

Chapter 1

Introduction

1.1 Feature Extraction

Pattern recognition and data compression are two applications that rely critically on efficient data representation [1]. The task of pattern recognition is to decide to which class of objects an observed pattern belonging to, and the compression of data is motivated by the need to save the number of bits to represent the data while incurring the smallest possible distortion [1]. In these applications, it is desirable to extract measurements that are invariant or insensitive to the variations within each class. The process of extracting such measurements is called *feature extraction*. It is also to say feature extraction is a data processing which maps a high-dimensional space to a low-dimensional space with minimum information loss.

Principal component analysis (PCA) is a well-known feature extraction method, while minor component analysis (MCA) and independent component analysis (ICA) can be regarded as variants or generalizations of the PCA. MCA is most useful for solving total least squares (TLS) problems, and ICA is usually used for blind signal separation (BSS).

In the following, we briefly review PCA, PCA neural networks, and extensions or generalizations of PCA.

1.1.1 PCA and Subspace Tracking

The principal components (PC) are the directions in which the data have the largest variances and capture most of the information contents of data. They correspond to the eigenvectors associated with the largest eigenvalues of the autocorrelation matrix of the data vectors. Expressing data vectors in terms of the PC is called PCA. On the contrary, the eigenvectors that correspond to the smallest eigenvalues of the autocorrelation matrix of the data vectors are defined as the minor components

(MC), and MC are the directions in which the data have the smallest variances (they represent the noise in the data). Expressing data vectors in terms of the MC is called MCA. Now, PCA has been successfully applied in many data processing problems, such as high-resolution spectral estimation, system identification, image compression, and pattern recognition, and MCA is also applied in total least squares, moving target indication, clutter cancelation, curve and surface fitting, digital beamforming, and frequency estimation.

The PCA or MCA is usually one dimensional. However, in real applications, PCA or MCA is mainly multiple dimensional. The eigenvectors associated with the r largest (or smallest) eigenvalues of the autocorrelation matrix of the data vectors is called principal (or minor) components, and r is referred to as the number of the principal (or minor) components. The eigenvector associated with the largest (smallest) eigenvalue of the autocorrelation matrix of the data vectors is called largest (or smallest) component. The subspace spanned by the principal components is called principal subspace (PS), and the subspace spanned by the minor components is called minor subspace (MS). In some applications, we are only required to find the PS (or MS) spanned by r orthonormal eigenvectors. The PS is sometimes called signal subspace, and the MS is called noise subspace. Principal and minor component analyzers of a symmetric matrix are matrix differential equations that converge on the PCs and MCs, respectively. Similarly, the principal (PSA) and minor (MSA) subspace analyzers of a symmetric matrix are matrix differential equations that converge on a matrix whose columns' span is the PS and MS, respectively. PCA/PSA and MCA/MSA are powerful techniques in many information processing fields. For example, PCA/PSA is a useful tool in feature extraction, data compression, pattern recognition, and time series prediction [2, 3], and MCA/MSA has been widely applied in total least squares, moving target indication, clutter cancelation, curve and surface fitting, digital beamforming, and frequency estimation [4].

As discussed before, the PC is the direction which corresponds to the eigenvector associated with the largest eigenvalue of the autocorrelation matrix of the data vectors, and the MC is the direction which corresponds to the eigenvector associated with the smallest eigenvalue of the autocorrelation matrix of the data vectors. Thus, implementations of these techniques can be based on batch eigenvalue decomposition (ED) of the sample correlation matrix or on singular value decomposition (SVD) of the data matrix. This approach is unsuitable for adaptive processing because it requires repeated ED/SVD, which is a very time-consuming task [5]. Thus, the attempts to propose adaptive algorithms are still continuing even though the field has been active for three decades up to now.

1.1.2 PCA Neural Networks

In order to overcome the difficulty faced by ED or SVD, a number of adaptive algorithms for subspace tracking were developed in the past. Most of these

techniques can be grouped into three classes [5]. In the first class, classical batch ED/SVD methods such as QR algorithm, Jacobi rotation, power iteration, and Lanczos method have been modified for the use in adaptive processing [6–10]. In the second class, variations in Bunch’s rank-one updating algorithm [11], such as subspace averaging [12, 13], have been proposed. The third class of algorithms considers the ED/SVD as a constrained or unconstrained optimization problem. Gradient-based methods [14–19], Gauss–Newton iterations [20, 21], and conjugate gradient techniques [22] can then be applied to seek the largest or smallest eigenvalues and their corresponding eigenvectors adaptively. Rank revealing URV decomposition [23] and rank revealing QR factorization [24] have been proposed to track the signal or noise subspace.

Neural network approaches on PCA or MCA pursue an effective “online” approach to update the eigen direction after each presentation of a data point, which possess many obvious advantages, such as lower computational complexity, compared with the traditional algebraic approaches such as SVD. Neural network methods are especially suited for high-dimensional data, since the computation of the large covariance matrix can be avoided, and for the tracking of nonstationary data, where the covariance matrix changes slowly over time. The attempts to improve the methods and to suggest new approaches are continuing even though the field has been active for two decades up to now.

In the last decades, many neural network learning algorithms were proposed to extract PS [25–31] or MS [4, 32–40]. In the class of PS tracking, lots of learning algorithms such as Oja’s subspace algorithm [41], the symmetric error correction algorithm [42], and the symmetric version of the back propagation algorithm [43] were proposed based on some heuristic reasoning [44]. Afterward, some information criterions were proposed and the corresponding algorithms such as LMSER algorithm [31], the projection approximation subspace tracking (PAST) algorithm [5], the conjugate gradient method [45], the Gauss–Newton method [46], and the novel information criterion (NIC) algorithm were developed [44]. These gradient-type algorithms could be claimed to be globally convergent.

In the class of MS tracking, many algorithms [32–40] have been proposed on the basis of the feedforward neural network models. Mathew and Reddy proposed the MS algorithm based on a feedback neural network structure with sigmoid activation function [46]. Using the inflation method, Luo and Unbehauen proposed an MSA algorithm that does not need any normalization operation [36]. Douglas et al. presented a self-stabilizing minor subspace rule that does not need periodically normalization and matrix inverses [40]. Chiang and Chen showed that a learning algorithm can extract multiple MCs in parallel with the appropriate initialization instead of inflation method [47]. On the basis of an information criterion, Ouyang et al. developed an adaptive MC tracker that automatically finds the MS without using the inflation method [37]. Recently, Feng et al. proposed the OJAm algorithm and extended it for tracking multiple MCs or the MS, which makes the corresponding state matrix tend to a column orthonormal basis of the MS [35].

1.1.3 *Extension or Generalization of PCA*

It can be found that the above-mentioned algorithms only focused on eigenvector extraction or eigen-subspace tracking with noncoupled rules. However, a serious speed stability problem exists in the most noncoupled rules [28]. This problem is that in noncoupled PCA rules the eigen motion in all directions mainly depends on the principal eigenvalue of the covariance matrix; thus, numerical stability and fast convergence can only be achieved by guessing this eigenvalue in advance [28]; in noncoupled MCA rules the speed of convergence does not only depend on the minor eigenvalue, but also depend on all other eigenvalues of the covariance matrix, and if these extend over a large interval, no suitable learning rate may be found for a numerical solution that can still guarantee stability and ensure a sufficient speed of convergence in all eigen directions. Therefore, the problem is even more severe for MCA rules. To solve this common problem, Moller proposed some coupled PCA algorithms and some coupled MCA algorithms based on a special information criteria [28]. In coupled rules, the eigen pair (eigenvector and eigenvalue) is simultaneously estimated in coupled equations, and the speed of convergence only depends on the eigenvalue of its Jacobian. Thus, the dependence of the eigenvalues on the covariance matrix can be eliminated [28]. Recently, some modified coupled rules have been proposed [48].

It is well known that the generalized eigen decomposition (GED) plays very important roles in various signal processing applications, e.g., data compression, feature extraction, denoising, antenna array processing, and classification. Though PCA, which is the special case of GED problem, has been widely studied, the adaptive algorithms for the GED problem are scarce. Fortunately, a few efficient online adaptive algorithms for the GED problem that can be applied in real-time applications have been proposed [49–54]. In [49], Chatterjee et al. present new adaptive algorithms to extract the generalized eigenvectors from two sequences of random vectors or matrices. Most algorithms in literatures including [49] are gradient-based algorithms [50, 51]. The main problem of this type of algorithms is slow convergence and the difficulty in selecting an appropriate step size which is essential: A too small value will lead to slow convergence and a too large value will lead to overshooting and instability. Rao et al. [51] have developed a fast recursive least squares (RLS)-like, not true RLS, sequential algorithm for GED. In [54], by reinterpreting the GED problem as an unconstrained minimization problem via constructing a novel cost function and applying projection approximation method and RLS technology to the cost function, RLS-based parallel adaptive algorithms for generalized eigen decomposition was proposed. In [55], a power method-based algorithm for tracking generalized eigenvectors was developed when stochastic signals having unknown correlation matrices are observed. Attallah proposed a new adaptive algorithm for the generalized symmetric eigenvalue problem, which can extract the principal and minor generalized eigenvectors, as well as their corresponding subspaces, at a low computational cost [56]. Recently, a fast and

numerically stable adaptive algorithm for the generalized Hermitian eigenvalue problem (GHEP) was proposed and analyzed in [48].

Other extensions of PCA also include dual-purpose algorithm [57–64], the details of which can be found in Chap. 5, and adaptive or neural networks-based SVD singular vector tracking [6, 65–70], the details of which can be found in Chap. 9.

1.2 Basis for Subspace Tracking

In Sect. 1.1, we have reviewed the PCA algorithm and its extensions and generalizations from the viewpoint of the feature extraction. In this section, from another viewpoint of subspace, we will discuss the concept of subspace and subspace tracking method.

1.2.1 Concept of Subspace

Definition 1 If $S = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m\}$ is the vector subset of vector space V , then the set W of all linear combinations of $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m$ is called the *subspace* spanned by $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m$, namely

$$W = \text{Span}\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m\} = \{\mathbf{u} : \mathbf{u} = \alpha_1 \mathbf{u}_1 + \alpha_2 \mathbf{u}_2 + \dots + \alpha_m \mathbf{u}_m\}, \quad (1.1)$$

where each vector in W is called the generator of W , and the set $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m\}$ which is composed of all the generators is called the spanning set of the subspace. A vector subspace which only comprises zero vector is called a trivial subspace. If the vector set $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m\}$ is linearly irrelative, then it is called a group basis of W .

Definition 2 The number of vectors in any group basis of subspace W is called the *dimension* of W , which is denoted by $\dim(W)$. If any group basis of W is not composed of finite linearly irrelative vectors, then W is called an infinite-dimensional vector subspace.

Definition 3 Assume that $A = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n] \in \mathbf{C}^{m \times n}$ is a complex matrix and all the linear combinations of its column vectors constitute a subspace, which is called *column space* of matrix A and is denoted by $\text{Col}(A)$, namely

$$\text{Col}(A) = \text{Span}\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n\} = \left\{ \mathbf{y} \in \mathbf{C}^m : \mathbf{y} = \sum_{j=1}^n \alpha_j \mathbf{a}_j : \alpha_j \in \mathbf{C} \right\}. \quad (1.2)$$

Row space of matrix A can be defined similarly.

As stated in the above, the column space and row space of matrix $\mathbf{A}_{m \times n}$ are spanned by n column vectors and m row vectors, respectively. If $\text{rank}(\mathbf{A})$ is equal to r , then only r column or row vectors of matrix, which are linearly irrespctive, can constitute column space $\text{Span}(\mathbf{A})$ and row space $\text{Span}(\mathbf{A}^H)$, respectively. Obviously, it is an economical and better subspace expression method to use basis vector. The methods of constituting a subspace have primary transforms, and one can also use singular value decomposition to set up a normal orthogonal basis of base space.

Suppose that the data matrix \mathbf{A} has measure error or noises, and define measure data matrix as

$$\mathbf{X} = \mathbf{A} + \mathbf{W} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n] \in \mathbf{C}^{m \times n}, \quad (1.3)$$

where $\mathbf{x}_i \in \mathbf{C}^{m \times 1}$. In the fields of signal processing and system science, the column space of measure data matrix $\text{Span}(\mathbf{X}) = \text{Span}\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ is called *measure data space*.

Define the correlation matrix as:

$$\mathbf{R}_X = E\{\mathbf{X}^H \mathbf{X}\} = E\{(\mathbf{A} + \mathbf{W})^H (\mathbf{A} + \mathbf{W})\}. \quad (1.4)$$

Suppose that error matrix $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n]$ is statistically irrespctive of real data matrix \mathbf{A} , then

$$\mathbf{R}_X = E\{\mathbf{X}^H \mathbf{X}\} = E\{\mathbf{A}^H \mathbf{A}\} + E\{\mathbf{W}^H \mathbf{W}\}. \quad (1.5)$$

Define $\mathbf{R} = E\{\mathbf{A}^H \mathbf{A}\}$ and $E\{\mathbf{W}^H \mathbf{W}\} = \sigma_w^2 \mathbf{I}$, namely every measure noise is statistically irrespctive and they have the same variance σ_w^2 , it holds that

$$\mathbf{R}_X = \mathbf{R} + \sigma_w^2 \mathbf{I}. \quad (1.6)$$

Define $\text{rank}(\mathbf{A}) = r$, and the eigenvalue decomposition of matrix $\mathbf{R}_X = E\{\mathbf{X}^H \mathbf{X}\}$ can be written as $\mathbf{R}_X = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^H + \sigma_w^2 \mathbf{I} = \mathbf{U} (\mathbf{\Lambda} + \sigma_w^2 \mathbf{I}) \mathbf{U}^H = \mathbf{U} \mathbf{\Pi} \mathbf{U}^H$, where $\mathbf{\Pi} = \mathbf{\Sigma} + \sigma_w^2 \mathbf{I} = \text{diag}(\sigma_1^2 + \sigma_w^2, \dots, \sigma_r^2 + \sigma_w^2, \sigma_w^2, \dots, \sigma_w^2)$, $\mathbf{\Sigma} = \text{diag}(\sigma_1^2, \dots, \sigma_r^2, 0, \dots, 0)$, and $\sigma_1^2 \geq \sigma_2^2 \geq \dots \geq \sigma_r^2$ are the nonzero eigenvalues of the real autocorrelation matrix $\mathbf{R} = E\{\mathbf{A}^H \mathbf{A}\}$.

Obviously, if the signal-to-noise ratio is large enough, that is, σ_r^2 is obviously bigger than σ_w^2 , then the first r largest eigenvalues of autocorrelation matrix \mathbf{R}_X , namely $\lambda_1 = \sigma_1^2 + \sigma_w^2, \lambda_2 = \sigma_2^2 + \sigma_w^2, \dots, \lambda_r = \sigma_r^2 + \sigma_w^2$ are called the principal eigenvalues, and the remaining $n - r$ small eigenvalues $\lambda_{r+1} = \sigma_w^2, \lambda_{r+2} = \sigma_w^2, \dots, \lambda_n = \sigma_w^2$ are called the minor eigenvalues. Thus, the eigen decomposition of autocorrelation matrix \mathbf{R}_X can be written as

$$\mathbf{R}_X = [\mathbf{U}_S \quad \mathbf{U}_n] \begin{bmatrix} \boldsymbol{\Sigma}_S & \mathbf{O} \\ \mathbf{O} & \boldsymbol{\Sigma}_n \end{bmatrix} \begin{bmatrix} \mathbf{U}_S^H \\ \mathbf{U}_n^H \end{bmatrix} = \mathbf{S}\boldsymbol{\Sigma}_S\mathbf{S}^H + \mathbf{G}\boldsymbol{\Sigma}_n\mathbf{G}^H, \quad (1.7)$$

where $\mathbf{S} \stackrel{\text{def}}{=} [\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_r] = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_r]$, $\mathbf{G} \stackrel{\text{def}}{=} [\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_{n-r}] = [\mathbf{u}_{r+1}, \mathbf{u}_{r+2}, \dots, \mathbf{u}_n]$, $\boldsymbol{\Sigma}_S = \text{diag}(\sigma_1^2 + \sigma_w^2, \sigma_2^2 + \sigma_w^2, \dots, \sigma_r^2 + \sigma_w^2)$, $\boldsymbol{\Sigma}_n = \text{diag}(\sigma_w^2, \sigma_w^2, \dots, \sigma_w^2)$; $m \times r$ unitary matrix \mathbf{S} is the matrix composed of the eigenvectors which correspond to the r principal eigenvalues, and $m \times (n - r)$ unitary matrix \mathbf{G} is the matrix composed of the eigenvectors which correspond to the $n - r$ minor eigenvalues.

Definition 4 Define \mathbf{S} as the eigenvector matrix which correspond to the first r largest eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_r$ of the autocorrelation matrix of the measurement data. Then its column space $\text{Span}(\mathbf{S}) = \text{Span}\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_r\}$ is called the *signal subspace* of measurement data space $\text{Span}(X)$, and the column space $\text{Span}(\mathbf{G}) = \text{Span}\{\mathbf{u}_{r+1}, \mathbf{u}_{r+2}, \dots, \mathbf{u}_n\}$ of the eigenvector matrix \mathbf{G} which correspond to the $n - r$ minor eigenvalues is called the *noise subspace* of measurement data space.

In the following, we analyze the geometric meaning of the signal subspace and the noise subspace. From the constitution method of subspace and the feature of unitary matrix, we know that the signal subspace and noised subspace are orthogonal, that is,

$$\text{Span}\{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_r\} \perp \text{Span}\{\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_{n-r}\}. \quad (1.8)$$

Since \mathbf{U} is a unitary matrix, it holds that

$$\mathbf{U}\mathbf{U}^H = [\mathbf{S} \quad \mathbf{G}] \begin{bmatrix} \mathbf{S}^H \\ \mathbf{G}^H \end{bmatrix} = \mathbf{S}\mathbf{S}^H + \mathbf{G}\mathbf{G}^H = \mathbf{I},$$

that is,

$$\mathbf{G}\mathbf{G}^H = \mathbf{I} - \mathbf{S}\mathbf{S}^H. \quad (1.9)$$

Define the projection matrix of signal subspace as

$$\mathbf{P}_S \stackrel{\text{def}}{=} \mathbf{S}\langle \mathbf{S}, \mathbf{S} \rangle^{-1} \mathbf{S}^H = \mathbf{S}\mathbf{S}^H, \quad (1.10)$$

where the matrix inner product $\langle \mathbf{S}, \mathbf{S} \rangle = \mathbf{S}^H \mathbf{S} = \mathbf{I}$.

Thus, $\mathbf{P}_S \mathbf{x}$ can be considered as the projection of vector \mathbf{x} on the signal subspace, and $(\mathbf{I} - \mathbf{P}_S) \mathbf{x}$ means the orthogonal projection of vector \mathbf{x} on the signal subspace. From $\langle \mathbf{G}, \mathbf{G} \rangle = \mathbf{G}^H \mathbf{G} = \mathbf{I}$, it holds that the projection matrix on the noise subspace is $\mathbf{P}_n = \mathbf{G}\langle \mathbf{G}, \mathbf{G} \rangle^{-1} \mathbf{G}^H = \mathbf{G}\mathbf{G}^H$. Therefore, the following matrix

$$\mathbf{G}\mathbf{G}^H = \mathbf{I} - \mathbf{S}\mathbf{S}^H = \mathbf{I} - \mathbf{P}_S \quad (1.11)$$

is usually called as the orthogonal projection matrix of signal subspace.

The subspace applications have the following characteristics [5, 71]:

- (1) Only a few singular vectors or eigenvectors are needed. Since the number of larger singular values (or eigenvalues) of matrix $\mathbf{A}_{m \times n}$ is smaller than the number of smaller singular values (or eigenvalues), it is more efficient to use the signal subspace with smaller dimension than the noise subspace.
- (2) In many application occasions, one does not need to know the singular values or eigenvalues, and only needs to know the matrix rank and singular vectors or eigenvectors of matrix.
- (3) In most instances, one does not need to know the singular vectors or eigenvectors of matrix well and truly, and only needs to know the basis vectors spanned by the signal subspace or noise subspace.

1.2.2 Subspace Tracking Method

The iterative computation of an extreme (maximal or minimum) eigen pair (eigenvalue and eigenvector) can date back to 1966 [72]. In 1980, Thompson proposed a LMS-type adaptive algorithm for estimating eigenvector, which correspond to the smallest eigenvalue of sample covariance matrix, and provided the adaptive tracking algorithm of the angle/frequency combing with Pisarenko's harmonic estimator [14]. Sarkar et al. [73] used the conjugate gradient algorithm to track the variation of the extreme eigenvector which corresponds to the smallest eigenvalue of the covariance matrix of the slowly changing signal and proved its much faster convergence than Thompson's LMS-type algorithm. These methods were only used to track single extreme value and eigenvector with limited application, but later they were extended for the eigen-subspace tracking and updating methods. In 1990, Comon and Golub [6] proposed the Lanczos method for tracking the extreme singular value and singular vector, which is a common method designed originally for determining some big and sparse symmetrical eigen problem $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ [74].

The earliest eigenvalue and eigenvector updating method was proposed by Golub in 1973 [75]. Later, Golub's updating idea was extended by Bunch et al. [76, 77], the basic idea of which is to update the eigenvalue decomposition of the covariance matrix after every rank-one modification, and then go to the matrix's latent root using the interlacing theorem, and then update the place of the latent root using the iterative resolving root method. Thus, the eigenvector can be updated. Later, Schreiber [78] introduced a transform to change a majority of complex number arithmetic operation into real-number operation and made use of Karasalo's subspace mean method [79] to further reduce the operation quantity. DeGroat and

Roberts [80] developed a numerically stabilized rank-one eigen structure updating method based on mutual Gram–Schmidt orthogonalization. Yu [81] extended the rank-one eigen structure update to block update and proposed recursive update of the eigenvalue decomposition of a covariance matrix.

The earliest adaptive signal subspace tracking method was proposed by Owsley [7] in 1978. Using the stochastic gradient method, Yang and Kaveh [18] proposed a LMS-type subspace tracking algorithm and extended Owsley’s method and Thompson’s method. This LMS-type algorithm has a high parallel structure and low computational complexity. Karhunen [17] extended Owsley’s idea by developing a stochastic approaching method based on computing subspace. Like Yang and Kaveh’s extension of Thompson’s idea to develop an LMS-type subspace tracking algorithm, Fu and Dowling [45] extended Sarkar’s idea to develop a subspace tracking algorithm based on conjugate gradient. During the recent 20 years, eigen-subspace tracking and update has been an active research field. Since eigen-subspace tracking is mainly applied to real signal processing, these methods should be fast algorithms.

According to [71], the eigen-subspace tracking and updating methods can be classified into the following four classes:

- (1) In some applications of eigen-subspace method such as MUSIC, one only needs to use the orthogonal basis of the noise subspace eigenvectors and does not need to use the eigenvector itself. This characteristic can predigest the adaptive tracking problem of a class of eigenvectors. The methods which only track the orthogonal basis of noise subspace are classified as the first class, and they are based on rank revealing URV [82] and rank revealing QR [83] decomposition of matrix, respectively.
- (2) In the method conducting tracking and updating problem of the eigenvalues and eigen-subspace simultaneously, a common sight is to regard the covariance matrix of the nonstationary signal at the k th as the sum of the covariance matrix at the $k - 1$ th and another rank-one matrix (the product of the conjugate transpose of measure vector and itself). Thus, tracking the eigenvalue decomposition of the covariance matrix has much to do with the so-called rank-one updating [81, 84].
- (3) Regarding the determination of eigen-subspace as an optimization problem: The one is a constrained optimization problem, and the other is unconstrained optimization problem. The constrained optimization problem can be solved using the stochastic gradient [18] and conjugate gradient [45] methods. The unconstrained optimization problem presents a new explanation for the eigen-subspace, and its corresponding method was called projection approximation subspace tracking [5]. The other classical representative is that it is based on Lanczos algorithm, and to use the Lanczos iteration and stochastic approach concept to conduct on the computation of subspace of slowly changing data matrix [85]. Xu et al., proposed [86, 87] three Lanczos and dual Lanczos subspace tracking algorithms, and the former is suitable for the eigen decomposition of covariance matrix, and the latter is for the singular value

decomposition of data matrix, and at the processing of Lanczos iteration they can test and estimate the number of principal eigenvalues and principal singular values. In view of the close mathematics connections between the Lanczos algorithm and conjugate gradient, this algorithm, though it has not direct connections with the optimization problem, still falls into the third type of method.

- (4) Modify and extend the classical eigen decomposition/singular value decomposition batch processing methods such as QR decomposition, Jacobi method, and power iteration to make them adaptive. For example, the singular value decomposition updating algorithm based on QR updating and Jacobi-type method [88] falls into this class.

1.3 Main Features of This Book

This book presents principal component analysis algorithms and its extensions using neural networks approach. Pertinent features include the following:

- (1) A tutorial-style overview of neural networks-based principal component analysis algorithms, minor component analysis algorithms, principal subspace tracking, and minor subspace tracking.
- (2) Analysis of self-stabilizing feature of neural-based PCA/MCA algorithms, and development of a self-stabilizing neural-based minor component analysis algorithm.
- (3) Total least squares estimation application of MCA algorithms, and development of a novel neural-based algorithm for total least squares filtering.
- (4) Development of a novel dual-purpose principal and minor subspace gradient flow and unified self-stabilizing algorithm for principal and minor components' extraction.
- (5) Analysis of a discrete-time dynamics of a class of self-stabilizing MCA learning algorithms and a convergence analysis of deterministic discrete-time system of a unified self-stabilizing algorithm for PCA and MCA.
- (6) Extension of PCA algorithm to generalized feature extraction and development of a novel adaptive algorithm for minor generalized eigenvector extraction and a novel multiple generalized minor component extraction algorithm.
- (7) Development of a unified and coupled PCA and MCA rules and an adaptive coupled generalized eigen pairs extraction algorithm, based on Moller's coupled PCA neural algorithm.
- (8) Generalization of feature extraction from autocorrelation matrix to cross-correlation matrix, and development of an effective neural algorithm for extracting cross-correlation feature between two high-dimensional data streams and a coupled principal singular triplet extraction algorithm of a cross-covariance matrix.

1.4 Organization of This Book

As reflected in the title, this book is concerned with three areas of principal component analysis method, namely neural-based algorithm, performance analysis method, and generalized/extension algorithm. Consequently, the book can be naturally divided into three parts with a common theme. In the three areas, many novel algorithms were proposed by us. To appreciate these new algorithms, the conventional approaches and existing methods also need to be understood. Fundamental knowledge of conventional principal component analysis, neural-based feature extraction, subspace tracking, performance analysis methods, and even feature extraction based on matrix theory is essential for understanding the advanced material presented in this book. Thus, each part of this book starts with a tutorial type of introduction of the area.

Part I starts from Chap. 2, which provides an overview of some important concepts and theorems of decomposition and singular value decomposition related to principal component analysis. Chapter 3 serves as a starting point to introduce the neural-based principal component analysis. The key Hebbian network and Oja's network forming the core of neural network-based PCA algorithms can be founded in this chapter. Chapter 4 provides an introduction to neural network-based MCA algorithms and the self-stabilizing analysis of these algorithms, followed by a novel self-stabilizing MCA algorithm and a novel neural algorithm for total least squares filtering proposed by us. Part I ends on Chap. 5, which addresses the theoretical issue of the dual-purpose principal and minor component analysis. In this chapter, several important dual-purpose algorithms proposed by us are introduced, and their performance and numerical consideration are analyzed. Part II starts from a tutorial-style introduction to deterministic continuous-time (DCT) system, the stochastic discrete-time (SDT) system, the deterministic discrete-time (DDT) system, followed by a detailed analysis of DDT systems of a new self-stabilizing MCA algorithm and Chen's unified PCA/MCA algorithm in Chap. 6. Part III starts from Chap. 7. The generalized Hermitian eigenvalue problem and existing adaptive algorithms to extract generalized eigen pairs are reviewed, and then, a minor generalized eigenvector extraction algorithm and a novel adaptive algorithm for generalized coupled eigen pairs of ours are introduced and discussed. The other two chapters of Part III are devoted to coupled principal component analysis and cross-correlation feature extraction, respectively, in which our novel coupled or extension algorithms are introduced and analyzed.

Some of the materials presented in this book have been published in archival journals by the authors, and is included in this book after necessary modifications or updates (some modifications are major ones) to ensure accuracy, relevance, completeness and coherence. This portion of materials includes:

- Section 4.4 of Chapter 4, reprinted from *Neural Networks*, Xiangyu Kong, Changhua Hu, Chongzhao Han, "A self-stabilizing MSA algorithm in high-dimensional data stream", Vol. 23, 865–871, © 2010 Elsevier Ltd., with permission from Elsevier.

- Section 4.5 of Chapter 4, reprinted from Neural Processing Letter, Xiangyu Kong, Changhua Hu, Chongzhao Han, “A self-stabilizing neural algorithm for total least squares filtering”, Vol. 30, 257–271, © 2009 Springer Science+Business Media, LLC., reprinted with permission.
- Section 5.3 of Chapter 5, reprinted from IEEE Transactions on Signal Processing, Xiangyu Kong, Changhua Hu, Chongzhao Han, “A Dual purpose principal and minor subspace gradient flow”, Vol. 60, No. 1, 197–210, © 2012 IEEE., with permission from IEEE.
- Section 6.3 of Chapter 6, reprinted from IEEE Transactions on Neural Networks, Xiangyu Kong, Changhua Hu, Chongzhao Han, “On the discrete time dynamics of a class of self-stabilizing MCA learning algorithm”, Vol. 21, No. 1, 175–181, © 2010 IEEE., with permission from IEEE.
- Section 6.4 of Chapter 6, reprinted from Neural Networks, Xiangyu Kong, Qiusheng an, Hongguang Ma, Chongzhao Han, Qizhang, “Convergence analysis of deterministic discrete time system of a unified self-stabilizing algorithm for PCA and MCA”, Vol. 36, 64–72, © 2012 Elsevier Ltd., with permission from Elsevier.
- Section 7.3 and 7.4 of Chapter 7, reprinted from IEEE Transactions on Signal Processing, Gao Yingbin, Kong Xiangyu, Hu Changhua, Li Hongzeng, and Hou Li'an, “A Generalized Information Criterion for generalized Minor Component Extraction”, Vol. 65, No. 4, 947–959, © 2017 IEEE., with permission from IEEE.
- Section 8.3 of Chapter 8, reprinted from Neural Processing Letter, Xiaowei Feng, Xiangyu Kong, Hongguang Ma, and Haomiao Liu, “Unified and coupled self-stabilizing algorithm for minor and principal eigen-pair extraction”, doi: [10.1007/s11063-016-9520-3](https://doi.org/10.1007/s11063-016-9520-3), © 2016 Springer Science+Business Media, LLC., reprinted with permission.
- Section 8.4 of Chapter 8, reprinted from IEEE Transactions on Signal Processing, Xiaowei Feng, Xiangyu Kong, Zhansheng Duan, and Hongguang Ma, “Adaptive generalized eigen-pairs extraction algorithm and their convergence analysis”, Vol. 64, No. 11, 2976–2989, © 2016 IEEE., with permission from IEEE.
- Section 9.3 of Chapter 9, reprinted from Neural Processing Letter, Xiangyu Kong, Hongguang Ma, Qiu sheng An, Qi Zhang, “An effective neural learning algorithm for extracting cross-correlation feature between two high-dimensional data streams”, Vol. 42, 459–477, © 2015 Springer Science+Business Media, LLC., reprinted with permission.

References

1. Diamantaras, K. I., & Kung, S. Y. (1996). *Principal component neural networks: Theory and application*. Wiley, INC.
2. Zhang, Q., & Leung, Y.-W. (2000). A class of learning algorithms for principal component analysis and minor component analysis. *IEEE Transactions on Neural Network*, 11(1), 200–204.

3. Washizawa, Y. (2010). Feature extraction using constrained approximation and suppression. *IEEE Transactions on Neural Network*, 21(2), 201–210.
4. Cirrincione, G., Cirrincione, M., Heralut, J., & Huffel, S. V. (2002). The MCA EXIN neuron for the minor component analysis. *IEEE Transactions on Neural Network*, 13(1), 160–187.
5. Yang, B. (1995). Projection approximation subspace tracking. *IEEE Transactions on Signal Processing*, 43(1), 95–107.
6. Comon, P., & Golub, G. H. (1990). Tracking a few extreme singular values and vectors in signal processing. In *Processing of the IEEE* (pp. 1327–1343).
7. Owsley, N. L. (1978). Adaptive data orthogonalization. In *Proceedings of IEEE ICASSP* (pp. 100–112).
8. Tufts, D. W., & Melissinos, C. D. (1986). Simple, effective computation of principal eigenvectors and their eigenvalues and applications to high-resolution estimation of frequencies. *IEEE Transactions on Acoustic, Speech and Signal Processing*, ASSP-34, 1046–1053.
9. Shaman, K. C. (1986). Adaptive algorithms for estimating the complete covariance eigenstructure. In *Proceedings of IEEE ICASSP (Tokyo, Japan)* (pp. 1401–1404).
10. Moonen, M., Van Dooren, P., & Vandewalle, J. (1989). Updating singular value decompositions: A parallel implementation. In *PWC. SPIE Adv. Algorithms Arthitectures Signal Processing* (San Diego, CA) (pp. 80–91).
11. Bunch, J. R., Nielsen, C. P., & Sorenson, D. (1978). Rank-one modification of the symmetric eigenproblem. *Numerical Mathematics*, 31, 31–48.
12. Karasalo, I. (1986). Estimating the covariance matrix by signal subspace averaging. *IEEE Transactions on Acoustic Speech and Signal Processing*, ASSP-34, 8–12.
13. DeGroat, R. D. (1992). Noniterative subspace tracking. *IEEE Transactions on Signal Processing*, 40(3), 571–577.
14. Thompson, P. A. (1980). An adaptive spectral analysis technique for unbiased frequency estimation in the presence of white noise. In *Proceeding 13th Asilnmur Conj:Circuit System. Computation* (pp. 529–533).
15. Oja, E. (1982). A simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15(3), 267–273.
16. Karhunen, J., & Oja, E. (1982). New methods for stochastic approximation of truncated Karhunen-LoCve expansions. In *Prw. 6th International Conference: Putt. Recop.* (5Xk553).
17. Karhunen, J. (1984). Adaptive algorithms for estimating eigenvectors of correlation type matrices. In *Proceedings of IEEE ICASSP* (San Diego, CA) (pp. 14.6.1–14.6.3).
18. Yang, J., & Kaveh, M. (1988). Adaptive eigensubspace algorithms for direction or frequency estimation and tracking. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 36 (2), 241–251.
19. Kung, S. Y. (1993). *Digital neural processing*. Englewood Cliffs, NJ: Prentice Hall.
20. Reddy, V. U., Egardt, B., & Kailath, T. (1982). Least squares type algorithm for adaptive implementation of Pisarenko’s harmonic retrieval method. *IEEE Transactions on Acoustic Speech and Signal processing*, 30(3), 399–405.
21. Bannour, S., & Azimi-Sadjadi, M. R. (1992). An adaptive approach for optimal data reduction using recursive least squares learning method. In *Pwc. IEEE ICASSP* (San Francisco, CA) (pp. 11297–11300).
22. Yang, X., Sarkar, T. K., & Arvas, E. (1989). A survey of conjugate gradient algorithms for solution of extreme eigen-problems of a symmetric matrix. *IEEE Transactions on Acoustic, Speech and Signal Processing*, 37(10), 1550–1556.
23. Stewart, G. W. (1992). An updating algorithm for subspace tracking. *IEEE Transactions on Signal Processing*, 40(6), 1535–1541.
24. Bischof, C. H., & Shroff, G. M. (1992). On updating signal subspaces. *IEEE Transactions on Signal Processing*, 40(1), 96–105.
25. Bannour, S., & Azimi-Sadjadi, R. (1995). Principal component extraction using recursive least squares learning. *IEEE Transactions on Neural Networks*, 6(2), 457–469.

26. Cichocki, A., Kasprzak, W., & Skarbek, W. (1996). Adaptive learning algorithm for principal component analysis with partial data. *Cybernetic Systems*, 2, 1014–1019.
27. Kung, S. Y., Diamantaras, K., & Taur, J. (1994). Adaptive principal component extraction (APEX) and applications. *IEEE Transactions on Signal Processing*, 42(5), 1202–1217.
28. Möller, R., & Könies, A. (2004). Coupled principal component analysis. *IEEE Transactions on Neural Networks*, 15(1), 214–222.
29. Ouyang, S., Bao, Z., & Liao, G. S. (2000). Robust recursive least squares learning algorithm for principal component analysis. *IEEE Transactions on Neural Networks*, 11(1), 215–221.
30. Sanger, T. D. (1989). Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks*, 2(6), 459–473.
31. Xu, L. (1993). Least mean square error reconstruction principle for selforganizing neural-nets. *Neural Networks*, 6(5), 627–648.
32. Xu, L., Oja, E., & Suen, C. (1992). Modified Hebbian learning for curve and surface fitting. *Neural Networks*, 5, 441–457.
33. Oja, E. (1992). Principal component, minor component and linear neural networks. *Neural Networks*, 5(6), 927–935.
34. Feng, D. Z., Bao, Z., & Jiao, L. C. (1998). Total least mean squares algorithm. *IEEE Transactions on Signal Processing*, 46(6), 2122–2130.
35. Feng, D. Z., Zheng, W. X., & Jia, Y. (2005). Neural network learning algorithms for tracking minor subspace in high-dimensional data stream. *IEEE Transactions on Neural Networks*, 16(3), 513–521.
36. Luo, F. L., & Unbehauen, R. (1997). A minor subspace analysis algorithm. *IEEE Transactions on Neural Networks*, 8(5), 1149–1155.
37. Ouyang, S., Bao, Z., Liao, G. S., & Ching, P. C. (2001). Adaptive minor component extraction with modular structure. *IEEE Transactions on Signal Processing*, 49(9), 2127–2137.
38. Zhang, Q., & Leung, Y.-W. (2000). A class of learning algorithms for principal component analysis and minor component analysis. *IEEE Transactions on Neural Networks*, 11(2), 529–533.
39. Möller, R. (2004). A self-stabilizing learning rule for minor component analysis. *International Journal of Neural Systems*, 14(1), 1–8.
40. Douglas, S. C., Kung, S. Y., & Amari, S. (2002). A self-stabilized minor subspace rule. *IEEE Signal Processing Letter*, 5(12), 1342–1352.
41. Oja, E. (1989). Neural networks, principal components, and subspaces. *International Journal of Neural Systems*, 1(1), 61–68.
42. Williams R. J. (1985). *Feature discovery through error-correction learning*. Institute of Cognition Science, University of California, San Diego, Technical Report. 8501.
43. Baldi, P. (1989). Linear learning: Landscapes and algorithms. In D. S. Touretzky (Ed.), *Advances in neural information processing systems 1*. San Mateo, CA: Morgan Kaufmann.
44. Miao, Y. F., & Hua, Y. B. (1998). Fast subspace tracking and neural network learning by a novel information criterion. *IEEE Transactions on Signal Processing*, 46(7), 1967–1979.
45. Fu, Z., & Dowling, E. M. (1995). Conjugate gradient eigenstructure tracking for adaptive spectral estimation. *IEEE Transactions on Signal Processing*, 43(5), 1151–1160.
46. Mathew, G., Reddy, V. U., & Dasgupta, S. (1995). Adaptive estimation of eigensubspace. *IEEE Transactions on Signal Processing*, 43(2), 401–411.
47. Chiang, C. T., & Chen, Y. H. (1999). On the inflation method in adaptive noise subspace estimator. *IEEE Transactions on Signal Processing*, 47(4), 1125–1129.
48. Nguyen, T. D., & Yamada, I. (2013). Adaptive normalized quasi-newton algorithms for extraction of generalized eigen-pairs and their convergence analysis. *IEEE Transactions on Signal Processing*, 61(6), 1404–1418.
49. Chatterjee, C., Roychowdhury, V., Ramos, P. J., & Zoltowski, M. D. (1997). Self-organizing algorithms for generalized eigen-decomposition. *IEEE Transactions on Neural Networks*, 8(6), 1518–1530.

50. Xu, D. X., Principe, J. C., & Wu, H. C. (1998). Generalized eigendecomposition with an on-line local algorithm. *IEEE Signal Processing Letter*, 5(11), 298–301.
51. Rao, Y. N., Principe, J. C., & Wong, T. F. (2004). Fast RLS-like algorithm for generalized eigendecomposition and its applications. *Journal of VLSI Signal Processing*, 37(2–3), 333–344.
52. Wong, T. F., Lok, T. M., Lehnert, J. S., & Zoltowski, M. D. (1998). A linear receiver for direct-sequence spread-spectrum multiple-access systems with antenna arrays and blind adaptation. *IEEE Transactions on Information Theory*, 44(2), 659–676.
53. Strobach, P. (1998). Fast orthogonal iteration adaptive algorithms for the generalized symmetric eigenproblem. *IEEE Transactions on Signal Processing*, 46(12), 3345–3359.
54. Yang, J., Xi, H. S., Yang, F., & Zhao, Y. (2006). RLS-based adaptive algorithms for generalized eigen-decomposition. *IEEE Transactions on Signal Processing*, 54(4), 1177–1188.
55. Tanaka, T. (2009). Fast generalized eigenvector tracking based on the power method. *IEEE Signal Processing Letter*, 16(11), 969–972.
56. Attallah, S., & Abed-Meraim, K. (2008). A fast adaptive algorithm for the generalized symmetric eigenvalue problem. *IEEE Signal Processing Letter*, 15, 797–800.
57. Chen, T. P., & Amari, S. (2001). Unified stabilization approach to principal and minor components extraction algorithms. *Neural Networks*, 14(10), 1377–1387.
58. Hasan, M. A. (2007). Self-normalizing dual systems for minor and principal component extraction. In *Proceedings of ICASSP 2007 IEEE International Conference on Acoustic, Speech and Signal Processing* (15–20, Vol. 4, pp. IV-885–IV-888).
59. Peng, D. Z., Zhang, Y., & Xiang, Y. (2009). A unified learning algorithm to extract principal and minor components. *Digit Signal Processing*, 19(4), 640–649.
60. Manton, J. H., Helmke, U., & Mareels, I. M. Y. (2005). A dual purpose principal and minor component flow. *Systems Control Letter*, 54(8), 759–769.
61. Chen, T., Amari, S. I., & Lin, Q. (1998). A unified algorithm for principal and minor component extraction. *Neural Networks*, 11(3), 365–369.
62. Peng, D. Z., & Zhang, Y. (2007). Dynamics of generalized PCA and MCA learning algorithms. *IEEE Transactions on Neural Networks*, 18(6), 1777–1784.
63. Kong, X. Y., Hu, C. H., & Han, C. Z. (2012). A dual purpose principal and minor subspace gradient flow. *IEEE Transactions on Signal Processing*, 60(1), 197–210.
64. Kong, X. Y., Hu, C. H., Ma, H. G., & Han, C. Z. (2012). A unified self-stabilizing neural network algorithm for principal and minor components extraction. *IEEE Transactions on Neural Networks and Learning Systems*, 23(2), 185–198.
65. Ferali, W., & Proakis J. G. (1990). Adaptive SVD algorithm for covariance matrix eigenstructure computation. In *Proceedings of IEEE International Conference on Acoustic, Speech, Signal Processing* (pp. 176–179).
66. Cichocki, A. (1992). Neural network for singular value decomposition. *Electronic Letter*, 28(8), 784–786.
67. Diamantaras, K. I., & Kung, S. Y. (1994). Cross-correlation neural network models. *IEEE Transactions on Signal Processing*, 42(11), 3218–3223.
68. Feng, D. Z., Bao, Z., & Zhang, X. D. (2001). A cross-associative neural network for SVD of nonsquared data matrix in signal processing. *IEEE Transactions on Neural Networks*, 12(9), 1215–1221.
69. Feng, D. Z., Zhang, X. D., & Bao, Z. (2004). A neural network learning for adaptively extracting cross-correlation features between two high-dimensional data streams. *IEEE Transactions on Neural Networks*, 15(6), 1541–1554.
70. Kong, X. Y., Ma, H. G., An, Q. S., & Zhang, Q. (2015). An effective neural learning algorithm for extracting cross-correlation feature between two high-dimensional data streams. *Neural Processing Letter*, 42(2), 459–477.
71. Zhang, X. D. (2004) *Matrix analysis and applications*, Tsinghua University Press, Beijing.
72. Bradbury, W. W., & Fletcher, R. (1966). New iterative methods for solutions of the eigenproblem. *Numerical Mathematics*, 9(9), 259–266.

73. Sarkar, T. K., Dianat, S. A., Chen, H., & Brule, J. D. (1986). Adaptive spectral estimation by the conjugate gradient method. *IEEE Transactions on Acoustic, Speech, and Signal Processing*, 34(2), 272–284.
74. Golub, G. H., & Van Loan, C. F. (1989). *Matrix computation* (2nd ed.). Baltimore: The John Hopkins University Press.
75. Golub, G. H. (1973). Some modified matrix eigenvalue problems. *SIAM Review*, 15(2), 318–334.
76. Bunch, J. R., Nielsen, C. P., & Sorensen, D. C. (1978). Rank-one modification of the symmetric eigenproblem. *Numerical Mathematics*, 31(1), 31–48.
77. Bunch, J. R., & Nielsen, C. P. (1978). Updating the singular value decomposition. *Numerical Mathematics*, 31(31), 111–129.
78. Schreiber, R. (1986). Implementation of adaptive array algorithms. *IEEE Transactions on Acoustic, Speech, and Signal Processing*, 34(5), 1038–1045.
79. Karasalo, I. (1986). Estimating the covariance matrix by signal subspace averaging. *IEEE Transactions on Acoustic, Speech, and Signal Processing*, 34(1), 8–12.
80. DeGroat, R. D., & Roberts, R. A. (1990). Efficient, numerically stabilized rank-one eigenstructure updating. *IEEE Transactions on Acoustic, Speech, and Signal Processing*, 38(2), 301–316.
81. Yu, K. B. (1991). Recursive updating the eigenvalue decomposition of a covariance matrix. *IEEE Transactions on Signal Processing*, 39(5), 1136–1145.
82. Stewart, G. W. (1992). An updating algorithm for subspace tracking. *IEEE Transactions on Signal Processing*, 40(6), 1135–1541.
83. Bischof, C. H., & Shroff, G. M. (1992). On updating signal subspace. *IEEE Transactions on Signal Processing*, 40(1), 96–105.
84. Champagne, B. (1994). Adaptive eigendecomposition of data covariance matrices based on first-order perturbations. *IEEE Transactions on Signal Processing*, 42(10), 2758–2770.
85. Fuhrmann, D. R. (1988). An algorithm for subspace computation with applications in signal processing. *SIAM Journal of Matrix Analysis and Applications*, 9(2), 213–220.
86. Xu, G., Cho, Y., & Kailath, T. (1994). Application of fast subspace decomposition to signal processing and communication problems. *IEEE Transactions on Signal Processing*, 42(6), 1453–1461.
87. Xu, G., & Kailath, T. (1994). Fast subspace decomposition. *IEEE Transactions on Signal Processing*, 42(3), 539–551.
88. Moonen, M., Dooren, P. V., & Vandewalle, J. (1992). A singular value decomposition updating algorithm for subspace tracking. *SIAM Journal of Matrix Analysis and Applications*, 13(4), 1015–1038.