associated with the Largest Eigenvalue of a Positive Definite Matrix. *IEEE Transactions on Neural Networks*, **6**, 790-791. <https://doi.org/10.1109/72.377989>
- [10] Zhang, Q. and Leung, Y.-W. (2000) A Class of Learning Algorithms for Principal Component Analysis and Minor Component Analysis. *IEEE Transactions on Neural Networks*, **11**, 529-533. <https://doi.org/10.1109/72.839022>
- [11] Zhang, Q.F. (2003) On the Discrete-Time Dynamics of a PCA Learning Algorithm. *Neurocomputing*, **55**, 761-769. [https://doi.org/10.1016/S0925-2312\(03\)00439-9](https://doi.org/10.1016/S0925-2312(03)00439-9)
- [12] Zufiria, P.J. (2002) On the Discrete-Time Dynamics of the Basic Hebbian Neural-Network Nods. *IEEE Transactions on Neural Networks*, **13**, 1342-1352.
<https://doi.org/10.1109/TNN.2002.805752>
- [13] Yi, Z., Ye, M., Lv, J.C. and Tan, K.K. (2005) Convergence Analysis of a Deterministic Discrete Time System of Oja's PCA Learning Algorithm. *IEEE Transactions on Neural Networks*, **16**, 1318-1328. <https://doi.org/10.1109/TNN.2005.852236>
- [14] Jian, C.L., Zhang, Y. and Tan, K.K. (2006) Global Convergence of Oja's PCA Learning Algorithm with a Non-Zero-Approaching Adaptive Learning Rate. *Theoretical Computer Science*, **367**, 286-307. <https://doi.org/10.1016/j.tcs.2006.07.012>
- [15] Kong, X.Y., Hu, C.H. and Duan, Z.S. (2017) *Principal Component Analysis Networks and Algorithms*. Springer Nature, Berlin.
<https://doi.org/10.1007/978-981-10-2915-8>