CHAPTER 42

SUBSPACE TRACKING FOR SIGNAL PROCESSING

42.1 INTRODUCTION

Research in subspace and component-based techniques were originated in Statistics in the middle of the last century through the problem of linear feature extraction solved by the Karhunen-Loève Transform (KLT). Then, it application to signal processing was initiated three decades ago, and has met considerable progress. Thorough studies have shown that the estimation and detection tasks in many signal processing and communications applications such as data compression, data filtering, parameter estimation, pattern recognition, and neural analysis can be significantly improved by using the subspace and component-based methodology. Over the past few years new potential applications have emerged, and subspace and component methods have been adopted in several diverse new fields such as smart antennas, sensor arrays, multiuser detection, time delay estimation, image segmentation, speech enhancement, learning systems, magnetic resonance spectroscopy, and radar systems, to mention only a few examples. The interest in subspace and component-based methods stems from the fact that they consist in splitting the observations into a set of desired and a set of disturbing components. They not only provide new insight into many such problems, but they also offer a good tradeoff between achieved performance and computational complexity. In most cases they can be considered to be low-cost alternatives to computationally intensive maximum-likelihood approaches.

In general, subspace and component-based methods are obtained by using batch methods, such as the eigenvalue decomposition (EVD) of the sample covariance matrix or the singular value decomposition (SVD) of the data matrix. However, these two approaches are not suitable for adaptive applications for tracking nonstationary signal parameters, where the required repetitive estimation of the subspace or the eigenvectors can be a real computational burden because their iterative implementation needs $O(n^3)$ operations at each update, where n is the dimension of the vector-valued data sequence. Before proceeding with a brief literature review of the main contributions of adaptive estimation of subspace or eigenvectors, let us first classify these algorithms with respect to their computational complexity. If r denotes the rank of the principal or dominant) or minor subspace we would like to estimate, since usually $r \ll n$, it is classic to refer to the following classification. Algorithms requiring $O(n^2r)$ or $O(n^2)$ operations by update are classified as high complexity; algorithms with $O(nr^2)$ operations as medium complexity and finally, algorithms with $O(nr)$ operations as low complexity. This last category constitutes the most important one from a real time implementation point of view, and schemes belonging to this class are also known in the literature as fast subspace tracking algorithms. It should be mentioned that methods belonging to the high complexity class usually present faster convergence rates compared to the other two classes. From the paper by Owsley [55], that first introduced an adaptive procedure for the estimation of the signal subspace with $O(n^2r)$ operations, the literature referring to the problem of subspace or eigenvectors tracking from a signal processing point of view is extremely rich. The survey paper [20] constitutes an excellent review of results up to 1990, treating the first two classes, since the last class was not available at the time. The most popular algorithm of the medium class was proposed by Karasalo in [39]. In [20], it is stated that this dominant subspace algorithm offers the best performance to cost ratio and thus serves as a point of reference for subsequent algorithms by many authors. The merger of signal processing and neural networks in the early 1990s [38] brought much attention to a method originated by Oja [49] and applied by many others. The Oja method requires only $O(nr)$ operations at each update. It is clearly the continuous interest in the subject and significant recent developments that gave rise to this third class. It is out of the scope of this chapter to give a comprehensive survey of all the contributions, but rather to focus on some of them. The interested reader may refer to [28, pp. 30-43] for an exhaustive literature review and to [8] for tables containing exact computational complexities and ranking with respect to convergence of recent subspace tracking algorithms. In the present work, we mainly emphasize on the low complexity class for both dominant and minor subspace, and dominant and minor eigenvector tracking, while we briefly address the most important schemes of the other two classes. For these algorithms, we will focus on their derivation from different iterative procedures coming from linear algebra and on their theoretical convergence and performance in stationary environments. Many important issues such as the finite precisions effects on their behavior (e.g., possible numerical instabilities due to roundoff error accumulation), the different adaptive step size strategies and the tracking capabilities of these algorithms in nonstationary environments will be left aside. The interested reader may refer to the simulation Sections of the different papers that deal with these issues.

The derivation and analysis of algorithms for subspace tracking require a minimum background from linear algebra and matrix analysis. This is the reason why in Section 2, standard linear algebra materials necessary to this chapter are recalled. This is followed in Section 3 by the general studied observation model to fix the main notations and by the statement of the adaptive and tracking of principal or minor subspaces (or eigenvectors) problems. Then, Oja's neuron is introduced in Section 4 as a preliminary example to show that the subspace or component adaptive algorithms are derived empirically from different adaptations of standard iterative computational techniques issued from numerical methods. In Sections 5 and 6 different adaptive algorithms for principal (or minor) subspace and component analysis are introduced respectively. As for Oja's neuron, the majority of these algorithms can be viewed as some heuristic variations of the power method. These heuristic approaches need to be validated by convergence and performance analysis. Several tools such as the stability of the ordinary differential equation (ODE) associated with a stochastic approximation algorithm and the Gaussian approximation to address these points in stationary environment are given in Section 7. Some illustrative applications of principal and minor subspace tracking in signal processing are given in Section 8. Section 9 contains some concluding remarks. Finally, some exercices are proposed in Section 10, essentially to prove some properties and relations introduced in the other sections.

42.2 LINEAR ALGEBRA REVIEW

In this section several useful notions coming from linear algebra as the EVD, the QR decomposition and the variational characterization of eigenvalues/eigenvectors of real symmetric matrices, and matrix analysis as a class of standard subspace iterative computational techniques are recalled. Finally a characterization of the principal subspace of a covariance matrix derived from the minimization of a mean square error will complete this section.

42.2.1 Eigenvalue value decomposition

Let $\mathbf{C}$ be an $n \times n$ real symmetric [resp. complex Hermitian] matrix, which is also *non-negative definite* because $\mathbf{C}$ will represent throughout this chapter a covariance matrix. Then, there exists (see e.g., [36, Sec.2.5]) an orthonormal [resp. unitary] matrix $\mathbf{U} = [\mathbf{u}_1, ..., \mathbf{u}_n]$ and a real diagonal matrix $\mathbf{\Delta} = \mathrm{Diag}(\lambda_1, ..., \lambda_n)$ such that $\mathbf{C}$ can be decomposed[1] as follows

$$\mathbf{C} = \mathbf{U}\mathbf{\Delta}\mathbf{U}^T = \sum_{i=1}^{n} \lambda_i \mathbf{u}_i \mathbf{u}_i^T, \quad \text{[resp., } \mathbf{U}\mathbf{\Delta}\mathbf{U}^H = \sum_{i=1}^{n} \lambda_i \mathbf{u}_i \mathbf{u}_i^H\text{]}. \tag{42.2.1}$$

The diagonal elements of $\mathbf{\Delta}$ are called *eigenvalues* and arranged in decreasing order, satisfy $\lambda_1 \geq ... \geq \lambda_n > 0$, while the orthogonal columns $(\mathbf{u}_i)_{i=1,...,n}$ of $\mathbf{U}$ are the corresponding unit 2-norm *eigenvectors* of $\mathbf{C}$.

For the sake of simplicity, only real-valued data will be considered from the next subsection and throughout this chapter. The extension to complex-valued data is often straightforward by changing the transposition operator to the conjugate transposition one. But we note two difficulties. First, for simple[2] eigenvalues, the associated eigenvectors are unique up to a multiplicative sign in the real case, but only to a unit modulus constant in the complex case, and consequently a constraint ought to be added to fix them to avoid any discrepancies between the statistics observed in numerical simulations and the theoretical formulas. The interested reader by the consequences of this nonuniqueness on the derivation of the asymptotic variance of estimated eigenvectors from sample covariance matrices can refer to [33], (see also Exercices 42.1). Second, in the complex case, the second-order properties of multidimensional zero-mean random variables $\mathbf{x}$ are not characterized by the complex Hermitian covariance matrix $\mathrm{E}(\mathbf{x}\mathbf{x}^H)$ only, but also by the complex symmetric complementary covariance [57] matrix $\mathrm{E}(\mathbf{x}\mathbf{x}^T)$.

[1] Note that for non-negative real symmetric or complex Hermitian matrices, this EVD is identical to the SVD where the associated left and right singular vectors are identical.

[2] This is in contrast to multiple eigenvalues for which only the subspaces generated by the eigenvectors associated with these multiple eigenvalues are unique.

The computational complexity of the most efficient existing iterative algorithms that perform EVD of real symmetric matrices is cubic by iteration with respect to the matrix dimension (more details can be sought in [34, chap. 8]).

42.2.2 QR factorization

The QR factorization of an $n \times r$ real-valued matrix $\mathbf{W}$, with $n \geq r$ is defined as (see e.g., [36, Sec. 2.6])

$$\mathbf{W} = \mathbf{QR} = \mathbf{Q}_1\mathbf{R}_1, \tag{42.2.2}$$

where $\mathbf{Q}$ is an $n \times n$ orthonormal matrix, $\mathbf{R}$ an $n \times r$ upper triangular matrix, $\mathbf{Q}_1$ denotes the first r columns of $\mathbf{Q}$ and $\mathbf{R}_1$ the $r \times r$ matrix constituted with the first r rows of $\mathbf{R}$. If $\mathbf{W}$ is of full column rank, the columns of $\mathbf{Q}_1$ form an orthonormal basis for the range of $\mathbf{W}$. Furthermore, in this case the "skinny" factorization $\mathbf{Q}_1\mathbf{R}_1$ of $\mathbf{W}$ is unique if $\mathbf{R}_1$ is constrained to have positive diagonal entries. The computation of the QR decomposition can be performed in several ways. Existing methods are based on Householder, block Householder, Givens or fast Givens transformations. Alternatively, the Gram-Schmidt orthonormalization process or a more numerically stable variant called modified Gram-Schmidt can be used. The interested reader can seek details for the aforementioned QR implementations in [34, pp. 224-233]), where the complexity is of the order of $O(nr^2)$ operations.

42.2.3 Variational characterization of eigenvalues/eigenvectors of real symmetric matrices

The eigenvalues of a general $n \times n$ matrix $\mathbf{C}$ are only characterized as the roots of the associated characteristic equation. But for real symmetric matrices, they can be characterized as the solutions of a series of optimization problems. In particular, the largest λ_1 and the smallest λ_n eigenvalues of $\mathbf{C}$ are solutions of the following constrained maximum and minimum problem (see e.g., [36, Sec.4.2]).

$$\lambda_1 = \max_{\|\mathbf{w}\|_2=1,\ \mathbf{w}\in\mathcal{R}^n} \mathbf{w}^T\mathbf{C}\mathbf{w} \quad \text{and} \quad \lambda_n = \min_{\|\mathbf{w}\|_2=1,\ \mathbf{w}\in\mathcal{R}^n} \mathbf{w}^T\mathbf{C}\mathbf{w}. \tag{42.2.3}$$

Furthermore, the maximum and minimum are attained by the unit 2-norm eigenvectors $\mathbf{u}_1$ and $\mathbf{u}_n$ associated with λ_1 and λ_n respectively, which are unique up to a sign for simple eigenvalues λ_1 and λ_n. For non-zero vectors $\mathbf{w} \in \mathcal{R}^n$, the expression $\frac{\mathbf{w}^T\mathbf{C}\mathbf{w}}{\mathbf{w}^T\mathbf{w}}$ is known as the *Rayleigh's quotient* and the constrained maximization and minimization (42.2.3) can be replaced by the following unconstrained maximization and minimization

$$\lambda_1 = \max_{\mathbf{w}\neq\mathbf{0},\ \mathbf{w}\in\mathcal{R}^n} \frac{\mathbf{w}^T\mathbf{C}\mathbf{w}}{\mathbf{w}^T\mathbf{w}} \quad \text{and} \quad \lambda_n = \min_{\mathbf{w}\neq\mathbf{0},\ \mathbf{w}\in\mathcal{R}^n} \frac{\mathbf{w}^T\mathbf{C}\mathbf{w}}{\mathbf{w}^T\mathbf{w}}. \tag{42.2.4}$$

For simple eigenvalues $\lambda_1, \lambda_2, ..., \lambda_r$ or $\lambda_n, \lambda_{n-1}, ..., \lambda_{n-r+1}$, (42.2.3) extends by the following iterative constrained maximizations and minimizations (see e.g., [36, Sec.4.2])

$$\lambda_k = \max_{\|\mathbf{w}\|_2=1,\ \mathbf{w}\perp\mathbf{u}_1,\mathbf{u}_2,..,\mathbf{u}_{k-1},\ \mathbf{w}\in\mathcal{R}^n} \mathbf{w}^T\mathbf{C}\mathbf{w},\ k = 2,..,r \tag{42.2.5}$$

$$= \min_{\|\mathbf{w}\|_2=1,\ \mathbf{w}\perp\mathbf{u}_n,\mathbf{u}_{n-1},..,\mathbf{u}_{k+1},\ \mathbf{w}\in\mathcal{R}^n} \mathbf{w}^T\mathbf{C}\mathbf{w},\ k = n-1,..,n-r+1, \tag{42.2.6}$$

and the constrained maximum and minimum are attained by the unit 2-norm eigenvectors $\mathbf{u}_k$ associated with λ_k which are unique up to a sign.

Note that when $\lambda_r > \lambda_{r+1}$ or $\lambda_{n-r} > \lambda_{n-r+1}$, the following global constrained maximizations or minimizations (denoted *subspace criterion*)

$$\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}_r} \mathrm{Tr}(\mathbf{W}^T\mathbf{C}\mathbf{W}) = \max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}_r} \sum_{k=1}^{r} \mathbf{w}_k^T\mathbf{C}\mathbf{w}_k$$

or

$$\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}_r} \mathrm{Tr}(\mathbf{W}^T\mathbf{C}\mathbf{W}) = \min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}_r} \sum_{k=1}^{r} \mathbf{w}_k^T\mathbf{C}\mathbf{w}_k, \qquad (42.2.7)$$

where $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_r]$ is an arbitrary $n \times r$ matrix, have for solutions (see e.g., [69] and Exercice 42.6) $\mathbf{W} = [\mathbf{u}_1, ..., \mathbf{u}_r]\mathbf{Q}$ or $\mathbf{W} = [\mathbf{u}_{n-r+1}, ..., \mathbf{u}_n]\mathbf{Q}$ respectively, where $\mathbf{Q}$ is an arbitrary $r \times r$ orthogonal matrix. Thus, subspace criterion (42.2.7) determines the subspace spanned by $\{\mathbf{u}_1, ..., \mathbf{u}_r\}$ or $\{\mathbf{u}_{n-r+1}, ..., \mathbf{u}_n\}$, but does not specify the basis of this subspace at all.

Finally, when now, $\lambda_1 > \lambda_2 > ... > \lambda_r > \lambda_{r+1}$ or $\lambda_{n-r} > \lambda_{n-r+1} > ... > \lambda_{n-1} > \lambda_n$,[3] if $(\omega_k)_{k=1,..,r}$ denotes r arbitrary positive and different real numbers such that $\omega_1 > \omega_2 > ... > \omega_r > 0$, the following modification of subspace criterion (42.2.7) denoted *weighted subspace criterion*

$$\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}_r} \mathrm{Tr}(\mathbf{\Omega}\mathbf{W}^T\mathbf{C}\mathbf{W}) = \max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}_r} \sum_{k=1}^{r} \omega_k\mathbf{w}_k^T\mathbf{C}\mathbf{w}_k$$

or

$$\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}_r} \mathrm{Tr}(\mathbf{\Omega}\mathbf{W}^T\mathbf{C}\mathbf{W}) = \min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}_r} \sum_{k=1}^{r} \omega_k\mathbf{w}_k^T\mathbf{C}\mathbf{w}_k, \qquad (42.2.8)$$

with $\mathbf{\Omega} = \mathrm{Diag}(\omega_1, .., \omega_r)$, has [53] the unique solution $\{\pm\mathbf{u}_1, ..., \pm\mathbf{u}_r\}$ or $\{\pm\mathbf{u}_{n-r+1}, ..., \pm\mathbf{u}_n\}$, respectively.

42.2.4 Standard subspace iterative computational techniques

The first subspace problem consists in computing the eigenvector associated with the largest eigenvalue. The *power method* presented in the sequel is the simplest iterative techniques for this task. Under the condition that λ_1 is the unique dominant eigenvalue associated with $\mathbf{u}_1$ of the real symmetric matrix $\mathbf{C}$, and starting from arbitrary unit 2-norm $\mathbf{w}_0$ not orthogonal to $\mathbf{u}_1$, the following iterations produce a sequence $(\alpha_i, \mathbf{w}_i)$ that converges to the largest eigenvalue λ_1 and its corresponding eigenvector unit 2-norm $\pm\mathbf{u}_1$.

$$\begin{array}{rrcl} & \mathbf{w}_0 & \multicolumn{2}{l}{\text{arbitrary such that } {\mathbf{w}_0}^T\mathbf{u}_1 \neq 0} \\ \text{for } i = 0, 1, ... & \mathbf{w}'_{i+1} & = & \mathbf{C}\mathbf{w}_i \\ & \mathbf{w}_{i+1} & = & \mathbf{w}'_{i+1}/\|\mathbf{w}'_{i+1}\|_2 \\ & \alpha_{i+1} & = & \mathbf{w}_{i+1}^T\mathbf{C}\mathbf{w}_{i+1}. \end{array} \qquad (42.2.9)$$

The proof can be found in [34, p. 406], where the definition and the speed of this convergence are specified in the following. Define $\theta_i \in [0, \pi/2]$ by $\cos(\theta_i) \overset{\text{def}}{=} |\mathbf{w}_i^T\mathbf{u}_1|$ satisfying $\cos(\theta_0) \neq 0$, then

$$|\sin(\theta_i)| \leq \tan(\theta_0)\left|\frac{\lambda_2}{\lambda_1}\right|^i \quad \text{and} \quad |\alpha_i - \lambda_1| \leq |\lambda_1 - \lambda_n|\tan^2(\theta_0)\left|\frac{\lambda_2}{\lambda_1}\right|^{2i}. \qquad (42.2.10)$$

[3] Or simply $\lambda_1 > \lambda_2 > ... > ...\lambda_n$ when $r = n$, if we are interested by all the eigenvectors.

Consequently the convergence rate of the power method is exponential and proportional to the ratio $\left|\frac{\lambda_2}{\lambda_1}\right|^i$ for the eigenvector and to $\left|\frac{\lambda_2}{\lambda_1}\right|^{2i}$ for the associated eigenvalue. If $\mathbf{w}_0$ is selected randomly, the probability that this vector is orthogonal to $\mathbf{u}_1$ is equal to zero. Furthermore, if $\mathbf{w}_0$ is deliberately chosen orthogonal to $\mathbf{u}_1$, the effect of finite precision in arithmetic computations will introduce errors that will finally provoke loss of this orthogonality and therefore convergence to $\pm\mathbf{u}_1$.

Suppose now that $\mathbf{C}$ is non-negative. A straightforward generalization of the power method allows for the computation of the r eigenvectors associated with the r largest eigenvalues of $\mathbf{C}$ when its first $r+1$ eigenvalues are distinct, or of the subspace corresponding to the r largest eigenvalues of $\mathbf{C}$ when $\lambda_r > \lambda_{r+1}$ only. This method can be found in the literature under the name of *orthogonal iteration*, e.g., in [34], *subspace iteration*, e.g., in [56] or *simultaneous iteration method*, e.g., in [63]. First, consider the case where the $r+1$ largest eigenvalues of $\mathbf{C}$ are distinct. With $\mathbf{U}_r \overset{\text{def}}{=} [\mathbf{u}_1, ..., \mathbf{u}_r]$ and $\mathbf{\Delta}_r = \text{Diag}(\lambda_1, ..., \lambda_r)$, the following iterations produce a sequence $(\mathbf{\Lambda}_i, \mathbf{W}_i)$ that converges to $(\mathbf{\Delta}_r, [\pm\mathbf{u}_1, ..., \pm\mathbf{u}_r])$.

$$\begin{array}{rrcl} & \mathbf{W}_0 & \multicolumn{2}{l}{\text{arbitrary } n \times r \text{ matrix such that } \mathbf{W}_0^T\mathbf{U}_r \text{ not singular}} \\ \text{for } i = 0, 1, ... & \mathbf{W}'_{i+1} & = & \mathbf{C}\mathbf{W}_i \\ & \mathbf{W}'_{i+1} & = & \mathbf{W}_{i+1}\mathbf{R}_{i+1} \;\; \text{"skinny" QR factorization} \\ & \mathbf{\Lambda}_{i+1} & = & \text{Diag}\left(\mathbf{W}_{i+1}^T\mathbf{C}\mathbf{W}_{i+1}\right). \end{array} \quad (42.2.11)$$

The proof can be found in [34, p. 411]. The definition and the speed of this convergence are similar to those of the power method, it is exponential and proportional to $\left(\frac{\lambda_{r+1}}{\lambda_r}\right)^i$ for the eigenvectors and to $\left(\frac{\lambda_{r+1}}{\lambda_r}\right)^{2i}$ for the eigenvalues. Note that if $r = 1$, then this is just the power method. Moreover for arbitrary r, the sequence formed by the first column of $\mathbf{W}_i$ is precisely the sequence of vectors produced by the power method with the first column of $\mathbf{W}_0$ as starting vector.

Consider now the case where $\lambda_r > \lambda_{r+1}$. Then the following iteration method

$$\begin{array}{rrcl} & \mathbf{W}_0 & \multicolumn{2}{l}{\text{arbitrary } n \times r \text{ matrix such that } \mathbf{W}_0^T\mathbf{U}_r \text{ not singular}} \\ \text{for } i = 0, 1, ... & \mathbf{W}_{i+1} & = & \text{Orthonorm}\{\mathbf{C}\mathbf{W}_i\}, \end{array} \quad (42.2.12)$$

where the orthonormalization (Orthonorm) procedure is not necessarily given by the QR factorization, generates a sequence $\mathbf{W}_i$ that "converges" to the dominant subspace generated by $\{\mathbf{u}_1, ..., \mathbf{u}_r\}$ only. This means precisely that the sequence $\mathbf{W}_i\mathbf{W}_i^T$ (which here is a projection matrix because $\mathbf{W}_i^T\mathbf{W}_i = \mathbf{I}_r$) converges to the projection matrix $\mathbf{\Pi}_r \overset{\text{def}}{=} \mathbf{U}_r\mathbf{U}_r^T$. In the particular case where the QR factorization is used in the orthonormalization step, the speed of this convergence is exponential and proportional to $\left(\frac{\lambda_{r+1}}{\lambda_r}\right)^i$, i.e., more precisely [34, p. 411]

$$\|\mathbf{W}_i\mathbf{W}_i^T - \mathbf{\Pi}_r\|_2 \leq \tan(\theta)\left(\frac{\lambda_{r+1}}{\lambda_r}\right)^i$$

where $\theta \in [0, \pi/2]$ is specified by $\cos(\theta) = \min_{\mathbf{u}\in\text{Span}(\mathbf{W}_0), \mathbf{v}\in\text{Span}(\mathbf{U}_r)} \frac{|\mathbf{u}^T\mathbf{v}|}{\|\mathbf{u}\|_2\|\mathbf{v}\|_2} > 0$. This type of convergence is very specific. The r orthonormal columns of $\mathbf{W}_i$ do not necessary converge to a particular orthonormal basis of the dominant subspace generated by $\mathbf{u}_1, ..., \mathbf{u}_r$, but may eventually rotate in this dominant subspace as i increases. Note that the orthonormalization step (42.2.12) can be realized by other means that the QR decomposition. For example, extending the $r = 1$ case

$$\mathbf{w}_{i+1} = \mathbf{C}\mathbf{w}_i / \|\mathbf{C}\mathbf{w}_i\|_2 = \mathbf{C}\mathbf{w}_i \left(\mathbf{w}_i^T \mathbf{C}^2 \mathbf{w}_i\right)^{-1/2},$$

to arbitrary r, yields

$$\mathbf{W}_{i+1} = \mathbf{C}\mathbf{W}_i \left(\mathbf{W}_i^T \mathbf{C}^2 \mathbf{W}_i\right)^{-1/2}, \tag{42.2.13}$$

where the square root inverse of the matrix $\mathbf{W}_i^T \mathbf{C}^2 \mathbf{W}_i$ is defined by the EVD of the matrix with its eigenvalues replaced by their square root inverses. The speed of convergence of the associated algorithm is exponential and proportional to $\left(\frac{\lambda_{r+1}}{\lambda_r}\right)^i$ as well [37].

Finally, note that the power and the orthogonal iteration methods can be extended to obtain the minor subspace or eigenvectors by replacing the matrix $\mathbf{C}$ by $\mathbf{I}_n - \mu\mathbf{C}$ where $0 < \mu < 1/\lambda_1$ such that the eigenvalues $1 - \mu\lambda_n > ..., \geq 1 - \mu\lambda_1 > 0$ of $\mathbf{I}_n - \mu\mathbf{C}$ are strictly positive.

42.2.5 Characterization of the principal subspace of a covariance matrix from the minimization of a mean square error

In the particular case where the matrix $\mathbf{C}$ is the covariance of the zero-mean random variable $\mathbf{x}$, consider the scalar function $J(\mathbf{W})$ where $\mathbf{W}$ denotes an arbitrary $n \times r$ matrix

$$J(\mathbf{W}) \stackrel{\text{def}}{=} \mathrm{E}(\|\mathbf{x} - \mathbf{W}\mathbf{W}^T\mathbf{x}\|^2). \tag{42.2.14}$$

The following two properties are proved (e.g., see [70] and Exercices 42.7 and 42.8):

First, the stationary points $\mathbf{W}$ of $J(\mathbf{W})$ (i.e., the points $\mathbf{W}$ that cancel $J(\mathbf{W})$) are given by $\mathbf{W} = \mathbf{U}_r\mathbf{Q}$ where the r columns of $\mathbf{U}_r$ denotes here arbitrary r distinct unit-2 norm eigenvectors among $\mathbf{u}_1, ..., \mathbf{u}_n$ of $\mathbf{C}$ and where $\mathbf{Q}$ is an arbitrary $r \times r$ orthogonal matrix. Furthermore at each stationary point, $J(\mathbf{W})$ equals the sum of eigenvalues whose eigenvectors are not included in $\mathbf{U}_r$.

Second, in the particular case where $\lambda_r > \lambda_{r+1}$, all stationary points of $J(\mathbf{W})$ are saddle points except the points $\mathbf{W}$ whose associated matrix $\mathbf{U}_r$ contains the r dominant eigenvectors $\mathbf{u}_1, ..., \mathbf{u}_r$ of $\mathbf{C}$. In this case $J(\mathbf{W})$ attains the global minimum $\sum_{i=r+1}^{n} \lambda_i$. It is important to note that at this global minimum, $\mathbf{W}$ does not necessarily contain the r dominant eigenvectors $\mathbf{u}_1, ..., \mathbf{u}_r$ of $\mathbf{C}$, but rather an arbitrary orthogonal basis of the associated dominant subspace. This is not surprising because

$$J(\mathbf{W}) = \mathrm{Tr}(\mathbf{C}) - 2\mathrm{Tr}(\mathbf{W}^T\mathbf{C}\mathbf{W}) + \mathrm{Tr}(\mathbf{W}\mathbf{W}^T\mathbf{C}\mathbf{W}\mathbf{W}^T)$$

with $\mathrm{Tr}(\mathbf{W}^T\mathbf{C}\mathbf{W}) = \mathrm{Tr}(\mathbf{C}\mathbf{W}\mathbf{W}^T)$ and thus $J(\mathbf{W})$ is expressed as a function of $\mathbf{W}$ through $\mathbf{W}\mathbf{W}^T$ which is invariant with respect to rotation $\mathbf{W}\mathbf{Q}$ of $\mathbf{W}$. Finally, note that when $r = 1$ and $\lambda_1 > \lambda_2$, the solution of the minimization of $J(\mathbf{w})$ (42.2.14) is given by the unit 2-norm dominant eigenvector $\pm\mathbf{u}_1$.

42.3 OBSERVATION MODEL AND PROBLEM STATEMENT

42.3.1 Observation model

The general iterative subspace determination problem described in the previous section, will be now specialized to a class of matrices $\mathbf{C}$ computed from observation data. In typical applications of subspace-based signal processing, a sequence[4] of data vectors $\mathbf{x}(k) \in \mathcal{R}^n$ is observed, satisfying the following very common observation signal model

$$\mathbf{x}(k) = \mathbf{s}(k) + \mathbf{n}(k), \tag{42.3.1}$$

where $\mathbf{s}(k)$ is a vector containing the information signal lying on an r-dimensional linear subspace of $\mathcal{R}^n$ with $r < n$, while $\mathbf{n}(k)$ is a zero-mean additive random white noise (AWN) random vector, uncorrelated from $\mathbf{s}(k)$. Note that $\mathbf{s}(k)$ is often given by $\mathbf{s}(k) = \mathbf{A}(k)\mathbf{r}(k)$ where the full rank $n \times r$ matrix $\mathbf{A}(k)$ is deterministically parameterized and $\mathbf{r}(k)$ is a r-dimensional zero-mean full random vector (i.e., with $\mathrm{E}\left(\mathbf{r}(k)\mathbf{r}^T(k)\right)$ non singular). The signal part $\mathbf{s}(k)$ may also randomly select among r deterministic vectors. This random selection does not necessarily result in a zero-mean signal vector $\mathbf{s}(k)$.

In these assumptions, the covariance matrix $\mathbf{C}_s(k)$ of $\mathbf{s}(k)$ is r-rank deficient and

$$\mathbf{C}_x(k) \stackrel{\text{def}}{=} \mathrm{E}\left(\mathbf{x}(k)\mathbf{x}^T(k)\right) = \mathbf{C}_s(k) + \sigma_n^2(k)\mathbf{I}_n, \tag{42.3.2}$$

where $\sigma_n^2(k)$ denotes the AWN power. Taking into account that $\mathbf{C}_s(k)$ is of rank r and applying the EVD (42.2.1) on $\mathbf{C}_x(k)$ yields

$$\mathbf{C}_x(k) = [\mathbf{U}_s(k), \mathbf{U}_n(k)] \left[\begin{array}{cc} \mathbf{\Delta}_s(k) + \sigma_n^2(k)\mathbf{I}_r & \mathbf{O} \\ \mathbf{O} & \sigma_n^2(k)\mathbf{I}_{n-r} \end{array} \right] \left[\begin{array}{c} \mathbf{U}_s^T(k) \\ \mathbf{U}_n^T(k) \end{array} \right], \tag{42.3.3}$$

where the $n \times r$ and $n \times (n-r)$ matrices $\mathbf{U}_s(k)$ and $\mathbf{U}_n(k)$ are orthonormal bases for the denoted *signal or dominant* and *noise or minor subspace* of $\mathbf{C}_x(k)$ and $\mathbf{\Delta}_s(k)$ is a $r \times r$ diagonal matrix constituted by the r non-zero eigenvalues of $\mathbf{C}_s(k)$. We note that the column vectors of $\mathbf{U}_s(k)$ are generally unique up to a sign, in contrast to the column vectors of $\mathbf{U}_n(k)$ for which $\mathbf{U}_n(k)$ is defined up to a right multiplication by a $(n-r) \times (n-r)$ orthonormal matrix $\mathbf{Q}$. However, the associated orthogonal projection matrices $\mathbf{\Pi}_s(k) \stackrel{\text{def}}{=} \mathbf{U}_s(k)\mathbf{U}_s^T(k)$ and $\mathbf{\Pi}_n(k) \stackrel{\text{def}}{=} \mathbf{U}_n(k)\mathbf{U}_n^T(k)$ respectively denoted *signal or dominant projection matrices* and *noise or minor projection matrices* that will be introduced in the next sections are both unique.

42.3.2 Statement of the problem

A very important problem in signal processing consists in continuously updating the estimate $\mathbf{U}_s(k)$, $\mathbf{U}_n(k)$, $\mathbf{\Pi}_s(k)$ or $\mathbf{\Pi}_n(k)$ and sometimes with $\mathbf{\Delta}_s(k)$ and $\sigma_n^2(k)$, assuming that we have available consecutive observation vectors $\mathbf{x}(i)$, $i = ..., k-1, k, ...$ when the signal or noise subspace is slowly time-varying compared to $\mathbf{x}(k)$. The dimension r of the signal subspace may be known a priori or estimated from the observation vectors. A straightforward way to come up with a method that solves these problems is to provide efficient adaptive estimates $\mathbf{C}(k)$ of $\mathbf{C}_x(k)$ and simply apply an EVD at each time step k. Candidates for this estimate $\mathbf{C}(k)$ are generally given by sliding windowed sample data covariance matrices when the sequence of $\mathbf{C}_x(k)$ undergoes relatively slow changes. With an *exponential window*, the estimated covariance matrix is defined as

$$\mathbf{C}(k) = \sum_{i=0}^{k} \beta^{k-i}\mathbf{x}(i)\mathbf{x}^T(i), \tag{42.3.4}$$

[4] Note that k generally represents successive instants, but it can also represent successive spatial coordinates (e.g., in [11] where k denotes the position of the secondary range cells in Radar.

where $0 < \beta < 1$ is the *forgetting factor*. Its use is intended to ensure that the data in the distant past are downweighted in order to afford the tracking capability when we operate in a nonstationary environment. $\mathbf{C}(k)$ can be recursively updated according to the following scheme:

$$\mathbf{C}(k) = \beta\mathbf{C}(k-1) + \mathbf{x}(k)\mathbf{x}^T(k). \qquad (42.3.5)$$

Note that

$$\mathbf{C}(k) = (1-\beta')\mathbf{C}(k-1) + \beta'\mathbf{x}(k)\mathbf{x}^T(k) = \mathbf{C}(k-1) + \beta'\left(\mathbf{x}(k)\mathbf{x}^T(k) - \mathbf{C}(k-1)\right) \qquad (42.3.6)$$

is also used. These estimates $\mathbf{C}(k)$ tend to smooth the variations of the signal parameters and so are only suitable for slowly changing signal parameters. For sudden signal parameter changes, the use of a *truncated window* may offer faster tracking. In this case, the estimated covariance matrix is derived from a window of length l

$$\mathbf{C}(k) = \sum_{i=k-l+1}^{k} \beta^{k-i}\mathbf{x}(i)\mathbf{x}^T(i), \qquad (42.3.7)$$

where $0 < \beta \leq 1$. The case $\beta = 1$ corresponds to a rectangular window. This matrix can be recursively updated according to the following scheme:

$$\mathbf{C}(k) = \beta\mathbf{C}(k-1) + \mathbf{x}(k)\mathbf{x}^T(k) - \beta^l\mathbf{x}(k-l)\mathbf{x}^T(k-l). \qquad (42.3.8)$$

Both versions require $O(n^2)$ operations with the first having smaller computational complexity and memory needs. Note that for $\beta = 0$, (42.3.8) gives the coarse estimate $\mathbf{x}(k)\mathbf{x}^T(k)$ of $\mathbf{C}_x(k)$ as used in the least mean square (LMS) algorithms for adaptive filtering (see e.g., [35]).

Applying an EVD on $\mathbf{C}(k)$ at each time k is of course the best possible way to estimate the eigenvectors or subspaces we are looking for. This approach is known as direct EVD and has high complexity which is $O(n^3)$. This method usually serves as point of reference when dealing with different less computationally demanding approaches described in the next sections. These computationally efficient algorithms will compute signal or noise eigenvectors (or signal or noise projection matrices) at the time instant $k+1$ from the associated estimate at time k and the new arriving sample vector $\mathbf{x}(k)$.

42.4 PRELIMINARY EXAMPLE: OJA'S NEURON

Let us introduce these adaptive procedures by a simple example: the following Oja's neuron originated by Oja [49] and then applied by many others that estimates the eigenvector associated with the unique largest eigenvalue of a covariance matrix of the stationary vector $\mathbf{x}(k)$.

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\{[\mathbf{I}_n - \mathbf{w}(k)\mathbf{w}^T(k)]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k)\}. \qquad (42.4.1)$$

The first term on the right side is the previous estimate of $\pm\mathbf{u}_1$, which is kept as a memory of the iteration. The whole term in the brackets is the new information. This term is scaled by the step size μ and then added to the previous estimate $\mathbf{w}(k)$ to obtain the current estimate $\mathbf{w}(k+1)$. We note that this new information is formed by two terms. The first one $\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k)$ contains the first step of the power method (42.2.9) and the second one is simply the previous estimate $\mathbf{w}(k)$ adjusted by the scalar $\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k)$ so that these two terms are on the same scale. Finally, we note that if the previous estimate $\mathbf{w}(k)$ is already the desired eigenvector $\pm\mathbf{u}_1$, the expectation of this new information is zero, and hence, $\mathbf{w}(k+1)$ will be hovering around $\pm\mathbf{u}_1$. The step size μ controls the balance between the past and the new information. Introduced in the neural networks literature [49] within the framework of a new synaptic modification law, it is interesting to note that this algorithm can be derived from different heuristic variations of numerical methods introduced in Section 42.2.

First consider the variational characterization recalled in Subsection 42.2.3. Because $\boldsymbol{\nabla}_{\mathbf{w}}(\mathbf{w}^T\mathbf{C}_x\mathbf{w}) = 2\mathbf{C}_x\mathbf{w}$, the constrained maximization (42.2.3) or (42.2.7) can be solved using the following constrained gradient-search procedure

$$\begin{aligned} \mathbf{w}'(k+1) &= \mathbf{w}(k) + \mu\mathbf{C}_x(k)\mathbf{w}(k) \\ \mathbf{w}(k+1) &= \mathbf{w}'(k+1)/\|\mathbf{w}'(k+1)\|_2, \end{aligned}$$

in which the step size μ is "sufficiency enough". Using the approximation $\mu^2 \ll \mu$ yields

$$\begin{aligned} \mathbf{w}'(k+1)/\|\mathbf{w}'(k+1)\|_2 &= (\mathbf{I}_n + \mu\mathbf{C}_x(k))\mathbf{w}(k)/(\mathbf{w}^T(k)(\mathbf{I}_n + \mu\mathbf{C}_x(k))^2\mathbf{w}(k))^{1/2} \\ &\approx (\mathbf{I}_n + \mu\mathbf{C}_x(k))\mathbf{w}(k)/(1 + 2\mu\mathbf{w}^T(k)\mathbf{C}_x(k)\mathbf{w}(k))^{1/2} \\ &\approx (\mathbf{I}_n + \mu\mathbf{C}_x(k))\mathbf{w}(k)(1 - \mu\mathbf{w}^T(k)\mathbf{C}_x(k)\mathbf{w}(k)) \\ &\approx \mathbf{w}(k) + \mu\left(\mathbf{I}_n - \mathbf{w}(k)\mathbf{w}^T(k)\right)\mathbf{C}_x(k)\mathbf{w}(k). \end{aligned}$$

Then, using the instantaneous estimate $\mathbf{x}(k)\mathbf{x}^T(k)$ of $\mathbf{C}_x(k)$, Oja's neuron (42.4.1) is derived.

Consider now the power method recalled in Subsection 42.2.4. Noticing that $\mathbf{C}_x$ and $\mathbf{I}_n + \mu\mathbf{C}_x$ have the same eigenvectors, the step $\mathbf{w}'_{i+1} = \mathbf{C}_x\mathbf{w}_i$ of (42.2.9) can be replaced by $\mathbf{w}'_{i+1} = (\mathbf{I}_n + \mu\mathbf{C}_x)\mathbf{w}_i$ and using the previous approximations yields Oja's neuron (42.4.1) anew.

Finally, consider the characterization of the eigenvector associated with the unique largest eigenvalue of a covariance matrix derived from the mean square error $\mathrm{E}(\|\mathbf{x} - \mathbf{w}\mathbf{w}^T\mathbf{x}\|^2)$ recalled in Subsection 42.2.5. Because

$$\boldsymbol{\nabla}_{\mathbf{w}}(\mathrm{E}(\|\mathbf{x} - \mathbf{w}\mathbf{w}^T\mathbf{x}\|^2) = 2\left(-2\mathbf{C}_x + \mathbf{C}_x\mathbf{w}\mathbf{w}^T + \mathbf{w}\mathbf{w}^T\mathbf{C}_x\right)\mathbf{w},$$

an unconstrained gradient-search procedure yields

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu\left(-2\mathbf{C}_x(k) + \mathbf{C}_x(k)\mathbf{w}(k)\mathbf{w}^T(k) + \mathbf{w}(k)\mathbf{w}^T(k)\mathbf{C}_x(k)\right)\mathbf{w}(k).$$

Then, using the instantaneous estimate $\mathbf{x}(k)\mathbf{x}^T(k)$ of $\mathbf{C}_x(k)$ and the approximation $\mathbf{w}^T(k)$ $\mathbf{w}(k) = 1$ justified by the convergence of the deterministic gradient-search procedure to $\pm\mathbf{u}_1$ when $\mu \to 0$, Oja's neuron (42.4.1) is derived again.

Furthermore, if we are interested in adaptively estimating the associated single eigenvalue λ_1, the minimization of the scalar function $J(\lambda) = (\lambda - \mathbf{u}_1^T\mathbf{C}_x\mathbf{u}_1)^2$ by a gradient-search procedure can be used. With the instantaneous estimate $\mathbf{x}(k)\mathbf{x}^T(k)$ of $\mathbf{C}_x(k)$ and with the estimate $\mathbf{w}(k)$ of $\mathbf{u}_1$ given by (42.4.1), the following stochastic gradient algorithm is obtained.

$$\lambda(k+1) = \lambda(k) + \mu\left(\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k) - \lambda(k)\right). \qquad (42.4.2)$$

We note that the previous two heuristic derivations could be extended to the adaptive estimation of the eigenvector associated with the unique smallest eigenvalue of $\mathbf{C}_x(k)$.

Using the constrained minimization (42.2.3) or (42.2.7) solved by a constrained gradient-search procedure or the power method (42.2.9) where the step $\mathbf{w}'_{i+1} = \mathbf{C}_x \mathbf{w}_i$ of (42.2.9) is replaced by $\mathbf{w}'_{i+1} = (\mathbf{I}_n - \mu \mathbf{C}_x)\mathbf{w}_i$ (where $0 < \mu < 1/\lambda_1$) yields (42.4.1) after the same derivation, but where the sign of the step size μ is reversed.

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu \left([\mathbf{I}_n - \mathbf{w}(k)\mathbf{w}^T(k)]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k) \right). \qquad (42.4.3)$$

The associated eigenvalue λ_n could be also derived from the minimization of $J(\lambda) = (\lambda - \mathbf{u}_n^T \mathbf{C}_x \mathbf{u}_n)^2$ and consequently obtained by (42.4.2) as well, where $\mathbf{w}(k)$ is issued from (42.4.3).

These heuristic approaches derived from iterative computational techniques issued from numerical methods recalled in Section 42.2, need to be validated by convergence and performance analysis for stationary data $\mathbf{x}(k)$. These issues will be considered in Section 42.7. In particular it will be proved that the coupled stochastic approximation algorithms (42.4.1),(42.4.2) in which the step size μ is decreasing, "converge" to the pair $(\pm\mathbf{u}_1, \lambda_1))$, in contrast to the stochastic approximation algorithm (42.4.3) that diverges. Then, due to the possible accumulation of rounding errors, the algorithms that converge theoretically must be tested through numerical experiments to check their numerical stability in stationary environments. Finally extensive Monte Carlo simulations must be carried out with various step sizes, initialization conditions, signal to noise ratios and parameters configurations in nonstationary environments.

42.5 SUBSPACE TRACKING

In this section, we consider the adaptive estimation of dominant (signal) and minor (noise) subspaces. To derive such algorithms from the linear algebra material recalled in Subsections 42.2.3, 42.2.4 and 42.2.5 similarly as for Oja's neuron, we first note that the general orthogonal iterative step (42.2.12): $\mathbf{W}_{i+1} = \text{Orthonorm}\{\mathbf{C}\mathbf{W}_i\}$ allows for the following variant for adaptive implementation

$$\mathbf{W}_{i+1} = \text{Orthonorm}\{(\mathbf{I}_n + \mu\mathbf{C})\mathbf{W}_i\}$$

where $\mu > 0$ is a "small" parameter known as *step size*, because $\mathbf{I}_n + \mu\mathbf{C}$ has the same eigenvectors as $\mathbf{C}$ with associated eigenvalues $(1 + \mu\lambda_i)_{i=1,\dots,n}$. Noting that $\mathbf{I}_n - \mu\mathbf{C}$ has also the same eigenvectors as $\mathbf{C}$ with associated eigenvalues $(1 - \mu\lambda_i)_{i=1,\dots,n}$, arranged exactly in the opposite order as $(\lambda_i)_{i=1,\dots,n}$ for μ sufficiently small ($\mu < 1/\lambda_1$), the general orthogonal iterative step (42.2.12) allows for the following second variant of this iterative procedure to "converge" to the r-dimensional minor subspace of $\mathbf{C}$ if $\lambda_{n-r} > \lambda_{n-r+1}$.

$$\mathbf{W}_{i+1} = \text{Orthonorm}\{(\mathbf{I}_n - \mu\mathbf{C})\mathbf{W}_i\}.$$

When the matrix $\mathbf{C}$ is unknown and, instead we have sequentially the data sequence $\mathbf{x}(k)$, we can replace $\mathbf{C}$ by an adaptive estimate $\mathbf{C}(k)$ (see Section 42.3.2). This leads to the adaptive orthogonal iteration algorithm

$$\mathbf{W}(k+1) = \text{Orthonorm}\{(\mathbf{I}_n \pm \mu_k \mathbf{C}(k))\mathbf{W}(k)\}, \qquad (42.5.1)$$

where the "+" sign generates estimates for the signal subspace (if $\lambda_r > \lambda_{r+1}$) and the "-" sign for the noise subspace (if $\lambda_{n-r} > \lambda_{n-r+1}$). Depending on the choice of the estimate $\mathbf{C}(k)$ and of the orthonormalization (or approximate orthonormalization), we can obtain alternative subspace tracking algorithms.

We note that maximization or minimization in (42.2.7) of $J(\mathbf{W}) \stackrel{\text{def}}{=} \mathrm{Tr}(\mathbf{W}^T\mathbf{C}\mathbf{W})$ subject to the constraint $\mathbf{W}^T\mathbf{W} = \mathbf{I}_r$ can be solved by a constrained gradient-descent technique. Because $\boldsymbol{\nabla}_{\mathbf{W}} J = 2\mathbf{C}(k)\mathbf{W}$, we obtain the following Rayleigh quotient-based algorithm

$$\mathbf{W}(k+1) = \text{Orthonorm}\{\mathbf{W}(k) \pm \mu_k \mathbf{C}(k)\mathbf{W}(k)\}, \tag{42.5.2}$$

whose general expression is the same as general expression (42.5.1) derived from the orthogonal iteration approach. We will denote this family of algorithms as the power-based methods. It is interesting to note that a simple sign change enables one to switch from the dominant to minor subspaces. Unfortunately, similarly to Oja's neuron, many minor subspace algorithms will be unstable or stable but non robust (i.e., numerically unstable with a tendency to accumulate round-off errors until their estimates are meaningless), in contrast to the associated majorant subspace algorithms. Consequently, the literature of minor subspace tracking techniques is very limited as compared to the wide variety of methods that exists for the tracking of majorant subspaces.

42.5.1 Subspace power-based methods

Clearly the simplest selection for $\mathbf{C}(k)$ is the instantaneous estimate $\mathbf{x}(k)\mathbf{x}^T(k)$, which gives rise to the *Data Projection Method* (DPM) first introduced in [69] where the orthonormalization is performed using the Gram-Schmidt procedure.

$$\mathbf{W}(k+1) = \text{GS Orth.}\{\mathbf{W}(k) \pm \mu_k \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\}. \tag{42.5.3}$$

In nonstationary situations, estimates (42.3.5) or (42.3.6) of the covariance $\mathbf{C}_x(k)$ of $\mathbf{x}(k)$ at time k have been tested in [69]. For this algorithm to "converge", we need to select a step size μ such that $\mu \ll 1/\lambda_1$ (see e.g., [28]). To satisfy this requirement (in nonstationary situations included) and because most of the time we have $\mathrm{Tr}(\mathbf{C}_x(k)) \gg \lambda_1(k)$, the following two normalized step sizes have been proposed in [69]:

$$\mu_k = \frac{\mu}{\|\mathbf{x}(k)\|^2} \;\text{ and }\; \mu_k = \frac{\mu}{\sigma_x^2(k)} \;\text{ with }\; \sigma_x^2(k+1) = \nu\sigma_x^2(k) + (1-\nu)\|\mathbf{x}(k)\|^2,$$

where μ may be close to unity and where the choice of $\nu \in (0,1)$ depends on the rapidity of the change of the parameters of the observation signal model (42.3.1). Note that a better numerical stability can be achieved [5] if μ_k is chosen, similar to the normalized LMS algorithm [35], as $\mu_k = \frac{\mu}{\|\mathbf{x}(k)\|^2+\alpha}$ where α is a "very small" positive constant. Obviously, this algorithm (42.5.3) has very high computational complexity due to the Gram-Schmidt orthonormalization step.

To reduce this computational complexity, many algorithms have been proposed. Going back to the DPM algorithm (42.5.3), we observe that we can write

$$\mathbf{W}(k+1) = \{\mathbf{W}(k) \pm \mu_k \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\}\mathbf{G}(k+1), \tag{42.5.4}$$

where the matrix $\mathbf{G}(k+1)$ is responsable for performing exact or approximate orthonormalization while preserving the space generated by the columns of $\mathbf{W}'(k+1) \stackrel{\text{def}}{=} \mathbf{W}(k) \pm \mu_k \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)$. It is the different choices of $\mathbf{G}(k+1)$ that will pave the way to alternative less computationally demanding algorithms. Depending on whether to this orthonormalization is exact or approximate, two families of algorithms have been proposed in the literature.

42.5.1.1 The approximate symmetric orthonormalization family The columns of $\mathbf{W}'(k+1)$ can be approximately orthonormalized in a symmetrical way. Since $\mathbf{W}(k)$ has orthonormal columns, for sufficiently small μ_k the columns of $\mathbf{W}'(k+1)$ will be linearly independent, although not orthonormal. Then $\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)$ is positive definite, and $\mathbf{W}(k+1)$ will have orthonormal columns if $\mathbf{G}(k+1) = \{\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)\}^{-1/2}$ (unique if $\mathbf{G}(k+1)$ is constrained to be symmetric). A stochastic algorithm denoted *Subspace Network Learning* (SNL) and later *Oja's algorithm* have been derived in [52] to estimate dominant subspace. Assuming μ_k is sufficiency enough, $\mathbf{G}(k+1)$ can be expanded in μ_k as follows

$$\begin{aligned}\mathbf{G}(k+1) &= \{\left(\mathbf{W}(k)+\mu_k\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\right)^T\left(\mathbf{W}(k)+\mu_k\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\right)\}^{-1/2}\\ &= \{\mathbf{I}_r + 2\mu_k\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k) + O(\mu_k^2)\}^{-1/2}\\ &= \mathbf{I}_r - \mu_k\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k) + O(\mu_k^2).\end{aligned}$$

Omitting second-order terms, the resulting algorithm reads[5]

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \mu_k[\mathbf{I}_n - \mathbf{W}(k)\mathbf{W}^T(k)]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k). \qquad (42.5.5)$$

The convergence of this algorithm has been earlier studied in [77] and then in [68], where it was shown that the solution $\mathbf{W}(t)$ of its associated ODE (see Subsection 42.7.1) need not tend to the eigenvectors $\{\mathbf{v}_1,\ldots,\mathbf{v}_r\}$, but only to a rotated basis $\mathbf{W}_*$ of the subspace spanned by them. More precisely, it has been proved in [16] that under the assumption that $\mathbf{W}(0)$ is of full column rank such that its projection to the signal subspace of $\mathbf{C}_x$ is linearly independent, there exists a rotated basis $\mathbf{W}_*$ of this signal subspace such that $\|\mathbf{W}(t)-\mathbf{W}_*\|_{\mathrm{Fro}} = O(e^{-(\lambda_r-\lambda_{r+1})t})$. A performance analysis has been given in [24, 25]. This issue will be used as an example analysis of convergence and performance in Subsection 42.7.3.2. Note that replacing $\mathbf{x}(k)\mathbf{x}^T(k)$ by $\beta\mathbf{I}_n \pm \mathbf{x}(k)\mathbf{x}^T(k)$ (with $\beta > 0$) in (42.5.5), leads to a *modified Oja's algorithm* [15], which, not affecting its capability of tracking a signal subspace with the sign "+", can track a noise subspace by changing the sign (if $\beta > \lambda_1$). Of course, these modified Oja's algorithms enjoy the same convergence properties as Oja's algorithm (42.5.5).

Many other modifications of Oja's algorithm have appeared in the literature, particularly to adapt it to noise subspace tracking. To obtain such algorithms, it is interesting to point out that, in general, it is not possible to obtain noise subspace tracking algorithms by simply changing the sign of the step size of a signal subspace tracking algorithm. For example, changing the sign in (42.5.5) or (42.7.18) leads to an unstable algorithm (divergence) as will be explained in Subsection 42.7.3.1 for $r = 1$. Among these modified Oja's algorithms, Chen *et al.* [16] have proposed the following unified algorithm

$$\begin{aligned}\mathbf{W}(k+1) &= \mathbf{W}(k) \pm \mu_k[\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{W}^T(k)\mathbf{W}(k)\\ &\qquad -\mathbf{W}(k)\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)], \qquad (42.5.6)\end{aligned}$$

where the signs "+" and"-" are respectively associated with signal and noise tracking algorithms. While the associated ODE maintains $\mathbf{W}^T(t)\mathbf{W}(t) = \mathbf{I}_r$ if $\mathbf{W}^T(0)\mathbf{W}(0) = \mathbf{I}_r$ and enjoys [16] the same stability properties as Oja's algorithm, the stochastic approximation

[5] Note that this algorithm can be directly deduced from the optimization of the cost function $J(\mathbf{W}) = \mathrm{Tr}[\mathbf{W}^T\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}]$ defined on the set of $n\times r$ orthogonal matrices $\mathbf{W}$ ($\mathbf{W}^T\mathbf{W} = \mathbf{I}_r$) with the help of continuous-time matrix algorithms [21, Ch. 7.2] (see also (42.9.7) in Exercice 42.15).

to algorithm (42.5.6) suffers from numerical instabilities (see e.g., numerical simulations in [27]). Thus, its practical use requires periodic column reorthonormalization. To avoid these numerical instabilities, this algorithm has been modified [17] by adding the penalty term $\mathbf{W}(k)[\mathbf{I}_n - \mathbf{W}(k)\mathbf{W}^T(k)]$ to the field of (42.5.6). As far as noise subspace tracking is concerned, Douglas *et al.* [27] have proposed modifying the algorithm (42.5.6) by multiplying the first term of its field by $\mathbf{W}^T(k)\mathbf{W}(k)$ whose associated term in the ODE tends to $\mathbf{I}_r$, viz

$$\begin{aligned}\mathbf{W}(k+1) \quad = \quad & \mathbf{W}(k) - \mu_k[\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{W}^T(k)\mathbf{W}(k)\mathbf{W}^T(k)\mathbf{W}(k) \\ & -\mathbf{W}(k)\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)]. \end{aligned} \quad (42.5.7)$$

It is proved in [27] that the locally asymptotically stable points $\mathbf{W}$ of the ODE associated with this algorithm satisfy $\mathbf{W}^T\mathbf{W} = \mathbf{I}_r$ and $\mathrm{Span}(\mathbf{W}) = \mathrm{Span}(\mathbf{U}_n)$. But the solution $\mathbf{W}(t)$ of the associated ODE does not converge to a particular basis $\mathbf{W}_*$ of the noise subspace but rather, it is proved that $\mathrm{Span}(\mathbf{W}(t))$ tends to $\mathrm{Span}(\mathbf{U}_n)$ (in the sense that the projection matrix associated with the subspace $\mathrm{Span}(\mathbf{W}(t))$ tends to $\mathbf{\Pi}_n$). Numerical simulations presented in [27] show that this algorithm is numerically more stable than the minor subspace version of algorithm (42.5.6).

To eliminate the instability of the noise tracking algorithm derived from Oja's algorithm (42.5.5) where the sign of the step size is changed, Abed Meraim *et al.* [2] have proposed forcing the estimate $\mathbf{W}(k)$ to be orthonormal at each time step k (see Exercice 42.10) that can be used for signal subspace tracking (by reversing the sign of the step size) as well. But this algorithm converges with the same speed of convergence as Oja's algorithm (42.5.5). To accelerate its convergence, two normalized versions (denoted *Normalized Oja's algorithm* (NOja) and *Normalized Orthogonal Oja's algorithm* (NOOJa)) of this algorithm have been proposed in [4]. They can perform both signal and noise tracking by switching the sign of the step size for which an approximate closed-form expression has been derived. A convergence analysis of the NOja algorithm has been presented in [7] using the ODE approach. Because the ODE associated with the field of this stochastic approximation algorithm is the same as those associated with the projection approximation-based algorithm (42.5.18), it enjoys the same convergence properties.

42.5.1.2 The exact orthonormalization family

The orthonormalization (42.5.4) of the columns of $\mathbf{W}'(k+1)$ can be performed exactly at each iteration by the symmetric square root inverse of $\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)$ due to the fact that the latter is a rank one modification of the identity matrix:

$$\mathbf{W}'^T(k+1)\mathbf{W}'(k+1) = \mathbf{I}_r \pm \left(2\mu_k \pm \mu_k^2\|\mathbf{x}(k)\|^2\right)\mathbf{y}(k)\mathbf{y}^T(k) \stackrel{\text{def}}{=} \mathbf{I}_r \pm \mathbf{z}\mathbf{z}^T \quad (42.5.8)$$

with $\mathbf{y}(k) \stackrel{\text{def}}{=} \mathbf{W}^T(k)\mathbf{x}(k)$ and $\mathbf{z} \stackrel{\text{def}}{=} \sqrt{2\mu_k \pm \mu_k^2\|\mathbf{x}(k)\|^2}\,\mathbf{y}(k)$. Using the identity

$$\left(\mathbf{I}_r \pm \mathbf{z}\mathbf{z}^T\right)^{-1/2} = \mathbf{I}_r + \left(\frac{1}{(1 \pm \|\mathbf{z}\|^2)^{1/2}} - 1\right)\frac{\mathbf{z}\mathbf{z}^T}{\|\mathbf{z}\|^2}, \quad (42.5.9)$$

we obtain

$$\mathbf{G}(k+1) = \{\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)\}^{-1/2} = \mathbf{I}_r + \tau_k\mathbf{y}(k)\mathbf{y}^T(k) \quad (42.5.10)$$

with $\tau_k \stackrel{\text{def}}{=} \left(\frac{1}{\left(1\pm(2\mu_k\pm\mu_k^2\|\mathbf{x}(k)\|^2)\|\mathbf{y}(k)\|^2\right)^{1/2}} - 1\right)\frac{1}{\|\mathbf{y}(k)\|^2}$. Substituting (42.5.10) into (42.5.4) leads to

$$\mathbf{W}(k+1) = \mathbf{W}(k) \pm \mu_k\mathbf{p}(k)\mathbf{x}^T(k)\mathbf{W}(k), \quad (42.5.11)$$

where $\mathbf{p}(k) \stackrel{\text{def}}{=} \pm\frac{\tau_k}{\mu_k}\mathbf{W}(k)\mathbf{y}(k) + (1+\tau_k\|\mathbf{y}(k)\|^2)\mathbf{x}(k)$. All these steps lead to the *Fast Rayleigh quotient-based Adaptive Noise Subspace* algorithm (FRANS) introduced by Attallah *et al.* in [5]. As stated in [5], this algorithm is stable and robust in the case of signal subspace tracking (associated with the sign "+") including initialization with a nonorthonormal matrix $\mathbf{W}(0)$. By contrast, in the case of noise subspace tracking (associated with the sign "-"), this algorithm is numerically unstable because of round-off error accumulation. Even when initialized with an orthonormal matrix, it requires periodic re-orthonormalization of $\mathbf{W}(k)$ in order to maintain the orthonormality of the columns of $\mathbf{W}(k)$. To remedy this instability, another implementation of this algorithm based on the numerically well behaved Householder transform has been proposed [6]. This Householder FRANS algorithm (HFRANS) comes from (42.5.11) which can be rewritten after cumbersome manipulations as

$$\mathbf{W}(k+1) = \mathbf{H}(k)\mathbf{W}(k) \quad \text{with} \quad \mathbf{H}(k) = \mathbf{I}_n - 2\mathbf{u}(k)\mathbf{u}^T(k)$$

with $\mathbf{u}(k) \stackrel{\text{def}}{=} \frac{\mathbf{p}(k)}{\|\mathbf{p}(k)\|_2}$. With no additional numerical complexity, this Householder transform allows one to stabilize the noise subspace version of the FRANS algorithm[6]. The interested reader may refer to [74] that analyzes the orthonormal error propagation (i.e., a recursion of the distance to orthonormality $\|\mathbf{W}^T(k)\mathbf{W}(k)-\mathbf{I}_r\|^2_{\text{Fro}}$ from a non-orthogonal matrix $\mathbf{W}(0)$) in the FRANS and HFRANS algorithms.

Another solution to orthonormalize the columns of $\mathbf{W}'(k+1)$ has been proposed in [28, 29]. It consists of two steps. The first one orthogonalizes these columns using a matrix $\mathbf{G}(k+1)$ to give $\mathbf{W}''(k+1) = \mathbf{W}'(k+1)\mathbf{G}(k+1)$, and the second one normalizes the columns of $\mathbf{W}''(k+1)$. To find such a matrix $\mathbf{G}(k+1)$ which is of course not unique, notice that if $\mathbf{G}(k+1)$ is an orthogonal matrix having as first column, the vector $\frac{\mathbf{y}(k)}{\|\mathbf{y}(k)\|_2}$ with the remaining $r-1$ columns completing an orthonormal basis, then using (42.5.8), the product $\mathbf{W}''^T(k+1)\mathbf{W}''(k+1)$ becomes the following diagonal matrix

$$\begin{aligned}\mathbf{W}''^T(k+1)\mathbf{W}''(k+1) &= \mathbf{G}^T(k+1)\left(\mathbf{I}_r + \delta_k\mathbf{y}(k)\mathbf{y}^T(k)\right)\mathbf{G}(k+1)\\ &= \mathbf{I}_r + \delta_k\|\mathbf{y}(k)\|^2\mathbf{e}_1\mathbf{e}_1^T.\end{aligned}$$

where $\delta_k \stackrel{\text{def}}{=} \pm 2\mu_k + \mu_k^2\|\mathbf{x}(k)\|^2$ and $\mathbf{e}_1 \stackrel{\text{def}}{=} [0,...,0]^T$. It is fortunate that there exists such an orthonogonal matrix $\mathbf{G}(k+1)$ with the desired properties known as a Householder reflector [34, Chap.5], and can be very easily generated since it is of the form

$$\mathbf{G}(k+1) = \mathbf{I}_r - \frac{2}{\|\mathbf{a}(k)\|^2}\mathbf{a}(k)\mathbf{a}^T(k) \quad \text{with} \quad \mathbf{a}(k) = \mathbf{y}(k) - \|\mathbf{y}(k)\|\mathbf{e}_1. \tag{42.5.12}$$

This gives the *Fast Data Projection Method* (FDPM)

$$\mathbf{W}(k+1) = \text{Normalize}\{\left(\mathbf{W}(k) \pm \mu_k\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\right)\mathbf{G}(k+1)\}, \tag{42.5.13}$$

where "Normalize$\{\mathbf{W}$"(k+1)$\}$" stands for normalization of the columns of $\mathbf{W}''(k+1)$, and $\mathbf{G}(k+1)$ is the Householder transform given by (42.5.12). Using the independence assumption [35, chap. 9.4] and the approximation $\mu_k \ll 1$, a simplistic theoretical analysis has been presented in [30] for both signal and noise subspace tracking. It shows that the FDPM algorithm is locally stable and the distance to orthonormality $\mathrm{E}\left(\|\mathbf{W}^T(k)\mathbf{W}(k) - \mathbf{I}_r\|^2\right)$

[6] However, if one looks very carefully at the simulation graphs representing the orthonormality error [74, Fig. 7], it is easy to realize that the HFRANS algorithm exhibits a slight linear instability.

tends to zero as $O(e^{-ck})$ where $c > 0$ does not depend on μ. Furthermore, numerical simulations presented in [28, 29, 30] with $\mu_k = \frac{\mu}{\|\mathbf{x}(k)\|^2}$ demonstrate that this algorithm is numerically stable for both signal and noise subspace tracking, and if for some reason, orthonormality is lost, or the algorithm is initialized with a matrix that is not orthonormal, the algorithm exhibits an extremely high convergence speed to an orthonormal matrix. This FDPM algorithm is to the best to our knowledge, the only power-based minor subspace tracking methods of complexity $O(nr)$ that is truly numerically stable since it do not accumulate rounding errors.

42.5.1.3 Power-based methods issued from exponential or sliding window

Of course, all the above algorithms that do not use the rank one property of the instantaneous estimate $\mathbf{x}(k)\mathbf{x}^T(k)$ of $\mathbf{C}_x(k)$ can be extended to the exponential (42.3.5) or sliding windowed (42.3.8) estimates $\mathbf{C}(k)$, but with an important increase in complexity. To keep the $O(nr)$ complexity, the orthogonal iteration method (42.2.12) must be adapted to the following iterations

$$\begin{aligned}\mathbf{W}'(k+1) &= \mathbf{C}(k)\mathbf{W}(k)\\ \mathbf{W}(k+1) &= \text{Orthonorm}\{\mathbf{W}'(k+1)\}\\ &= \mathbf{W}'(k+1)\mathbf{G}(k+1),\end{aligned}$$

where the matrix $\mathbf{G}(k+1)$ is a square root inverse of $\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)$ responsable for performing orthonormalization of $\mathbf{W}'(k+1)$. It is the choice of $\mathbf{G}(k+1)$ that will pave the way to different adaptive algorithms.

Based on the approximation

$$\mathbf{C}(k-1)\mathbf{W}(k) = \mathbf{C}(k-1)\mathbf{W}(k-1), \tag{42.5.14}$$

which is clearly valid if $\mathbf{W}(k)$ is slowly varying with k, an adaptation of the power method denoted *Natural Power method 3* (NP3) has been proposed in [37] for the exponential windowed estimate (42.3.5) $\mathbf{C}(k) = \beta\mathbf{C}(k-1) + \mathbf{x}(k)\mathbf{x}^T(k)$. Using (42.3.5) and (42.5.14), we obtain

$$\mathbf{W}'(k+1) = \beta\mathbf{W}'(k) + \mathbf{x}(k)\mathbf{y}^T(k),$$

with $\mathbf{y}(k) \stackrel{\text{def}}{=} \mathbf{W}^T(k)\mathbf{x}(k)$. It then follows that

$$\begin{aligned}\mathbf{W}'^T(k+1)\mathbf{W}'(k+1) &= \beta^2\mathbf{W}'^T(k)\mathbf{W}'(k) + \mathbf{z}(k)\mathbf{y}^T(k) + \mathbf{y}(k)\mathbf{z}^T(k)\\ &\quad + \|\mathbf{x}(k)\|^2\mathbf{y}(k)\mathbf{y}^T(k)\end{aligned} \tag{42.5.15}$$

with $\mathbf{z}(k) \stackrel{\text{def}}{=} \beta\mathbf{W}'^T(k)\mathbf{x}(k)$, which implies (see Exercice 42.9) the following recursions

$$\mathbf{G}(k+1) = \frac{1}{\beta}[\mathbf{I}_n - \tau_1\mathbf{e}_1\mathbf{e}_1^T - \tau_2\mathbf{e}_2\mathbf{e}_2^T]\mathbf{G}(k), \tag{42.5.16}$$

$$\begin{aligned}\mathbf{W}(k+1) &= \mathbf{W}(k)[\mathbf{I}_n - \tau_1\mathbf{e}_1\mathbf{e}_1^T - \tau_2\mathbf{e}_2\mathbf{e}_2^T]\\ &+ \frac{1}{\beta}\mathbf{x}(k)\mathbf{y}^T(k)\mathbf{G}^T(k)[\mathbf{I}_n - \tau_1\mathbf{e}_1\mathbf{e}_1^T - \tau_2\mathbf{e}_2\mathbf{e}_2^T],\end{aligned} \tag{42.5.17}$$

where τ_1, τ_2 and $\mathbf{e}_1, \mathbf{e}_2$ are defined in Exercice 42.9.

Note that the square root inverse matrix $\mathbf{G}(k+1)$ of $\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)$ is asymmetric even if $\mathbf{G}(0)$ is symmetric. Expressions (42.5.16) and (42.5.17) provide an algorithm which does not involve any matrix-matrix multiplications and in fact requires only $O(nr)$ operations.

Based on the approximation that $\mathbf{W}(k)$ and $\mathbf{W}(k+1)$ span the same r-dimensional subspace, another power-based algorithm referred to as the *Approximated Power Iteration* (API) algorithm and its fast implementation (FAPI) have been proposed in [8]. Compared to the NP3 algorithm, this scheme has the advantage that it can handle the exponential (42.3.5) or the sliding windowed (42.3.8) estimates of $\mathbf{C}_x(k)$ in the same framework (and with the same complexity of $O(nr)$ operations) by writing (42.3.5) and (42.3.8) in the form

$$\mathbf{C}(k) = \beta\mathbf{C}(k-1) + \mathbf{x}'(k)\mathbf{J}\mathbf{x}'^T(k)$$

with $\mathbf{J} = 1$ and $\mathbf{x}'(k) = \mathbf{x}(k)$ for the exponential window and $\mathbf{J} = \begin{bmatrix} 1 & 0 \\ 0 & -\beta^l \end{bmatrix}$ and $\mathbf{x}'(k) = [\mathbf{x}(k), \mathbf{x}(k-l)]$ for the sliding window (see (42.3.8)). Among the power-based minor subspace tracking methods issued from exponential of sliding window, this FAPI algorithm has been considered by many practitioners (e.g., [11]) as outperforming the other algorithms having the same computational complexity.

42.5.2 Projection approximation-based methods

Since (42.2.14) describes an unconstrained cost function to be minimized, it is straightforward to apply the gradient-descent technique for dominant subspace tracking. Using expression (42.9.4) of the gradient given in Exercice 42.7 with the estimate $\mathbf{x}(k)\mathbf{x}^T(k)$ of $\mathbf{C}_x(k)$ gives:

$$\begin{aligned} \mathbf{W}(k+1) &= \mathbf{W}(k) - \mu_k \left[-2\mathbf{x}(k)\mathbf{x}^T(k) + \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{W}^T(k)\right. \\ &\qquad \left. + \mathbf{W}(k)\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\right]\mathbf{W}(k). \quad (42.5.18) \end{aligned}$$

We note that this algorithm can be linked to Oja's algorithm (42.5.5). First, the term between brackets is the symmetrization of the term $-\mathbf{x}(k)\mathbf{x}^T(k) + \mathbf{W}(k)\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)$ of Oja's algorithm (42.5.5). Second, we see that when $\mathbf{W}^T(k)\mathbf{W}(k)$ is approximated by $\mathbf{I}_r$ (which is justified from the stability property below), algorithm (42.5.18) gives Oja's algorithm (42.5.5). We note that because the field of the stochastic approximation algorithm (42.5.18) is the opposite of the derivative of the positive function (42.2.14), the orthonormal bases of the dominant subspace are globally asymptotically stable for its associated ODE (see Subsection 42.7.1) in contrast to Oja's algorithm (42.5.5), for which they are only locally asymptotically stable. A complete performance analysis of the stochastic approximation algorithm (42.5.18) has been presented in [24] where closed-form expressions of the asymptotic covariance of the estimated projection matrix $\mathbf{W}(k)\mathbf{W}^T(k)$ are given and commented on for independent Gaussian data $\mathbf{x}(k)$ and constant step size μ.

If now $\mathbf{C}_x(k)$ is estimated by the exponentially weighted sample covariance matrix $\mathbf{C}(k) = \sum_{i=0}^k \beta^{k-i}\mathbf{x}(i)\mathbf{x}^T(i)$ (42.3.4) instead of $\mathbf{x}(k)\mathbf{x}^T(k)$, the scalar function $J(\mathbf{W})$ becomes

$$J(\mathbf{W}) = \sum_{i=0}^{k} \beta^{k-i}\|\mathbf{x}(i) - \mathbf{W}\mathbf{W}^T\mathbf{x}(i)\|^2, \quad (42.5.19)$$

and all data $\mathbf{x}(i)$ available in the time interval $\{0, ..., k\}$ are involved in estimating the dominant subspace at time instant $k+1$ supposing this estimate known at time instant k. The key issue of the projection approximation subspace tracking algorithm (PAST) proposed by Yang in [70] is to approximate $\mathbf{W}^T(k)\mathbf{x}(i)$ in (42.5.19), the unknown projection of $\mathbf{x}(i)$ onto the columns of $\mathbf{W}(k)$ by the expression $y(i) = \mathbf{W}^T(i)\mathbf{x}(i)$ which can be calculated for all $0 \leq i \leq k$ at the time instant k. This results in the following modified cost function

$$J'(\mathbf{W}) = \sum_{i=0}^{k} \beta^{k-i} \|\mathbf{x}(i) - \mathbf{W}\mathbf{y}(i)\|^2, \tag{42.5.20}$$

which is now quadratic in the elements of $\mathbf{W}$. This projection approximation, hence the name PAST, changes the error performance surface of $J(\mathbf{W})$. For stationary or slowly varying $\mathbf{C}_x(k)$, the difference between $\mathbf{W}^T(k)\mathbf{x}(i)$ and $\mathbf{W}^T(i)\mathbf{x}(i)$ is small, in particular when i is close to k. However, this difference may be larger in the distant past with $i \ll k$, but the contribution of the past data to the cost function (42.5.20) is decreasing for growing k, due to the exponential windowing. It is therefore expected that $J'(\mathbf{W})$ will be a good approximation to $J(\mathbf{W})$ and the matrix $\mathbf{W}(k)$ minimizing $J'(\mathbf{W})$ be a good estimate for the dominant subspace of $\mathbf{C}_x(k)$. In case of sudden parameter changes of the model (42.3.1), the numerical experiments presented in [70] show that the algorithms derived from this PAST approach still converge. The main advantage of this scheme is that the least square minimization of (42.5.20) whose solution is given by $\mathbf{W}(k+1) = \mathbf{C}_{x,y}(k)\mathbf{C}_y^{-1}(k)$ where $\mathbf{C}_{x,y}(k) \stackrel{\text{def}}{=} \sum_{i=0}^{k} \beta^{k-i}\mathbf{x}(i)\mathbf{y}^T(i)$ and $\mathbf{C}_y(k) \stackrel{\text{def}}{=} \sum_{i=0}^{k} \beta^{k-i}\mathbf{y}(i)\mathbf{y}^T(i)$ has been extensively studied in adaptive filtering (see e.g., [35, chap. 13] and [67, chap. 12]) where various *Recursive Least Square* algorithms (RLS) based on the matrix inversion lemma have been proposed[7] We note that because of the approximation of $J(\mathbf{W})$ by $J'(\mathbf{W})$, the columns of $\mathbf{W}(k)$ are not exactly orthonormal. But this lack of orthonormality does not mean that we need to perform a reorthonormalization of $\mathbf{W}(k)$ after each update. For this algorithm, the necessity of orthonormalization depends solely on the post processing method which uses this signal subspace estimate to extract the desired signal information (see e.g., Section 42.8). It is shown in the numerical experiments presented in [70] that the deviation of $\mathbf{W}(k)$ from orthonormality is very "small" and for a growing sliding window ($\beta = 1$), $\mathbf{W}(k)$ converges to a matrix with exactly orthonormal columns under signal stationary. Finally, note that a theoretical study of convergence and a derivation of the asymptotic distribution of the recursive subspace estimators have been presented in [72] and [73] respectively. Using the ODE associated with this algorithm (see Section 42.7.1) which is here a pair of coupled matrix differential equations, it is proved that under signal stationarity and other weak conditions, the PAST algorithm converges to the desired signal subspace with probability one.

To speed up the convergence of the PAST algorithm and to guarantee the orthonormality of $\mathbf{W}(k)$ at each iteration, an orthonormal version of the PAST algorithm dubbed OPAST has been proposed in [1]. This algorithm consists of the PAST algorithm where $\mathbf{W}(k+1)$ is related to $\mathbf{W}(k)$ by $\mathbf{W}(k+1) = \mathbf{W}(k) + \mathbf{p}(k)\mathbf{q}(k)$, plus an orthonormalization step of $\mathbf{W}(k)$ based on the same approach as those used in the FRANS algorithm (see Subsection 42.5.1.2) which leads to the update $\mathbf{W}(k+1) = \mathbf{W}(k) + \mathbf{p}'(k)\mathbf{q}(k)$.

Note that the PAST algorithm cannot be used to estimate the noise subspace by simply changing the sign of the step size because the associated ODE is unstable. Efforts to eliminate this instability were attempted in [4] by forcing the orthonormality of $\mathbf{W}(k)$ at

[7] For possible sudden signal parameter changes (see Subsection 42.3.1), the use of a sliding exponential window (42.3.7) version of the cost function may offer faster convergence. In this case, $\mathbf{W}(k)$ can be calculated recursively as well [70] by applying the general form of the matrix inversion lemma $(\mathbf{A} + \mathbf{B}\mathbf{D}\mathbf{C}^T)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{D}^{-1} + \mathbf{C}^T\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}^T\mathbf{A}^{-1}$ which requires inversion of a 2×2 matrix.

each time step. Although there was a definite improvement in the stability characteristics, the resulting algorithm remains numerically unstable.

42.5.3 Additional methodologies

Various generalizations of criteria (42.2.7) and (42.2.14) have been proposed (e.g., in [40]), which generally yield robust estimates of principal subspaces or eigenvectors that are totally different from the standard ones. Among them, the following *Novel Information Criterion* (NIC) [47] results in a fast algorithm to estimate the principal subspace with a number of attractive properties

$$\max_{\mathbf{W}}\{J(\mathbf{W})\} \text{ with } J(\mathbf{W}) \overset{\text{def}}{=} \text{Tr}[\ln(\mathbf{W}^T\mathbf{C}\mathbf{W})] - \text{Tr}(\mathbf{W}^T\mathbf{W}), \tag{42.5.21}$$

given that $\mathbf{W}$ lies in the domain $\{\mathbf{W}$ such that $\mathbf{W}^T\mathbf{C}\mathbf{W} > 0\}$, where the matrix logarithm is defined e.g. in [34, chap. 11]. It is proved in [47] (see also Exercices 42.11 and 42.12) that the above criterion has a global maximum that is attained when and only when $\mathbf{W} = \mathbf{U}_r\mathbf{Q}$ where $\mathbf{U}_r = [\mathbf{u}_1, ..., \mathbf{u}_r]$ and $\mathbf{Q}$ is an arbitrary $r \times r$ orthogonal matrix and all the other stationary points are saddle points. Taking the gradient of (42.5.21) (which is given explicitly by (42.9.6)), the following gradient ascent algorithm has been proposed in [47] for updating the estimate $\mathbf{W}(k)$:

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \mu_k\left[\mathbf{C}(k)\mathbf{W}(k)(\mathbf{W}^T(k)\mathbf{C}(k)\mathbf{W}(k))^{-1} - \mathbf{W}(k)\right]. \tag{42.5.22}$$

Using the recursive estimate $\mathbf{C}(k) = \sum_{i=0}^{k}\beta^{k-i}\mathbf{x}(i)\mathbf{x}^T(i)$ (42.3.4), and the projection approximation introduced in [70] $\mathbf{W}^T(k)\mathbf{x}(i) = \mathbf{W}^T(i)\mathbf{x}(i)$ for all $0 \leq i \leq k$, the update (42.5.22) becomes

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \mu_k\left[\left(\sum_{i=0}^{k}\beta^{k-i}\mathbf{x}(i)\mathbf{y}^T(i)\right)\left(\sum_{i=0}^{k}\beta^{k-i}\mathbf{y}(i)\mathbf{y}^T(i)\right)^{-1} - \mathbf{W}(k)\right], \tag{42.5.23}$$

with $\mathbf{y}(i) \overset{\text{def}}{=} \mathbf{W}^T(i)\mathbf{x}(i)$. Consequently, similarly to the PAST algorithms, standard RLS techniques used in adaptive filtering can be applied. According to the numerical experiments presented in [37], this algorithm performs very similarly to the PAST algorithm having also the same complexity. Finally, we note that it has been proved in [47] that the points $\mathbf{W} = \mathbf{U}_r\mathbf{Q}$ are the only asymptotically stable points of the ODE (see Subsection 42.7.1) associated with the gradient ascent algorithm (42.5.22) and that the attraction set of these points is the domain $\{\mathbf{W}$ such that $\mathbf{W}^T\mathbf{C}\mathbf{W} > 0\}$. But to the best of our knowledge, no complete theoretical performance analysis of algorithm (42.5.23) has been carried out so far.

42.6 EIGENVECTORS TRACKING

Although, the adaptive estimation of the dominant or minor subspace through the estimate $\mathbf{W}(k)\mathbf{W}^T(k)$ of the associated projector is of most importance for subspace-based algorithms, there are situations where the associated eigenvalues are simple ($\lambda_1 > ... > \lambda_r > \lambda_{r+1}$ or $\lambda_n < ... < \lambda_{n-r+1} < \lambda_{n-r}$) and the desired estimated orthonormal basis of this space must form an eigenbasis. This is the case for the statistical technique of principal component analysis in data compression and coding, optimal feature extraction in pattern recognition and for optimal fitting in the total least square sense or for Karhunen-Loève transformation of signals, to mention only a few examples. In these applications, $\{y_1(k),...,y_r(k)\}$ or $\{y_n(k),...,y_{n-r+1}(k)\}$ with $y_i(k) \stackrel{\text{def}}{=} \mathbf{w}_i^T(k)\mathbf{x}(k)$ where $\mathbf{W} = [\mathbf{w}_1(k),...,\mathbf{w}_r(k)]$ or $\mathbf{W} = [\mathbf{w}_n(k),...,\mathbf{w}_{n-r+1}(k)]$ are the estimated r first *principal* or r last *minor components* of the data $\mathbf{x}(k)$. To derive such adaptive estimates, the stochastic approximation algorithms that have been proposed, are issued from adaptations of the iterative constrained maximizations (42.2.5) and minimizations (42.2.6) of Rayleigh quotients; the weighted subspace criterion (42.2.8); the orthogonal iterations (42.2.11) and, finally the gradient-descent technique applied to the minimization of (42.2.14).

42.6.1 Rayleigh quotient-based methods

To adapt maximization (42.2.5) and minimization (42.2.6) of Rayleigh quotients to adaptive implementations, a method has been proposed in [60]. It is derived from a Givens parametrization of the constraint $\mathbf{W}^T\mathbf{W} = \mathbf{I}_r$, and from a gradient-like procedure. The Givens rotations approach introduced by Regalia [60] is based on the properties that any $n \times 1$ unit 2-norm vector and any orthogonal vector to this vector can be respectively written as the last column of an $n \times n$ orthogonal matrix and as a linear combinaison of the first $n-1$ columns of this orthogonal matrix, i.e.,

$$\mathbf{w}_1 = \mathbf{Q}_1 \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}, \mathbf{w}_2 = \mathbf{Q}_1 \begin{bmatrix} \mathbf{Q}_2 \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} \\ 0 \end{bmatrix}, \ldots, \mathbf{w}_r = \mathbf{Q}_1 \begin{bmatrix} \mathbf{Q}_2 \begin{bmatrix} \mathbf{Q}_r \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} \\ \\ 0 \end{bmatrix} \\ 0 \end{bmatrix}$$

where $\mathbf{Q}_i$ is the following orthogonal matrix of order $n-i+1$:

$$\mathbf{Q}_i = \mathbf{U}_{i,1} \ldots \mathbf{U}_{i,j} \ldots \mathbf{U}_{i,n-i} \quad \text{with} \quad \mathbf{U}_{i,j} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{I}_{j-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\sin\theta_{i,j} & \cos\theta_{i,j} & \mathbf{0} \\ \mathbf{0} & \cos\theta_{i,j} & \sin\theta_{i,j} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{n-i-j} \end{bmatrix}$$

and $\theta_{i,j}$ belongs to $]-\frac{\pi}{2}, +\frac{\pi}{2}]$. The existence of such a parametrization[8] for all orthonormal sets $\{\mathbf{w}_1, \ldots, \mathbf{w}_r\}$ is proved in [60]. It consists of $r(2n-r-1)/2$ real parameters. Furthermore, this parametrization is unique if we add some constraints on $\theta_{i,j}$. A deflation procedure, inspired by the maximization (42.2.5) and minimization (42.2.6) has been proposed [60]. First maximization or minimization (42.2.3) is performed with the help of the classical stochastic gradient algorithm, in which the parameters are $\theta_{1,1}, \ldots, \theta_{1,n-1}$, whereas maximization (42.2.5) or minimization (42.2.6) are realized thanks to stochastic gradient algorithms with respect to the parameters $\theta_{i,1}, \ldots, \theta_{i,n-i}$, in which the preceding parameters $\theta_{l,1}(k), \ldots, \theta_{l,n-l}(k)$ for $l = 1, \ldots, i-1$ are injected from the $i-1$ previous algorithms. The deflation procedure is achieved by coupled stochastic gradient algorithms

$$\begin{bmatrix} \boldsymbol{\theta}_1(k+1) \\ \cdot \\ \boldsymbol{\theta}_r(k+1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\theta}_1(k) \\ \cdot \\ \boldsymbol{\theta}_r(k) \end{bmatrix} \pm \mu_k \begin{bmatrix} f_1(\boldsymbol{\theta}_1(k), \mathbf{x}(k)) \\ \cdot \\ f_r(\boldsymbol{\theta}_1(k), \ldots, \boldsymbol{\theta}_r(k), \mathbf{x}(k)) \end{bmatrix} \qquad (42.6.1)$$

[8] Note that this parametrization extends immediately to the complex case using the kernel $\begin{bmatrix} -\sin\theta_{i,j} & \cos\theta_{i,j} \\ e^{i\phi_{i,j}}\cos\theta_{i,j} & e^{i\phi_{i,j}}\sin\theta_{i,j} \end{bmatrix}$.

with $\boldsymbol{\theta}_i \stackrel{\text{def}}{=} [\theta_{i,1},\ldots,\theta_{i,n-i}]^T$ and $f_i(\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_i,\mathbf{x}) \stackrel{\text{def}}{=} \boldsymbol{\nabla}_{\boldsymbol{\theta}_i}(\mathbf{w}_i^T\mathbf{x}\mathbf{x}^T\mathbf{w}_i) = 2\boldsymbol{\nabla}_{\boldsymbol{\theta}_i}(\mathbf{w}_i^T)\mathbf{x}$ $\mathbf{x}^T\mathbf{w}_i$, $i=1,\ldots,r$. This rather intuitive computational process was confirmed by simulation results [60]. Later a formal analysis of the convergence and performance had been performed in [23] where it has been proved that the stationary points of the associated ODE are globally asymptotically stable (see Subsection 42.7.1) and that the stochastic algorithm (42.6.1) converges almost surely to these points for stationary data $\mathbf{x}(k)$ when μ_k is decreasing with $\lim_{k\to\infty}\mu_k = 0$ and $\sum_k \mu_k = \infty$. We note that this algorithm yields exactly orthonormal r dominant or minor estimated eigenvectors by a simple change of sign in its step size, and requires $O(nr)$ operations at each iteration but without accounting for the trigonometric functions.

Alternatively, a stochastic gradient-like algorithm denoted *Direct Adaptive Subspace Estimation* (DASE) has been proposed in [61] with a direct parametrization of the eigenvectors by means of their coefficients. Maximization or minimization (42.2.3) is performed with the help of a modification of the classical stochastic gradient algorithm to assure an approximate unit norm of the first estimated eigenvector $\mathbf{w}_1(k)$ (in fact a rewriting of Oja's neuron (42.4.1)). Then, a modification of the classical stochastic gradient algorithm using a deflation procedure, inspired by the constraint $\mathbf{W}^T\mathbf{W} = \mathbf{I}_r$ gives the estimates $(\mathbf{w}_i(k))_{i=2,\ldots,r}$

$$\begin{aligned}
\mathbf{w}_1(k+1) &= \mathbf{w}_1(k) \pm \mu_k \left[\mathbf{x}(k)\mathbf{x}^T(k) - (\mathbf{w}_1^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}_1(k))\mathbf{I}_n\right]\mathbf{w}_1(k) \\
\mathbf{w}_i(k+1) &= \mathbf{w}_i(k) \pm \mu_k \Big[\mathbf{x}(k)\mathbf{x}^T(k) - \big(\mathbf{w}_i^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}_i(k)\big) \\
&\qquad \left(\mathbf{I}_n - \sum_{j=1}^{i-1}\mathbf{w}_j(k)\mathbf{w}_j^T(k)\right)\Big]\mathbf{w}_i(k) \quad \text{for } i=2,\ldots,r. \qquad (42.6.2)
\end{aligned}$$

This totally empirical procedure has been studied in [62]. It has been proved that the stationary points of the associated ODE are all eigenvector bases $\{\pm\mathbf{u}_{i_1},...,\pm\mathbf{u}_{i_r}\}$. Using the eigenvalues of the derivative of the mean field (see Subsection 42.7.1), it is shown that all these eigenvector bases are unstable except $\{\pm\mathbf{u}_1\}$ for $r=1$ associated with the sign "+" (where algorithm (42.6.2) is Oja's neuron (42.4.1)). But a close examination of these eigenvalues that are all real-valued, shows that for only the eigenbasis $\{\pm\mathbf{u}_1,...,\pm\mathbf{u}_r\}$ and $\{\pm\mathbf{u}_n,...,\pm\mathbf{u}_{n-r+1}\}$ associated with the sign "+" and "-" respectively, all the eigenvalues of the derivative of the mean field are strictly negative except for the eigenvalues associated with variations of the eigenvectors $\{\pm\mathbf{u}_1,...,\pm\mathbf{u}_r\}$ and $\{\pm\mathbf{u}_n,...,\pm\mathbf{u}_{n-r+1}\}$ in their directions. Consequently, it is claimed in [62] that if the norm of each estimated eigenvector is set to one at each iteration, the stability of the algorithm is ensured. The simulations presented in [61] confirm this intuition.

42.6.2 Eigenvector power-based methods

Note that similarly to the subspace criterion (42.2.7), the maximization or minimization of the weighted subspace criterion (42.2.8) $J(\mathbf{W}) \stackrel{\text{def}}{=} \mathrm{Tr}(\boldsymbol{\Omega}\mathbf{W}^T\mathbf{C}(k)\mathbf{W})$ subject to the constraint $\mathbf{W}^T\mathbf{W} = \mathbf{I}_r$ can be solved by a constrained gradient-descent technique. Clearly, the simplest selection for $\mathbf{C}(k)$ is the instantaneous estimate $\mathbf{x}(k)\mathbf{x}^T(k)$. Because in this case, $\boldsymbol{\nabla}_{\mathbf{W}}J = 2\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}\boldsymbol{\Omega}$, we obtain the following stochastic approximation algorithm that will be a starting point for a family of algorithms that have been derived to adaptively estimate majorant or minor eigenvectors

$$\mathbf{W}(k+1) = \{\mathbf{W}(k) \pm \mu_k\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\boldsymbol{\Omega}\}\mathbf{G}(k+1), \qquad (42.6.3)$$

in which $\mathbf{W}(k) = [\mathbf{w}_1(k), \ldots, \mathbf{w}_r(k)]$ and the matrix $\mathbf{\Omega}$ is a diagonal matrix $\mathrm{Diag}(\omega_1, ..., \omega_r)$ with $\omega_1 > ... > \omega_r > 0$. $\mathbf{G}(k+1)$ is a matrix depending on

$$\mathbf{W}'(k+1) \overset{\text{def}}{=} \mathbf{W}(k) \pm \mu_k \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{\Omega},$$

which orthonormalizes or approximately orthonormalizes the columns of $\mathbf{W}'(k+1)$. Thus, $\mathbf{W}(k)$ has orthonormal or approximately orthonormal columns for all k. Depending on the form of matrix $\mathbf{G}(k+1)$, variants of the basic stochastic algorithm are obtained. Going back to the general expression (42.5.4) of the subspace power-based algorithm, we note that (42.6.3) can also be derived from (42.5.4), where different step sizes $\mu_k\omega_1, ..., \mu_k\omega_r$ are introduced for each column of $\mathbf{W}(k)$.

Using the same approach as for deriving (42.5.5), i.e., where $\mathbf{G}(k+1)$ is the symmetric square root inverse of ${\mathbf{W}'}^T(k+1)\mathbf{W}'(k+1)$, we obtain the following stochastic approximation algorithm

$$\begin{aligned} \mathbf{W}(k+1) \quad = \quad & \mathbf{W}(k) \pm \mu_k[\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{\Omega} - \frac{1}{2}\mathbf{W}(k)\mathbf{\Omega}\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k) \\ - \quad & \frac{1}{2}\mathbf{W}(k)\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{\Omega}]. \end{aligned} \tag{42.6.4}$$

Note that in contrast to the Oja's algorithm (42.5.5), this algorithm is different from the algorithm issued from the optimization of the cost function $J(\mathbf{W}) \overset{\text{def}}{=} \mathrm{Tr}[\mathbf{\Omega}\mathbf{W}^T\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}]$ defined on the set of $n \times r$ orthogonal matrices $\mathbf{W}$ with the help of continuous-time matrix algorithms (see e.g., [21, Ch. 7.2], [19, Ch. 4] or (42.9.7) in Exercice 42.15)).

$$\mathbf{W}(k+1) = \mathbf{W}(k) \pm \mu_k \left[\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{\Omega} - \mathbf{W}(k)\mathbf{\Omega}\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\right]. \tag{42.6.5}$$

We note that these two algorithms reduce to the Oja's algorithm (42.5.5) for $\mathbf{\Omega} = \mathbf{I}_r$ and to Oja's neuron (42.4.1) for $r = 1$, which of course is unstable for tracking the minorant eigenvectors with the sign "-". But to the best of our knowledge, no complete theoretical performance analysis of these two algorithms has been carried out until now. Techniques used for stabilizing Oja's algorithm (42.5.5) for minor subspace tracking, has been transposed to stabilize the weighted Oja's algorithm for tracking the minorant eigenvectors. For example, in [9], $\mathbf{W}(k)$ is forced to be orthonormal at each time step k as in [2] (see Exercice 42.10) with the *MCA-OOja algorithm* and the *MCA-OOjaH algorithm* using Householder transforms. Note, that by proving a recursion of the distance to orthonormality $\|\mathbf{W}^T(k)\mathbf{W}(k) - \mathbf{I}_r\|^2_{\mathrm{Fro}}$ from a non-orthogonal matrix $\mathbf{W}(0)$, it has been shown in [10], that the latter algorithm is numerically stable in contrast to the former.

Instead of deriving a stochastic approximation algorithm from a specific orthonormalization matrix $\mathbf{G}(k+1)$, an analogy with Oja's algorithm (42.5.5) has been used in [53] to derive the following algorithm

$$\mathbf{W}(k+1) = \mathbf{W}(k) \pm \mu_k \left[\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k) - \mathbf{W}(k)\mathbf{\Omega}\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{\Omega}^{-1}\right]. \tag{42.6.6}$$

It has been proved in [54], that for tracking the dominant eigenvectors (i.e., with the sign "+"), the eigenvectors $\{\pm\mathbf{u}_1, ..., \pm\mathbf{u}_r\}$ are the only locally asymptotically stable points of the ODE associated with (42.6.6).

If now the matrix $\mathbf{G}(k+1)$ performs the Gram-Schmidt orthonormalization on the columns of $\mathbf{W}'(k+1)$, an algorithm, denoted *Stochastic Gradient Ascent* (SGA) algorithm, is obtained if the successive columns of matrix $\mathbf{W}(k+1)$ are expanded, assuming μ_k

sufficiently small. By omitting the $O(\mu_k^2)$ term in this expansion, we obtain [50] the following algorithm

$$\begin{aligned}\mathbf{w}_i(k+1) &= \mathbf{w}_i(k) + \alpha_i\mu_k \left[\mathbf{I}_n - \mathbf{w}_i(k)\mathbf{w}_i^T(k) - \sum_{j=1}^{i-1}(1+\frac{\alpha_j}{\alpha_i})\mathbf{w}_j(k)\mathbf{w}_j^T(k)\right] \\ &\qquad \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}_i(k) \quad \text{for } i=1,\ldots,r. \quad (42.6.7)\end{aligned}$$

where here $\boldsymbol{\Omega} = \text{Diag}(\alpha_1, \alpha_2, \ldots, \alpha_r)$ with α_i arbitrary strictly positive numbers.

The so called *Generalized Hebbian Algorithm* (GHA) is derived from Oja's algorithm (42.5.5) by replacing the matrix $\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)$ of Oja's algorithm by its diagonal and superdiagonal only:

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \mu_k[\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k) - \mathbf{W}(k)\text{upper}(\mathbf{W}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)]$$

in which the operator "upper" sets all subdiagonal elements of a matrix to zero. When written columnwise, this algorithm is similar to the SGA algorithm (42.6.7) where $\alpha_i = 1$, $i = 1, .., r$, with the difference that there is no coefficient 2 in the sum:

$$\mathbf{w}_i(k+1) = \mathbf{w}_i(i) + \mu_k \left[\mathbf{I}_n - \sum_{j=1}^{i}\mathbf{w}_j(k)\mathbf{w}_j^T(k)\right]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}_i(k) \quad \text{for } i=1,\ldots,r. \quad (42.6.8)$$

Oja *et al* [53] proposed an algorithm denoted *Weighted Subspace Algorithm* (WSA), which is similar to the Oja's algorithm, except for the scalar parameters $\beta_1, \ldots, \beta_r$:

$$\mathbf{w}_i(k+1) = \mathbf{w}_i(k) + \mu_k \left[\mathbf{I}_n - \sum_{j=1}^{r}\frac{\beta_j}{\beta_i}\mathbf{w}_j(k)\mathbf{w}_j^T(k)\right]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}_i(k) \quad \text{for } i=1,\ldots,r, \quad (42.6.9)$$

with $\beta_1 > \ldots > \beta_r > 0$. If $\beta_i = 1$ for all i, this algorithm reduces to Oja's algorithm.

Following the deflation technique introduced in the *Adaptive Principal Component Extraction* (APEX) algorithm [41], note finally that Oja's neuron can be directly adapted to estimate the r principal eigenvectors by replacing the instantaneous estimate $\mathbf{x}(k)\mathbf{x}^T(k)$ of $\mathbf{C}_x(k)$ by $\mathbf{x}(k)\mathbf{x}^T(k)[\mathbf{I}_n - \sum_{j=1}^{i-1}\mathbf{w}_j(k)\mathbf{w}_j^T(k)]$ to successively estimate $\mathbf{w}_i(k)$, $i = 2, ..., r$

$$\begin{aligned}\mathbf{w}_i(k+1) &= \mathbf{w}_i(i) + \mu_k\left[\mathbf{I}_n - \mathbf{w}_i(k)\mathbf{w}_i^T(k)\right]\mathbf{x}(k)\mathbf{x}^T(k)\left[\mathbf{I}_n - \sum_{j=1}^{i-1}\mathbf{w}_j(k)\mathbf{w}_j^T(k)\right] \\ &\qquad \mathbf{w}_i(k) \quad \text{for } i=1,\ldots,r.\end{aligned}$$

Minor component analysis was also considered in neural networks to solve the problem of optimal fitting in the total least square sense. Xu *et al.* [78] introduced the *Optimal Fitting Analyzer* (OFA) algorithm by modifying the SGA algorithm. For the estimate $\mathbf{w}_n(k)$ of the eigenvector associated with the smallest eigenvalue, this algorithm is derived from the Oja's Neuron (42.4.1) by replacing $\mathbf{x}(k)\mathbf{x}^T(k)$ by $\mathbf{I}_n - \mathbf{x}(k)\mathbf{x}^T(k)$, viz

$$\mathbf{w}_n(k+1) = \mathbf{w}_n(k) + \mu[\mathbf{I}_n - \mathbf{w}_n(k)\mathbf{w}_n^T(k)][\mathbf{I}_n - \mathbf{x}(k)\mathbf{x}^T(k)]\mathbf{w}_n(k),$$

and for $i = n, \ldots, n-r+1$, his algorithm reads

$$\begin{aligned}\mathbf{w}_i(k+1) &= \mathbf{w}_i(k) + \mu_k\left([\mathbf{I}_n - \mathbf{w}(k)\mathbf{w}^T(k)][\mathbf{I}_n - \mathbf{x}(k)\mathbf{x}^T(k)]\right. \\ &\qquad \left. -\beta\sum_{i=k+1}^{n}\mathbf{w}_{t,i}\mathbf{w}_{t,i}^T\mathbf{x}(k)\mathbf{x}^T(k)\right)\mathbf{w}_i(k). (42.6.10)\end{aligned}$$

Oja [52] showed that, under the conditions that the eigenvalues are distinct, and that $\lambda_{n-r+1} < 1$ and $\beta > \frac{\lambda_{n-r+1}}{\lambda_n} - 1$, the only asymptotically stable points of the associated ODE are the eigenvectors $\{\pm\mathbf{v}_n, \ldots, \pm\mathbf{v}_{n-r+1}\}$. Note that the magnitude of the eigenvalues must be controlled in practice by normalizing $\mathbf{x}(k)$ so that the expression between brackets in (42.6.10) becomes homogeneous.

The derivation of these algorithms seems empirical. In fact, they have been derived from slight modifications of the ODE (42.7.8) associated with the Oja's neuron in order to keep adequate conditions of stability (see e.g., [52]). It was established by Oja [51], Sanger [66] and Oja *et al* [54] for the SGA, GHA and WSA algorithms respectively, that the only asymptotically stable points of their associated ODE are the eigenvectors $\{\pm\mathbf{v}_1, \ldots, \pm\mathbf{v}_r\}$. We note that the first vector ($k = 1$) estimated by the SGA and GHA algorithms, and the vector ($r = k = 1$) estimated by the SNL and WSA algorithms gives the *Constrained Hebbian learning rule* of the basic PCA neuron (42.4.1) introduced by Oja [49].

A performance analysis of different eigenvector power-based algorithms has been presented in [22]. In particular, the asymptotic distribution of the eigenvector estimates and of the associated projection matrices given by these stochastic algorithms with constant step size μ for stationary data has been derived, where closed-form expressions of the covariance of these distributions has been given and analyzed for independent Gaussian distributed data $\mathbf{x}(k)$. Closed-form expressions of the mean square error of these estimators has been deduced and analyzed. In particular, they allow us to specify the influence of the different parameters $(\alpha_2, \ldots, \alpha_r)$, $(\beta_1, \ldots, \beta_r)$ and β of these algorithms on their performance and to take into account tradeoffs between the misadjustment and the speed of convergence. An example of such derivation and analysis is given for the Oja's Neuron in Subsection 42.7.3.1.

42.6.2.1 Eigenvector power-based methods issued from exponential windows

Using the exponential windowed estimates (42.3.5) of $\mathbf{C}_x(k)$, and following the concept of power method (42.2.9) and the subspace deflation technique introduced in [41], the following algorithm has been proposed in [37]

$$\begin{aligned} \mathbf{w}'_i(k+1) &= \mathbf{C}_i(k)\mathbf{w}_i(k) && (42.6.11) \\ \mathbf{w}_i(k+1) &= \mathbf{w}'_i(k+1)/\|\mathbf{w}'_i(k+1)\|_2, && (42.6.12) \end{aligned}$$

where $\mathbf{C}_i(k) = \beta\mathbf{C}_i(k-1) + \mathbf{x}(k)\mathbf{x}^T(k)[\mathbf{I}_n - \sum_{j=1}^{i-1}\mathbf{w}_j(k)\mathbf{w}_j^T(k)]$ for $i = 1, ..., r$. Applying the approximation $\mathbf{w}'_i(k) \approx \mathbf{C}_i(k-1)\mathbf{w}_i(k)$ in (42.6.11) to reduce the complexity, (42.6.11) becomes

$$\mathbf{w}'_i(k+1) = \beta\mathbf{w}'_i(k) + \mathbf{x}(k)[g_i(k) - \mathbf{y}_i^T(k)\mathbf{c}_i(k)] \qquad (42.6.13)$$

with $g_i(k) \overset{\text{def}}{=} \mathbf{x}^T(k)\mathbf{w}_i(k)$, $\mathbf{y}_i(k) \overset{\text{def}}{=} [\mathbf{w}_1(k), ..., \mathbf{w}_{i-1}(k)]^T\mathbf{x}(k)$ and $\mathbf{c}_i(k) \overset{\text{def}}{=} [\mathbf{w}_1(k), ..., \mathbf{w}_{i-1}(k)]^T\mathbf{w}_i(k)$. Equations (42.6.13) and (42.6.11) should be run successively for $i = 1, ..., r$ at each iteration k.

Note that an up to a common factor estimate of the eigenvalues $\lambda_i(k+1)$ of $\mathbf{C}_x(k)$ can be updated as follows. From (42.6.11), one can write

$$\lambda_i(k+1) \overset{\text{def}}{=} \mathbf{w}_i^T(k)\mathbf{C}_i(k)\mathbf{w}_i(k) = \mathbf{w}_i^T(k)\mathbf{w}'_i(k+1). \qquad (42.6.14)$$

Using (42.6.13) and applying the approximations $\lambda_i(k) \approx \mathbf{w}_i^T(k)\mathbf{w}'_i(k)$ and $\mathbf{c}_i(k) \approx \mathbf{0}$, one can replace (42.6.14) by

$$\lambda_i(k+1) = \beta\lambda_i(k) + |g_i(k)|^2,$$

that can be used to track the rank r and the signal eigenvectors, as in [71].

42.6.3 Projection approximation-based methods

A variant of the PAST algorithm, named PASTd and presented in [70], allows one to estimate the r dominant eigenvectors. This algorithm is based on a deflation technique that consists in estimating sequentially the eigenvectors. First the most dominant estimated eigenvector $\mathbf{w}_1(k)$ is updated by applying the PAST algorithm with $r = 1$. Then the projection of the current data $\mathbf{x}(k)$ onto this estimated eigenvector is removed from $\mathbf{x}(k)$ itself. Because now the second dominant eigenvector becomes the most dominant one in the updated data vector ($\mathrm{E}\left[(\mathbf{x}(k) - \mathbf{v}_1\mathbf{v}_1^T\mathbf{x}(k))(\mathbf{x}(k) - \mathbf{v}_1\mathbf{v}_1^T\mathbf{x}(k))^T\right] = \mathbf{C}_x(k) - \lambda_1\mathbf{v}_1\mathbf{v}_1^T$), it can be extracted in the same way as before. Applying this procedure repeatedly, all the r dominant eigenvectors and the associated eigenvalues are estimated sequentially. These estimated eigenvalues may be used to estimate the rank r if it is not known a priori [71]. It is interesting to note that for $r = 1$, the PAST and the PASTd algorithms, that are identical, simplify as

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu_k[\mathbf{I}_n - \mathbf{w}(k)\mathbf{w}^T(k)]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k), \tag{42.6.15}$$

where $\mu_k = \frac{1}{\sigma_y^2(k)}$ with $\sigma_y^2(k+1) = \beta\sigma_y^2(k) + y^2(k)$ and $y(k) \overset{\text{def}}{=} \mathbf{w}^T(k)\mathbf{x}(k)$. A comparison with Oja's neuron (42.4.1) shows that both algorithms are identical except for the step size. While Oja's neuron uses a fixed step size μ which needs careful tuning, (42.6.15) implies a time varying, self-tuning step size μ_k. The numerical experiments presented in [70] show that this deflation procedure causes a stronger loss of orthonormality between $\mathbf{w}_i(k)$ and a slight increase of the error in the successive estimates $\mathbf{w}_i(k)$. By invoking the ODE approach (see Section 42.7.1), it has been proved in [72] for stationary signals and other weak conditions, the PASTd algorithm converges to the desired r dominant eigenvectors with probability one.

In contrast to the PAST algorithm, the PASTd algorithm can be used to estimate the minor eigenvectors by changing the sign of the step size with an orthonormalization of the estimated eigenvectors at each step. It has been proved [64] that for $\beta = 1$, the only locally asymptotically stable points of the associated ODE are the desired eigenvectors $\{\pm\mathbf{v}_n, \ldots, \pm\mathbf{v}_{n-r+1}\}$. To reduce the complexity of the Gram-Schmidt orthonormalization step used in [64], [9] proposed a modification of this part.

42.6.4 Additional methodologies

Among the other approaches to adaptively estimate the eigenvectors of a covariance matrix, the *Maximum Likelihood Adaptive Subspace Estimation* (MALASE) [18] provides a number of desirable features. It is based on the adaptive maximization of the log-likelihood of the EVD parameters associated with the covariance matrix $\mathbf{C}_x$ for Gaussian distributed zero-mean data $\mathbf{x}(k)$. Up to an additive constant, this log-likelihood is given by

$$\begin{aligned} L(\mathbf{W}, \mathbf{\Lambda}) &= -\ln(\det \mathbf{C}_x) - \mathbf{x}^T(k)\mathbf{C}_x^{-1}\mathbf{x}(k) \\ &= -\sum_{i=1}^{n} \ln(\lambda_i) - \mathbf{x}^T(k)\mathbf{W}\mathbf{\Lambda}^{-1}\mathbf{W}^T\mathbf{x}(k), \end{aligned} \tag{42.6.16}$$

where $\mathbf{C}_x = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^T$ represents the EVD of $\mathbf{C}_x$ with $\mathbf{W}$ an orthogonal $n \times n$ matrix and $\mathbf{\Lambda} = \mathrm{Diag}(\lambda_1, ..., \lambda_n)$. This is a quite natural criterion for statistical estimation purposes, even if the minimum variance property of the likelihood functional is actually an asymptotic property. To deduce an adaptive algorithm, a gradient ascent procedure has been proposed in [18] in which a new data $\mathbf{x}(k)$ is used at each time iteration k of the maximization of (42.6.16). Using the differential of $L(\mathbf{W}, \boldsymbol{\Lambda})$ defined on the manifold of $n \times n$ orthogonal matrices (see [21, pp. 62-63] or Exercice 42.15 (42.9.7)), we obtain the following gradient of $L(\mathbf{W}, \boldsymbol{\Lambda})$

$$\begin{aligned} \boldsymbol{\nabla}_{\mathbf{W}} L &= \mathbf{W}\left[\boldsymbol{\Lambda}^{-1}\mathbf{y}(k)\mathbf{y}^T(k) - \mathbf{y}(k)\mathbf{y}^T(k)\boldsymbol{\Lambda}^{-1}\right], \\ \boldsymbol{\nabla}_{\boldsymbol{\Lambda}} L &= -\boldsymbol{\Lambda}^{-1} + \boldsymbol{\Lambda}^{-2}\mathrm{Diag}(\mathbf{y}(k)\mathbf{y}^T(k)), \end{aligned}$$

where $\mathbf{y}(k) \overset{\text{def}}{=} \mathbf{W}^T\mathbf{x}(k)$. Then, the stochastic gradient update of $\mathbf{W}$ yields

$$\begin{aligned} \mathbf{W}(k+1) &= \mathbf{W}(k) + \mu_k \mathbf{W}(k)\left[\boldsymbol{\Lambda}^{-1}(k)\mathbf{y}(k)\mathbf{y}^T(k) - \mathbf{y}(k)\mathbf{y}^T(k)\boldsymbol{\Lambda}^{-1}(k)\right] && (42.6.17) \\ \boldsymbol{\Lambda}(k+1) &= \boldsymbol{\Lambda}(k) + \mu_k'\left[\boldsymbol{\Lambda}^{-2}(k)\mathrm{Diag}(\mathbf{y}(k)\mathbf{y}^T(k)) - \boldsymbol{\Lambda}^{-1}(k)\right], && (42.6.18) \end{aligned}$$

where the step sizes μ_k and μ_k' are possibly different. We note that, starting from an orthonormal matrix $\mathbf{W}(0)$, the sequence of estimates $\mathbf{W}(k)$ given by (42.6.17) is orthonormal up to the second-order term in μ_k only. To ensure in practice the convergence of this algorithm, is has been shown in [18] that it is necessary to orthonormalize quite often $\mathbf{W}(k)$ to compensate for the orthonormality drift in $O(\mu_k^2)$. Using continuous-time system theory and differential geometry [21], a modification of (42.6.17) has been proposed in [18]. It is clear that $\boldsymbol{\nabla}_{\mathbf{W}} L$ is tangent to the curve defined by

$$\mathbf{W}(t) = \mathbf{W}(0)\exp\left[t\left(\boldsymbol{\Lambda}^{-1}\mathbf{y}(k)\mathbf{y}^T(k) - \mathbf{y}(k)\mathbf{y}^T(k)\boldsymbol{\Lambda}^{-1}\right)\right]$$

for $t = 0$, where the matrix exponential is defined e.g., in [34, chap. 11]. Furthermore, we note that this curve lies in the manifold of orthogonal matrices if $\mathbf{W}(0)$ is orthogonal because $\exp(\mathbf{A})$ is orthogonal if and only if $\mathbf{A}$ is skew-symmetric ($\mathbf{A}^T = -\mathbf{A}$) and matrix $\boldsymbol{\Lambda}^{-1}\mathbf{y}(k)\mathbf{y}^T(k) - \mathbf{y}(k)\mathbf{y}^T(k)\boldsymbol{\Lambda}^{-1}$ is clearly skew-symmetric. Moving on the curve $\mathbf{W}(t)$ from point $t = 0$ in the direction of increasing values of $\boldsymbol{\nabla}_{\mathbf{W}} L$ amounts to letting t increase. Thus, a discretized version of the optimization of $L(\mathbf{W}, \boldsymbol{\Lambda})$ as a continuous function of $\mathbf{W}$ is given by the following update scheme

$$\mathbf{W}(k+1) = \mathbf{W}(k)\exp\left[\mu_k\left(\boldsymbol{\Lambda}^{-1}(k)\mathbf{y}(k)\mathbf{y}^T(k) - \mathbf{y}(k)\mathbf{y}^T(k)\boldsymbol{\Lambda}^{-1}(k)\right)\right], \quad (42.6.19)$$

and the coupled update equations (42.6.18) and (42.6.19) form the MALASE algorithm. As mentioned above the update factor $\exp\left[\mu_k\left(\boldsymbol{\Lambda}^{-1}(k)\mathbf{y}(k)\mathbf{y}^T(k) - \mathbf{y}(k)\mathbf{y}^T(k)\boldsymbol{\Lambda}^{-1}(k)\right)\right]$ is an orthogonal matrix. This ensures that the orthonormality property is preserved by MALASE algorithm, provided that the algorithm is initialized with an orthogonal matrix $\mathbf{W}(0)$. However, it has been shown by numerical experiments presented in [18], that it is not necessary to have $\mathbf{W}(0)$ orthogonal to ensure the convergence, since MALASE algorithm steers $\mathbf{W}(k)$ towards the manifold of orthogonal matrices. The MALASE algorithm seems to involve high computational cost, due to the matrix exponential that applies in (42.6.19). However, since $\exp\left[\mu_k\left(\boldsymbol{\Lambda}^{-1}(k)\mathbf{y}(k)\mathbf{y}^T(k) - \mathbf{y}(k)\mathbf{y}^T(k)\boldsymbol{\Lambda}^{-1}(k)\right)\right]$ is the exponential of a sum of two rank one matrices, the calculation of this matrix requires only $O(n^2)$ operations [18]. Originally, this algorithm that updates the EVD of the covariance matrix $\mathbf{C}_x(k)$ can be modified by a simple preprocessing to estimate the principal or minor r signal eigenvectors only, when the remaining $n - r$ eigenvectors are associated with a common eigenvalue $\sigma^2(k)$ (see Subsection 42.3.1). This algorithm, denoted MALASE(r) requires $O(nr)$ operations by iteration. Finally, note that a theoretical analysis of convergence has been presented in [18]. It is proved that in stationary environments, the stationary stable points of the algorithm (42.6.18),(42.6.19) correspond to the EVD of $\mathbf{C}_x$. Furthermore, the covariance of the asymptotic distribution of the estimated parameters is given for Gaussian independently distributed data $\mathbf{x}(k)$ using general results of Gaussian approximation (see Subsection 42.7.2).

42.6.5 Particular case of second-order stationary data

Finally, note that for $\mathbf{x}(k) = [x(k), x(k-1), ..., x(k-n+1)]^T$ comprising of time delayed versions of scalar valued second-order stationary data $x(k)$, the covariance matrix $\mathbf{C}_x(k) = \mathrm{E}[\mathbf{x}(k)\mathbf{x}^T(k)]$ is Toeplitz and consequently centro-symmetric. This property occurs in important applications: temporal covariance matrices obtained from a uniform sampling of a second-order stationary signals, and spatial covariance matrices issued from uncorrelated and band-limited sources observed on a centro-symmetric sensor array (for example on uniform linear arrays). This centro-symmetric structure of $\mathbf{C}_x$ allows us to use for real-valued data, the property[9] [14] that its EVD can be obtained from two orthonormal eigenbases of half-size real symmetric matrices. For example if n is even, $\mathbf{C}_x$ can be partitioned as follows

$$\mathbf{C}_x = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2^T \\ \mathbf{C}_2 & \mathbf{J}\mathbf{C}_1\mathbf{J} \end{bmatrix},$$

where $\mathbf{J}$ is an $n/2 \times n/2$ matrix with ones on its anti-diagonal and zeroes elsewhere. Then, the n unit 2-norm eigenvectors $\mathbf{v}_i$ of $\mathbf{C}_x$ are given by $n/2$ symmetric and $n/2$ skew symmetric vectors $\mathbf{v}_i = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{u}_i \\ \epsilon_i \mathbf{J}\mathbf{u}_i \end{bmatrix}$ where $\epsilon_i = \pm 1$, respectively issued from the unit 2-norm eigenvectors $\mathbf{u}_i$ of $\mathbf{C}_1 + \epsilon_i \mathbf{J}\mathbf{C}_2 = \frac{1}{2}\mathrm{E}[(\mathbf{x}'(k) + \epsilon_i \mathbf{J}\mathbf{x}"(k))(\mathbf{x}'(k) + \epsilon_i \mathbf{J}\mathbf{x}"(k))^T]$ with $\mathbf{x}(k) = [\mathbf{x}'^T(k), \mathbf{x}"^T(k)]^T$. This property has been exploited [23, 26] to reduce the computational cost of the previously introduced eigenvectors adaptive algorithms. Furthermore, the conditioning of these two independent EVD is improved with respect to the EVD of $\mathbf{C}_x$ since the difference between two consecutive eigenvalues increases in general. Compared to the estimators that do not take the centro-symmetric structure into account, the performance ought to be improved. This has been proved in [26], using closed-form expressions of the asymptotic bias and covariance of eigenvectors power-based estimators with constant step size μ derived in [22] for independent Gaussian distributed data $\mathbf{x}(k)$. Finally, note that the deviation from orthonormality is reduced and the convergence speed is improved, yielding a better tradeoff between convergence speed and misadjustment.

42.7 CONVERGENCE AND PERFORMANCE ANALYSIS ISSUES

Several tools may be used to assess the "convergence" and the performance of the previously described algorithms. First of all, note that despite the simplicity of the LMS algorithm (see e.g., [35])

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\mathbf{x}(k)[y(k) - \mathbf{x}^T(k)\mathbf{w}(k)],$$

its convergence and associated analysis has been the subject of many contributions in the past three decades (see e.g., [67] and references therein). However, in-depth theoretical studies is still a matter of utmost interest. Consequently, due to their complexity with

[9] Note that for Hermitian centro-symmetric covariance matrices, such property does not extend. But any eigenvector $\mathbf{v}_i$ satisfies the relation $[\mathbf{v}_i]_k = e^{i\phi_i}[\mathbf{v}_i^*]_{n-k}$, that can be used to reduce the computational cost by a factor 2.

respect to the LMS algorithm, results about the convergence and performance analysis of subspaces or eigenvectors tracking will be much weaker.

To study the convergence of the algorithms introduced in the previous two sections from a theoretical point of view, the data $\mathbf{x}(k)$ will be supposed stationary and the step size μ_k will be considered as decreasing. In these conditions, according to the addressed problem, some questions arise. Does the sequence $\mathbf{W}(k)\mathbf{W}^T(k)$ converge almost surely to the signal $\mathbf{\Pi}_s$ or the noise projector $\mathbf{\Pi}_n$ and does the sequence $\mathbf{W}^T(k)\mathbf{W}(k)$ converge almost surely to $\mathbf{I}_r$ for the subspace tracking problem or does the sequence $\mathbf{W}(k)$ converge to the signal or the noise eigenvectors $[\pm\mathbf{u}_1, ..., \pm\mathbf{u}_r]$ or $[\pm\mathbf{u}_{n-r+1}, ..., \pm\mathbf{u}_n]$ for the eigenvectors tracking problems? These questions are very challenging, but using the stability of the associated ODE, a partial response will be given in Subsection 42.7.1.

Now, from a practical point of view, the step size sequence μ_k is reduced to a "small" constant μ to track signal or noise subspaces (or signal or noise eigenvectors) with possible nonstationary data $\mathbf{x}(k)$. Under these conditions, the previous sequences do not converge almost surely any longer even for stationary data $\mathbf{x}(k)$. Nevertheless, if for stationary data, these algorithms converge almost surely with a decreasing step size, their estimate $\boldsymbol{\theta}(k)$ ($\mathbf{W}(k)\mathbf{W}^T(k)$, $\mathbf{W}^T(k)\mathbf{W}(k)$ or $\mathbf{W}(k)$ according to the problem) will oscillate around their limit $\boldsymbol{\theta}_*$ ($\mathbf{\Pi}_s$ or $\mathbf{\Pi}_n$, $\mathbf{I}_r$, $[\pm\mathbf{u}_1, ..., \pm\mathbf{u}_r]$ or $[\pm\mathbf{u}_{n-r+1}, ..., \pm\mathbf{u}_n]$, according to the problem) with a constant "small" step size. In these later conditions, the performance of the algorithms will be assessed by the covariance matrix of the errors $(\boldsymbol{\theta}(k) - \boldsymbol{\theta}_*)$ using some results of Gaussian approximation recalled in Subsection 42.7.2.

Unfortunately, the study of the stability of the associated ODE and the derivation of the covariance of the errors are not always possible due to their complex forms. In these cases, the "convergence" and the performance of the algorithms for stationary data will be assessed by first order analysis using coarse approximations. In practice, this analysis will be only possible for independent data $\mathbf{x}(k)$ and assuming the step size μ "sufficiently small" to keep terms that are at most of the order of μ in the different used expansions. An example of such analysis has been used in [29] and [74] to derive an approximate expression of the mean of the deviation from orthonormality $\mathrm{E}[\mathbf{W}^T(k)\mathbf{W}(k) - \mathbf{I}_r]$ for the estimate $\mathbf{W}(k)$ given by the FRANS algorithm (described in Subsection 42.5.1.2) that allows to explain the difference in behavior of this algorithm when estimating the noise and signal subspaces.

42.7.1 A short review of the ODE method

The so-called ODE [42, 13] is a powerful tool to study the asymptotic behavior of the stochastic approximation algorithms of the general form[10]

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \mu_k f(\boldsymbol{\theta}(k), \mathbf{x}(k)) + \mu_k^2 h(\boldsymbol{\theta}(k), \mathbf{x}(k)), \qquad (42.7.1)$$

with $\mathbf{x}(k) = g(\boldsymbol{\xi}(k))$, where $\boldsymbol{\xi}(k)$ is a Markov chain that does not depend on $\boldsymbol{\theta}$, $f(\boldsymbol{\theta}, \mathbf{x})$ and $h(\boldsymbol{\theta}, \mathbf{x})$ are "regular enough" functions, and where $(\mu_k)_{k\in\mathcal{N}}$ is a positive sequence of constants, converging to zero, and satisfying the assumption $\sum_k \mu_k = \infty$. Then, the convergence properties of the discrete time stochastic algorithm (42.7.1) is intimately connected to the stability properties of the deterministic ODE associated with (42.7.1),

[10] The most common form of stochastic approximation algorithms corresponds to $h(.) = 0$. This residual perturbation term $\mu_k^2 h(\boldsymbol{\theta}(k), \mathbf{x}(k))$ will be used to write the trajectories governed by the estimated projector $\mathbf{P}(k) = \mathbf{W}(k)\mathbf{W}^T(k)$.

which is defined as the first-order ordinary differential equation

$$\frac{d\boldsymbol{\theta}(t)}{dt} = \bar{f}(\boldsymbol{\theta}(t)), \tag{42.7.2}$$

where the function $\bar{f}(\boldsymbol{\theta})$ is defined by

$$\bar{f}(\boldsymbol{\theta}) \stackrel{\text{def}}{=} \mathrm{E}[f(\boldsymbol{\theta}, \mathbf{x}(k))], \tag{42.7.3}$$

where the expectation is taken only with respect to the data $\mathbf{x}(k)$ and $\boldsymbol{\theta}$ is assumed deterministic. We first recall in the following some definitions and results of stability theory of ODE (i.e., the asymptotic behavior of trajectories of the ODE) and then, we will specify its connection to the convergence of the stochastic algorithm (42.7.1). The *stationary points* of this ODE are the values $\boldsymbol{\theta}_*$ of $\boldsymbol{\theta}$ for which the driving term $\bar{f}(\boldsymbol{\theta})$ vanishes; hence the term stationary ponts. This gives $\bar{f}(\boldsymbol{\theta}_*) = \mathbf{0}$, so that the motion of the trajectory ceases. A stationary point $\boldsymbol{\theta}_*$ of the ODE is said to be

- *stable* if for an arbitrary neighborhood of $\boldsymbol{\theta}_*$, the trajectory $\boldsymbol{\theta}(t)$ stays in this neighborhood for an initial condition $\boldsymbol{\theta}(0)$ in another neighborhood of $\boldsymbol{\theta}_*$;

- *locally asymptotically stable* if there exists a neighborhood of $\boldsymbol{\theta}_*$ such that for all initial conditions $\boldsymbol{\theta}(0)$ in this neighborhood, the ODE (42.7.2) forces $\boldsymbol{\theta}(t) \to \boldsymbol{\theta}_*$ as $t \to \infty$;

- *globally asymptotically stable* if for all possible values of initial conditions $\boldsymbol{\theta}(0)$, the ODE (42.7.2) forces $\boldsymbol{\theta}(t) \to \boldsymbol{\theta}_*$ as $t \to \infty$;

- *unstable* if for all neighborhoods of $\boldsymbol{\theta}_*$, there exists some initial value $\boldsymbol{\theta}(0)$ in this neighborhood for which the ODE (42.7.2) do not force $\boldsymbol{\theta}(t)$ to converge to $\boldsymbol{\theta}_*$ as $t \to \infty$.

Assuming that the set of stationary points can be derived, two standard methods are used to test for stability. They are summarized in the following. The first one consists in finding a Lyapunov function $L(\boldsymbol{\theta})$ for the differential equation (42.7.2), i.e., a positive valued function that is decreasing along all trajectories. In this case, it is proved (see e.g., [12]) that the set of the stationary points $\boldsymbol{\theta}_*$ are asymptotically stable. This stability is local if this decrease occurs from an initial condition $\boldsymbol{\theta}(0)$ located in a neighborhood of the stationary points and global if the initial condition can be arbitrary. If $\boldsymbol{\theta}_*$ is a (locally or globally) stable stationary point, then such a Lyapunov function necessarily exists [12]. But for general nonlinear functions $\bar{f}(\boldsymbol{\theta})$, no general recipe exists for finding such a function. Instead, one must try many candidate Lyapunov functions in the hopes of uncovering one which works.

However, for specific functions $\bar{f}(\boldsymbol{\theta})$ which constitute negative gradient vectors of a positive scalar function $J(\boldsymbol{\theta})$:

$$\bar{f}(\boldsymbol{\theta}) = -\boldsymbol{\nabla}_{\boldsymbol{\theta}} J \;\; \text{with} \;\; J > 0,$$

then, all the trajectories of the ODE (42.7.2) converge to the set of the stationary points of the ODE (see Exercice 42.16). Consequently, the set of the stationary points is globally asymptotically stable for this ODE.

The second method consists in a local linearization of the ODE (42.7.2) about each stationary point $\boldsymbol{\theta}_*$ in which case a stationary point is locally asymptotically stable if and only if the locally linearized equation is asymptotically stable. Consequently the final conclusion amounts to an eigenvalue check of the matrix $\frac{d\bar{f}(\boldsymbol{\theta})}{d\boldsymbol{\theta}}_{|\boldsymbol{\theta}=\boldsymbol{\theta}_*}$. More precisely (see Exercice 42.17), if $\boldsymbol{\theta}_* \in \mathcal{R}^m$ is a stationary point of the ODE (42.7.2), and $\nu_1, ..., \nu_m$ are the eigenvalues of the $m \times m$ matrix $\frac{d\bar{f}(\boldsymbol{\theta})}{d\boldsymbol{\theta}}_{|\boldsymbol{\theta}=\boldsymbol{\theta}_*}$, then (see Exercice 42.17 or [12] for a formal proof)

- if all eigenvalues $\nu_1, ..., \nu_m$ have strictly negative real parts, $\boldsymbol{\theta}_*$ is a locally asymptotically stable point;
- if there exists ν_i among $\nu_1, ..., \nu_m$ such that $\Re(\nu_i) > 0$, $\boldsymbol{\theta}_*$ is an unstable point;
- if for all eigenvalues $\nu_1, ..., \nu_m$, $\Re(\nu_i) \leq 0$ and for at least one eigenvalue ν_{i_0} among $\nu_1, ..., \nu_m$, $\Re(\nu_{i_0}) = 0$, we cannot conclude.

Considering now the connection between the stability properties of the associated deterministic ODE (42.7.2) and the convergence properties of the discrete time stochastic algorithm (42.7.1), several results are available. First, the sequence $\boldsymbol{\theta}(k)$ generated by the algorithm (42.7.1) can only converge almost surely [42][13] to a (locally or globally) asymptotically stable stationary point of the associated ODE (42.7.2). But deducing some convergence results about the stochastic algorithm (42.7.1) from the stability of the associated ODE is not trivial because a stochastic algorithm have much more complex asymptotic behavior than a given solution of its associated deterministic ODE. However under additional technical assumptions, it is proved [31] that if the ODE has a finite number of globally (up to a Lebesgue measure zero set of initial conditions) asymptotically stable stationary points $(\boldsymbol{\theta}_{*_i})_{i=1,...,d}$ and if each trajectory of the ODE converges towards one of theses points, then the sequence $\boldsymbol{\theta}(k)$ generated by the algorithm (42.7.1) converges almost surely to one of these points. The conditions of the result are satisfied in particular if the mean field $\bar{f}(\boldsymbol{\theta})$ can be written as $\bar{f}(\boldsymbol{\theta}) = -\boldsymbol{\nabla}_{\boldsymbol{\theta}} J$ where $\boldsymbol{\nabla}_{\boldsymbol{\theta}} J$ is a positive valued function admitting a finite number of local minima. In this later case, this result has been extended for an infinite number of isolated minima in [32].

In adaptive processing, we do not wish a decreasing step size sequence, since we would then lose the tracking capability of the algorithms. To be able to track the possible non stationarity of the data $\mathbf{x}(k)$, the sequence of step size is reduced to a "small" constant parameter μ. In this case, the stochastic algorithm (42.7.1) does not converge almost surely even for stationary data and the rigorous results concerning the asymptotic behavior of (42.7.1) are less powerful. However, when the set of all stable points $(\boldsymbol{\theta}_{*_i})_{i=1,...,d}$ of the associated ODE (42.7.2) is globally asymptotically stable (up to a zero measure set of initial conditions), the weak convergence approach developed by Kushner [43] suggests that for a "sufficiently small" μ, $\boldsymbol{\theta}(k)$ will oscillate around one of the limit points $\boldsymbol{\theta}_{*_i}$ of the decreasing step size stochastic algorithm. In particular, one should note that, when there exist more than one possible limits ($d \neq 1$), the algorithm may oscillate around one of them $\boldsymbol{\theta}_{*_i}$, and then move into a neighborhood of another equilibrium point $\boldsymbol{\theta}_{*_j}$. However, the probability of such events decreases to zero as $\mu \rightarrow 0$, so that their implication is marginal in most cases.

42.7.2 A short review of a general Gaussian approximation result

For constant step size algorithms and stationary data, we will use the following result proved in [13, th.2, p.108] under a certain number of hypotheses. Consider the constant step size stochastic approximation algorithm (42.7.1). Suppose that $\boldsymbol{\theta}(k)$ converges almost surely to the unique globally asymptotically stable point $\boldsymbol{\theta}_*$ in the corresponding decreasing step size algorithm. Then, if $\boldsymbol{\theta}_\mu(k)$ denotes the value of $\boldsymbol{\theta}(k)$ associated with the algorithm of step size μ, we have when $\mu \to 0$ and $k \to \infty$ (where $\overset{\mathcal{L}}{\to}$ denotes the convergence in distribution and $\mathcal{N}(\mathbf{m}, \mathbf{C}_x)$, the Gaussian distribution of mean $\mathbf{m}$ and covariance $\mathbf{C}_x$)

$$\frac{1}{\sqrt{\mu}}\ (\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*) \overset{\mathcal{L}}{\to} \mathcal{N}\left(\mathbf{0}, \mathbf{C}_{\boldsymbol{\theta}}\right), \tag{42.7.4}$$

where $\mathbf{C}_{\boldsymbol{\theta}}$ is the unique solution of the continuous-time Lyapunov equation:

$$\mathbf{D}\mathbf{C}_{\boldsymbol{\theta}} + \mathbf{C}_{\boldsymbol{\theta}}\mathbf{D}^T + \mathbf{G} = \mathbf{O}, \tag{42.7.5}$$

where $\mathbf{D}$ and $\mathbf{G}$ are, respectively, the derivative of the mean field $\bar{f}(\boldsymbol{\theta})$ and the following sum of covariances of the field $f(\boldsymbol{\theta}, \mathbf{x}(k))$ of the algorithm (42.7.1):

$$\mathbf{D} \stackrel{\text{def}}{=} \frac{d\bar{f}(\boldsymbol{\theta})}{d\boldsymbol{\theta}}_{|\boldsymbol{\theta}=\boldsymbol{\theta}_*} \qquad \left([\mathbf{D}]_{i,j} \stackrel{\text{def}}{=} \frac{\partial \bar{f}_i(\boldsymbol{\theta})}{\partial \theta_j}\right) \tag{42.7.6}$$

$$\mathbf{G} \stackrel{\text{def}}{=} \sum_{k=-\infty}^{\infty} \text{Cov}\{f(\boldsymbol{\theta}_*,, \mathbf{x}(k)), f(\boldsymbol{\theta}_*, \mathbf{x}(0))\} = \sum_{k=-\infty}^{\infty} \text{E}\{[f(\boldsymbol{\theta}_*, \mathbf{x}(k))][f(\boldsymbol{\theta}_*, \mathbf{x}(0))]^T\}. \tag{42.7.7}$$

Note that all the eigenvalues of the derivative $\mathbf{D}$ of the mean field have strictly negative real parts since $\boldsymbol{\theta}_*$ is an asymptotically stable point of (42.7.2) and that for independent data $\mathbf{x}(k)$, $\mathbf{G}$ is simply the covariance of the field. Unless we have sufficient information about the data, which is often not the case, in practice we consider the simplifying hypothesis of independent identically Gaussian distributed data $\mathbf{x}(k)$.

It should be mentioned that the rigorous proof of this result (42.7.4) needs a very strong hypothesis on the algorithm (42.7.1), namely that $\boldsymbol{\theta}(k)$ converges almost surely to the unique globally asymptotically stable point $\boldsymbol{\theta}_*$ in the corresponding decreasing step size algorithm. However, the practical use of (42.7.4) in more general situations is usually justified by using formally a general diffusion approximation result [13, th.1, p.104].

In practice, μ is "small" and fixed, but it is assumed that the asymptotic distribution of $\mu^{-1/2}\,(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)$ when k tends to ∞ can still be approximated by a zero mean Gaussian distribution of covariance $\mathbf{C}_{\boldsymbol{\theta}}$, and consequently that for "large enough" k, the distribution of the residual error $(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)$ is a zero mean Gaussian distribution of covariance $\mu\mathbf{C}_{\boldsymbol{\theta}}$ where $\mathbf{C}_{\boldsymbol{\theta}}$ is solution of the Lyapunov equation (42.7.5). Note that the approximation $\text{E}[(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)^T] \approx \mu\mathbf{C}_{\boldsymbol{\theta}}$ enables us to derive an expression of the asymptotic bias $\text{E}[\boldsymbol{\theta}_\mu(k)] - \boldsymbol{\theta}_*$ from a perturbation analysis of the expectation of both sides of (42.7.1) when the field $f(\boldsymbol{\theta}(k), \mathbf{x}(k))$ is linear in $\mathbf{x}(k)\mathbf{x}^T(k)$. An example of such a derivation is given in Subsection 42.7.3.1, [26] and Exercice 42.18.

Finally, let us recall that there is no relation between the asymptotic performance of the stochastic approximation algorithm (42.7.1) and its *convergence rate*. As it is well known, the convergence rate depends on the transient behavior of the algorithm, for which no general result seems to be available. For this reason, different authors (e.g., [22],[26]) have resorted to simulations to compare the convergence speed of different algorithms whose associated step sizes μ are chosen to provide the same value of the mean square error $\text{E}\|(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)\|_2 \approx \mu\text{Tr}(\mathbf{C}_{\boldsymbol{\theta}})$.

42.7.3 Examples of convergence and performance analysis

Using the previously described methods, two examples of convergence and performance analysis will be given. Oja's neuron algorithm as the simplest algorithm will allow us to present a comprehensive study of an eigenvector tracking algorithm. Then the Oja's algorithm will be studied as an example of a subspace tracking algorithm.

42.7.3.1 Convergence and performance analysis of the Oja's neuron Consider Oja's neuron algorithms (42.4.1) and (42.4.3) introduced in Section 42.4. The stationary points of their associated ODE

$$\frac{d\mathbf{w}(t)}{dt} = \mathbf{C}_x\mathbf{w}(t) - \mathbf{w}(t)[\mathbf{w}(t)^T\mathbf{C}_x\mathbf{w}(t)] \quad \left[\text{resp.}, -\mathbf{C}_x\mathbf{w}(t) + \mathbf{w}(t)[\mathbf{w}(t)^T\mathbf{C}_x\mathbf{w}(t)]\right] \tag{42.7.8}$$

are the roots of $\mathbf{C}_x\mathbf{w} = \mathbf{w}[\mathbf{w}^T\mathbf{C}_x\mathbf{w}]$ and thus are clearly given by $(\pm\mathbf{u}_k)_{k=1,\ldots,n}$. To study the stability of these stationarity points, consider the derivative $\mathbf{D_w}$ of the mean field $\mathbf{C}_x\mathbf{w} - \mathbf{w}[\mathbf{w}^T\mathbf{C}_x\mathbf{w}]$ [resp., $-\mathbf{C}_x\mathbf{w} + \mathbf{w}[\mathbf{w}^T\mathbf{C}_x\mathbf{w}]$] at these points. Using a standard first order perturbation, we obtain

$$\begin{aligned}\mathbf{D_w}(\pm\mathbf{u}_k) \quad &= \quad \mathbf{C}_x - (\mathbf{w}^T\mathbf{C}_x\mathbf{w})\mathbf{I}_n - 2\mathbf{w}\mathbf{w}^T\mathbf{C}_{x|\mathbf{w}=\pm\mathbf{u}_k} \\ &\qquad\qquad [\text{resp.}, -\mathbf{C}_x + (\mathbf{w}^T\mathbf{C}_x\mathbf{w})\mathbf{I}_n + 2\mathbf{w}\mathbf{w}^T\mathbf{C}_{x|\mathbf{w}=\pm\mathbf{u}_k}].\end{aligned}$$

Because the eigenvalues of $\mathbf{D_w}(\pm\mathbf{u}_k)$ are $-2\lambda_k$, $(\lambda_i - \lambda_k)_{i\neq k}$ [resp., $2\lambda_k$, $-(\lambda_i-\lambda_k)_{i\neq k}$], these eigenvalues are all real negative for $k = 1$ only, for the stochastic approximation algorithms (42.4.1), in contrast to the stochastic approximation algorithms (42.4.3) for which $\mathbf{D_w}(\pm\mathbf{u}_k)$ has at least one nonnegative eigenvalue. Consequently only $\pm\mathbf{u}_1$ is locally asymptotically stable for the ODE associated with (42.4.1) and all the eigenvectors $(\pm\mathbf{u}_k)_{k=1,..,n}$ are unstable for the ODE associated with (42.4.3) and thus only (42.4.1) (Oja's neuron for dominant eigenvector) can be retained.

Note that the coupled stochastic approximation algorithms (42.4.1)(42.4.2) can be globally written as (42.7.1) as well. The associated ODE, given by

$$\frac{d}{dt}\begin{pmatrix}\mathbf{w}(t)\\ \lambda(t)\end{pmatrix} = \begin{pmatrix}\mathbf{C}_x\mathbf{w} - \mathbf{w}\mathbf{w}^T\mathbf{C}_x\mathbf{w}\\ \mathbf{w}^T\mathbf{C}_x\mathbf{w} - \lambda\end{pmatrix} \tag{42.7.9}$$

has the pairs $(\pm\mathbf{u}_k, \lambda_k)_{k=1,\ldots n}$ as stationary points. The derivative $\mathbf{D}$ of the mean field at theses points is given by

$$\mathbf{D} = \begin{pmatrix}\mathbf{D_w}(\pm\mathbf{u}_k) & \mathbf{0}\\ 2\mathbf{u}_k^T\mathbf{C}_x & -1\end{pmatrix}$$

whose eigenvalues are $-2\lambda_k$, $(\lambda_i - \lambda_k)_{i\neq k}$ and -1. Consequently the pair $(\pm\mathbf{u}_1, \lambda_1)$ is the only locally asymptotically stable point for the associated ODE (42.7.9) as well.

More precisely, it is proved in [49] that if $\mathbf{w}(0)^T\mathbf{u}_1 > 0$ [resp., < 0], the solution $\mathbf{w}(t)$ of the ODE (42.7.2) tends exponentially to $\mathbf{u}_1$ [resp., $-\mathbf{u}_1$] as $t \to \infty$. The pair $(\pm\mathbf{u}_1, \lambda_1)$ is thus globally asymptotically stable for the associated ODE.

Furthermore, using the stochastic approximation theory and in particular [43, th.2.3.1], it is proved in [50] that Oja's neuron (42.4.1) with decreasing step size μ_k, converges almost surely to $+\mathbf{u}_1$ or $-\mathbf{u}_1$ as k tends to ∞.

We have now the conditions to apply the Gaussian approximation results of Subsection 42.7.2. To solve the Lyapunov equation, the derivative $\mathbf{D}$ of the mean field at the pair

$(\pm\mathbf{u}_1, \lambda_1)$ is given by

$$\mathbf{D} = \begin{pmatrix} \mathbf{C}_x - \lambda_1 \mathbf{I}_n - 2\lambda_1 \mathbf{u}_1 \mathbf{u}_1^T & \mathbf{0} \\ 2\lambda_1 \mathbf{u}_1^T & -1 \end{pmatrix}.$$

In the case of independent Gaussian distributed data $\mathbf{x}(k)$, it has been proved ([22] [26]) that the covariance $\mathbf{G}$ (42.7.7) of the field is given by

$$\mathbf{G} = \begin{pmatrix} \mathbf{G}_{\mathbf{w}} & \mathbf{0} \\ \mathbf{0}^T & 2\lambda_1^2 \end{pmatrix}$$

with $\mathbf{G}_{\mathbf{w}} = \sum_{i=2}^n \lambda_1 \lambda_i \mathbf{u}_i \mathbf{u}_i^T$. Solving the Lyapunov equation (42.7.5), the following asymptotic covariance $\mathbf{C}_{\boldsymbol{\theta}}$ is obtained ([22][26])

$$\mathbf{C}_{\boldsymbol{\theta}} = \begin{pmatrix} \mathbf{C}_{\mathbf{w}} & \mathbf{0} \\ \mathbf{0}^T & \lambda_1^2 \end{pmatrix}$$

with $\mathbf{C}_{\mathbf{w}} = \sum_{i=2}^n \frac{\lambda_1 \lambda_i}{2(\lambda_1 - \lambda_i)} \mathbf{u}_i \mathbf{u}_i^T$. Consequently the estimates $(\mathbf{w}(k), \lambda(k))$ of $(\pm\mathbf{u}_1, \lambda_1)$ given by (42.4.1) and (42.4.2) respectively, are asymptotically independent and Gaussian distributed with

$$\mathrm{E}(\|\mathbf{w}(k) - (\pm\mathbf{u}_1)\|^2) \sim \sum_{i=2}^n \frac{\mu \lambda_1 \lambda_i}{2(\lambda_1 - \lambda_i)} \quad \text{and} \quad \mathrm{E}(\lambda(k) - \lambda_1)^2 \sim \mu \lambda_1^2.$$

We note that the behavior of the adaptive estimates $(\mathbf{w}(k), \lambda(k))$ of $(\pm\mathbf{u}_1, \lambda_1)$ are similar to the behavior of their batch estimates. More precisely if $\mathbf{w}(k)$ and $\lambda(k)$ denote now the dominant eigenvector and the associated eigenvalue of the sample estimate $\mathbf{C}(k) = \frac{1}{k} \sum_{i=1}^k \mathbf{x}(i) \mathbf{x}^T(i)$ of $\mathbf{C}_x$, a standard result [3, th.13.5.1, p.541]) gives

$$\sqrt{k}\ (\boldsymbol{\theta}(k) - \boldsymbol{\theta}_*) \xrightarrow{\mathcal{L}} \mathcal{N}(\mathbf{0}, \mathbf{C}_{\boldsymbol{\theta}}), \tag{42.7.10}$$

with $\mathbf{C}_{\boldsymbol{\theta}} = \begin{pmatrix} \mathbf{C}_{\mathbf{w}} & \mathbf{0} \\ \mathbf{0}^T & 2\lambda_1^2 \end{pmatrix}$ where $\mathbf{C}_{\mathbf{w}} = \sum_{i=2}^n \frac{\lambda_1 \lambda_i}{(\lambda_1 - \lambda_i)^2} \mathbf{u}_i \mathbf{u}_i^T$. The estimates $\mathbf{w}(k)$ and $\lambda(k)$ are asymptotically uncorrelated and the estimation of the eigenvalue λ_1 is well conditioned in contrast to those of the eigenvector $\mathbf{u}_1$ whose conditioning may be very bad when λ_1 and λ_2 are very close.

Expressions of the asymptotic bias $\lim_{k \to \infty} \mathrm{E}[\boldsymbol{\theta}(k)] - \boldsymbol{\theta}_*$ can be derived from (42.7.4). A word of caution is nonetheless necessary because the convergence of $\mu^{-1/2}\ (\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)$ to a limiting Gaussian distribution with covariance matrix $\mathbf{C}_{\boldsymbol{\theta}}$ does not guarantee the convergence of its moments to those of the limiting Gaussian distribution. In batch estimation, both the first and the second moments of the limiting distribution of $\sqrt{k}\,(\boldsymbol{\theta}(k) - \boldsymbol{\theta}_*)$ are equal to the corresponding asymptotic moments for independent Gaussian distributed data $\mathbf{x}(k)$. In the following, we assume the convergence of the second-order moments allowing us to write

$$\mathrm{E}[(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)]\,(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)^T] = \mu \mathbf{C}_{\boldsymbol{\theta}} + o(\mu).$$

Let $\boldsymbol{\theta}_\mu(k) = \boldsymbol{\theta}_* + \delta\boldsymbol{\theta}_k$ with $\boldsymbol{\theta}_* = \begin{pmatrix} \mathbf{u}_1 \\ \lambda_1 \end{pmatrix}$. Provided the data $\mathbf{x}(k)$ are independent (which implies that $\mathbf{w}(k)$ and $\mathbf{x}(k)\mathbf{x}^T(k)$ are independent) and $\boldsymbol{\theta}_\mu(k)$ is stationary, taking the expectation of both sides of (42.4.1) and (42.4.2) gives[11]

$$\begin{aligned}\mathrm{E}[\mathbf{C}_x(\mathbf{u}_1+\delta\mathbf{w}_k)-(\mathbf{u}_1+\delta\mathbf{w}_k)(\mathbf{u}_1+\delta\mathbf{w}_k)^T\mathbf{C}_x(\mathbf{u}_1+\delta\mathbf{w}_k)] &= \mathbf{0}\\ \mathrm{E}[(\mathbf{u}_1+\delta\mathbf{w}_k)^T\mathbf{C}_x(\mathbf{u}_1+\delta\mathbf{w}_k)-(\lambda_1+\delta\lambda_k)] &= 0.\end{aligned}$$

Using a second-order expansion, we get after some algebraic manipulations

$$\begin{bmatrix}\mathbf{C}_x-\lambda_1\mathbf{I}_n-2\lambda_1\mathbf{u}_1\mathbf{u}_1^T & \mathbf{0}\\ 2\lambda_1\mathbf{u}_1^T & -1\end{bmatrix}\begin{bmatrix}\mathrm{E}(\delta\mathbf{w}_k)\\ \mathrm{E}(\delta\lambda_k)\end{bmatrix} +\mu\begin{bmatrix}-(2\lambda_1\mathbf{C}_\mathbf{w}+\mathrm{Tr}(\mathbf{C}_x\mathbf{C}_\mathbf{w})\mathbf{I}_n)\mathbf{u}_1\\ \mathrm{Tr}(\mathbf{C}_x\mathbf{C}_\mathbf{w})\end{bmatrix}=o(\mu).$$

Solving this equation in $\mathrm{E}(\delta\mathbf{w}_k)$ and $\mathrm{E}(\delta\lambda_k)$ using the expression of $\mathbf{C}_\mathbf{w}$, gives the following expressions of the asymptotic bias

$$\mathrm{E}[\mathbf{w}(k)]-\mathbf{u}_1=-\mu\left(\sum_{i=2}^{n}\frac{\lambda_i^2}{4(\lambda_1-\lambda_i)}\right)\mathbf{u}_1+o(\mu) \quad \text{and} \quad \mathrm{E}[\lambda(k)]-\lambda_1=o(\mu).$$

We note that these asymptotic biases are similar to those obtained in batch estimation derived from a Taylor series expansion [76, p.68] with expression (42.7.10) of $\mathbf{C}_{\boldsymbol{\theta}}$.

$$\mathrm{E}[\mathbf{w}(k)]-\mathbf{u}_1=-\frac{1}{k}\left(\sum_{i=2}^{n}\frac{\lambda_1\lambda_i}{2(\lambda_1-\lambda_i)^2}\right)\mathbf{u}_1+o(\frac{1}{k}) \quad \text{and} \quad \mathrm{E}[\lambda(k)]-\lambda_1=o(\frac{1}{k}).$$

Finally, we see that in adaptive and batch estimation, the square of these biases are an order of magnitude smaller that the variances in $O(\mu)$ or $O(\frac{1}{k})$.

This methodology has been applied to compare the theoretical asymptotic performance of several adaptive algorithms for minor and principal component analysis in [22, 26]. For example, the asymptotic mean square error $\mathrm{E}(\|\mathbf{W}(k)-\mathbf{W}_*\|^2_{\mathrm{Fro}})$ of the estimate $\mathbf{W}(k)$ given by the WSA algorithm (42.6.9) is shown in Figure 1, where the step size μ is chosen to provide the same value for $\mu\mathrm{Tr}(\mathbf{C}_{\boldsymbol{\theta}})$. We clearly see in this figure that the value $\beta_2/\beta_1=0.6$ optimizes the asymptotic mean square error/speed of convergence tradeoff.

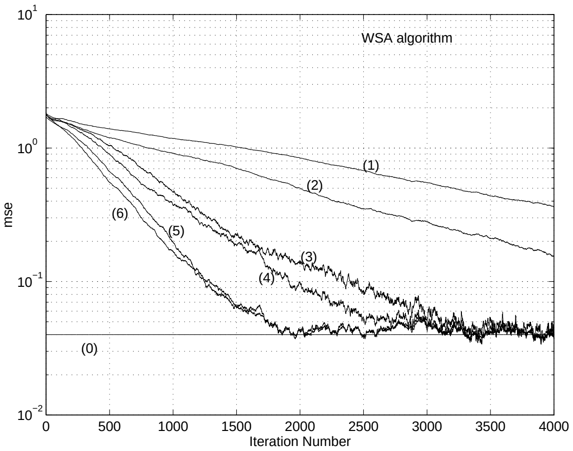

Figure 1 Learning curves of the mean square error $\mathrm{E}(\|\mathbf{W}(k)-\mathbf{W}_*\|^2_{\mathrm{Fro}})$ averaging 100 independent runs for the WSA algorithm, for different values of parameter $\beta_2/\beta_1=0.96$ (1), 0.9 (2), 0.1 (3), 0.2 (4), 0.4 (5) and 0.6 (6) compared with $\mu\mathrm{Tr}(\mathbf{C}_{\boldsymbol{\theta}})$ (0) in the case $n=4$, $r=2$, $\mathbf{C}_x=\mathrm{Diag}(1.75,1.5,0.5,0.25)$, where the entries of $\mathbf{W}(0)$ are chosen randomly uniformly in [0,1].

[11] We note that this derivation would not be possible for non-polynomial adaptations $f(\boldsymbol{\theta}(k),\mathbf{x}(k))$.

42.7.3.2 Convergence and performance analysis of Oja's algorithm Consider now Oja's algorithm (42.5.5) described in Subsection 42.5.1. A difficulty arises in the study of the behavior of $\mathbf{W}(k)$ because the set of orthonormal bases of the r-dominant subspace forms a *continuum* of attractors: the column vectors of $\mathbf{W}(k)$ do not in general tend to the eigenvectors $\mathbf{u}_1, \ldots, \mathbf{u}_r$, and we have no proof of convergence of $\mathbf{W}(k)$ to a particular orthonormal basis of their span. Thus, considering the asymptotic distribution of $\mathbf{W}(k)$ is meaningless. To solve this problem, in the same way as Williams [77] did when he studied the stability of the estimated projection matrix $\mathbf{P}(k) \stackrel{\text{def}}{=} \mathbf{W}(k)\mathbf{W}^T(k)$ in the dynamics induced by Oja's learning equation $\frac{d\mathbf{W}(t)}{dt} = [\mathbf{I}_n - \mathbf{W}(t)\mathbf{W}(t)^T]\mathbf{C}\mathbf{W}(t)$, viz

$$\frac{d\mathbf{P}(t)}{dt} = (\mathbf{I}_n - \mathbf{P}(t))\mathbf{C}\mathbf{P}(t) + \mathbf{P}(t)\mathbf{C}(\mathbf{I}_n - \mathbf{P}(t)), \tag{42.7.11}$$

we consider the trajectory of the matrix $\mathbf{P}(k) \stackrel{\text{def}}{=} \mathbf{W}(k)\mathbf{W}^T(k)$ whose dynamics are governed by the stochastic equation

$$\mathbf{P}(k+1) = \mathbf{P}(k) + \mu_k f(\mathbf{P}(k), \mathbf{x}(k)\mathbf{x}^T(k)) + \mu_k^2 h(\mathbf{P}(k), \mathbf{x}(k)\mathbf{x}^T(k)) \tag{42.7.12}$$

with $f(\mathbf{P}, \mathbf{C}) \stackrel{\text{def}}{=} (\mathbf{I}_n - \mathbf{P})\mathbf{C}\mathbf{P} + \mathbf{P}\mathbf{C}(\mathbf{I}_n - \mathbf{P})$ and $h(\mathbf{P}, \mathbf{C}) \stackrel{\text{def}}{=} (\mathbf{I}_n - \mathbf{P})\mathbf{C}\mathbf{P}\mathbf{C}(\mathbf{I}_n - \mathbf{P})$. A remarkable feature of (42.7.12) is that the field f and the complementary term h depend only on $\mathbf{P}(k)$ and *not* on $\mathbf{W}(k)$. This fortunate circumstance makes it possible to study the evolution of $\mathbf{P}(k)$ without determining the evolution of the underlying matrix $\mathbf{W}(k)$. The characteristics of $\mathbf{P}(k)$ are indeed the most interesting since they completely characterize the estimated subspace. Since (42.7.11) has a unique global asymptotically stable point $\mathbf{P}_* = \mathbf{\Pi}_s$ [68], we can conjecture from the stochastic approximation theory [13, 43] that (42.7.12) converges almost surely to $\mathbf{P}_*$. And consequently the estimate $\mathbf{W}(k)$ given by (42.5.5) converges almost surely to the signal subspace in the meaning recalled in Subsection 42.2.4.

To evaluate the asymptotic distributions of the subspace projection matrix estimator given by (42.7.12), we must adapt the results of Subsection 42.7.2 because the parameter $\mathbf{P}(k)$ is here an $n \times n$ rank-r symmetric matrix. Furthermore, we note that some eigenvalues of the derivative of the mean field $\bar{f}(\mathbf{P}) = \mathrm{E}[f(\mathbf{P}, \mathbf{x}(k)\mathbf{x}^T(k))]$ are positive real. To overcome this difficulty, let us now consider the following parametrization of $\mathbf{P}(k)$ in a neighborhood of $\mathbf{P}_*$ introduced in [24, 25]. If $\{\theta_{ij}(\mathbf{P}) | 1 \le i \le j \le n\}$ are the coordinates of $\mathbf{P} - \mathbf{P}_*$ in the orthonormal basis $(\mathbf{S}_{i,j})_{1 \le i \le j \le n}$ defined by

$$\mathbf{S}_{i,j} = \begin{cases} \mathbf{u}_i\mathbf{u}_i^T & i = j \\ \frac{\mathbf{u}_i\mathbf{u}_j^T + \mathbf{u}_j\mathbf{u}_i^T}{\sqrt{2}} & i < j \end{cases},$$

with the inner product under consideration is $(\mathbf{A}, \mathbf{B}) \stackrel{\text{def}}{=} \mathrm{Tr}(\mathbf{A}^T\mathbf{B})$, then,

$$\mathbf{P} = \mathbf{P}_* + \sum_{1 \le i,j \le n} \theta_{ij}\mathbf{P}\mathbf{S}_{i,j}$$

and $\theta_{ij}(\mathbf{P}) = \mathrm{Tr}\{\mathbf{S}_{ij}(\mathbf{P} - \mathbf{P}_*)\}$ for $1 \le i \le j \le n$. The relevance of this basis is shown by the following relation proved in [24, 25]

$$\mathbf{P} = \mathbf{P}_* + \sum_{(i,j) \in P_s} \theta_{ij}(\mathbf{P})\,\mathbf{S}_{ij} + O(\|\mathbf{P} - \mathbf{P}_*\|^2_{\mathrm{Fro}}), \tag{42.7.13}$$

where $P_s \stackrel{\text{def}}{=} \{(i,j) \mid 1 \leq i \leq j \leq n$ and $i \leq r\}$. There are $\frac{r}{2}(2n-r+1)$ pairs in P_s and this is exactly the dimension of the manifold of the $n \times n$ rank-r symmetric matrices. This point, together with relation (42.7.13), shows that the matrix set $\{\mathbf{S}_{ij} \mid (i,j) \in P_s\}$ is in fact an orthonormal basis of the tangent plane to this manifold at point $\mathbf{P}_*$. In other words, an $n \times n$ rank-r symmetric matrix $\mathbf{P}$ lying less than ϵ away from $\mathbf{P}_*$ (i.e., $\|\mathbf{P}-\mathbf{P}_*\| < \epsilon$) has negligible (of order ϵ^2) components in the direction of $\mathbf{S}_{ij}$ for $r < i \leq j \leq n$. It follows that, in a neighborhood of $\mathbf{P}_*$, the $n \times n$ rank-r symmetric matrices are uniquely determined by the $\frac{r}{2}(2n-r+1) \times 1$ vector $\boldsymbol{\theta}(\mathbf{P})$ defined by: $\boldsymbol{\theta}(\mathbf{P}) \stackrel{\text{def}}{=} \mathcal{S}^T \text{vec}(\mathbf{P}-\mathbf{P}_*)$, where $\mathcal{S}$ denotes the following $n^2 \times \frac{r}{2}(2n-r+1)$ matrix: $\mathcal{S} \stackrel{\text{def}}{=} [\ldots, \text{vec}(\mathbf{S}_{ij}), \ldots], \quad (i,j) \in P_s$. If $\mathcal{P}(\boldsymbol{\theta})$ denotes the unique (for $\|\boldsymbol{\theta}\|$ sufficiently small) $n \times n$ rank-r symmetric matrix such that $\mathcal{S}^T \text{vec}(\mathcal{P}(\boldsymbol{\theta}) - \mathbf{P}_*) = \boldsymbol{\theta}$, the following one-to-one mapping is exhibited for sufficiently small $\|\boldsymbol{\theta}(k)\|$:

$$\text{vec}(\mathcal{P}(\boldsymbol{\theta}(k))) = \text{vec}(\mathbf{P}_*) + \mathcal{S}\boldsymbol{\theta}(k) + O(\|\boldsymbol{\theta}(k)\|^2) \leftrightarrow \boldsymbol{\theta}(k) = \mathcal{S}^T \text{vec}(\mathbf{P}(k) - \mathbf{P}_*) \tag{42.7.14}$$

We are now in a position to solve the Lyapunov equation in the new parameter $\boldsymbol{\theta}$. The stochastic equation governing the evolution of $\boldsymbol{\theta}(k)$ is obtained by applying the transformation $\mathbf{P}(k) \rightarrow \boldsymbol{\theta}(k) = \mathcal{S}^T \text{vec}(\mathbf{P}(k) - \mathbf{P}_*)$ to the original equation (42.7.12), thereby giving

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \mu_k \phi(\boldsymbol{\theta}(k), \mathbf{x}(k)) + \mu_k^2 \psi(\boldsymbol{\theta}(k), \mathbf{x}(k)) \tag{42.7.15}$$

where $\phi(\boldsymbol{\theta}, \mathbf{x}) \stackrel{\text{def}}{=} \mathcal{S}^T \text{vec}(f(\mathcal{P}(\boldsymbol{\theta}), \mathbf{x}\mathbf{x}^T))$ and $\psi(\boldsymbol{\theta}, \mathbf{x}) \stackrel{\text{def}}{=} \mathcal{S}^T \text{vec}(h(\mathcal{P}(\boldsymbol{\theta}), \mathbf{x}\mathbf{x}^T))$. Solving now the Lyapunov equation associated with (42.7.15) after deriving the derivative of the mean field $\bar{\phi}(\boldsymbol{\theta})$ and the covariance of the field $\phi(\boldsymbol{\theta}(k), \mathbf{x}(k))$ for independent Gaussian distributed data $\mathbf{x}(k)$, yields the covariance $\mathbf{C}_{\boldsymbol{\theta}}$ of the asymptotic distribution of $\boldsymbol{\theta}(k)$. Finally using mapping (42.7.14), the covariance $\mathbf{C}_P = \mathcal{S}\mathbf{C}_{\boldsymbol{\theta}}\mathcal{S}^T$ of the asymptotic distribution of $\mathbf{P}(k)$ is deduced [25]

$$\mathbf{C}_P = \sum_{1 \leq i \leq r < j \leq n} \frac{\lambda_i \lambda_j}{2(\lambda_i - \lambda_j)} (\mathbf{u}_i \otimes \mathbf{u}_j + \mathbf{u}_j \otimes \mathbf{u}_i)(\mathbf{u}_i \otimes \mathbf{u}_j + \mathbf{u}_j \otimes \mathbf{u}_i)^T. \tag{42.7.16}$$

To improve the learning speed and misadjustment tradeoff of Oja's algorithm (42.5.5), it has been proposed in [25] to use the recursive estimate (42.3.6) for $\mathbf{C}_x(k) = \text{E}[\mathbf{x}(k)\mathbf{x}^T(k)]$. Thus the modified Oja's algorithm, called the smoothed Oja's algorithm, reads:

$$\begin{aligned} \mathbf{C}(k+1) &= \mathbf{C}(k) + \alpha\mu_k[\mathbf{x}(k)\mathbf{x}^T(k) - \mathbf{C}(k)], & (42.7.17) \\ \mathbf{W}(k+1) &= \mathbf{W}(k) + \mu_k[\mathbf{I}_n - \mathbf{W}(k)\mathbf{W}^T(k)]\mathbf{C}(k)\mathbf{W}(k), & (42.7.18) \end{aligned}$$

where α is introduced in order to normalize both algorithms because if the learning rate of (42.7.17) has no dimension, the learning rate of (42.7.18) must have the dimension of the inverse of the power of $\mathbf{x}(k)$. Furthermore α can take into account a better tradeoff between the misadjustments and the learning speed. Note that the performance derivations may be extended to this smoothed Oja's algorithm by considering that the coupled stochastic approximation algorithms (42.7.17)(42.7.18) can be globally written as (42.7.1) as well. Reusing now the parametrization $(\theta_{ij})_{1 \leq i \leq j \leq n}$ because $\mathbf{C}(k)$ is symmetric as well, and following the same approach, we obtain now [25]

$$\mathbf{C}_P = \sum_{1 \leq i \leq r < j \leq n} \frac{\alpha_{ij} \lambda_i \lambda_j}{2(\lambda_i - \lambda_j)} (\mathbf{u}_i \otimes \mathbf{u}_j + \mathbf{u}_j \otimes \mathbf{u}_i)(\mathbf{u}_i \otimes \mathbf{u}_j + \mathbf{u}_j \otimes \mathbf{u}_i)^T. \tag{42.7.19}$$

with $\alpha_{ij} \overset{\text{def}}{=} \alpha/(\alpha+\lambda_i-\lambda_j) < 1$.

This methodology has been applied to compare the theoretical asymptotic performance of several minor and principal subspace adaptive algorithms in [24, 25]. For example, the asymptotic mean square error $\mathrm{E}(\|\mathbf{P}(k)-\mathbf{P}_*\|^2_{\mathrm{Fro}})$ of the estimate $\mathbf{P}(k)$ given by the Oja's algorithm (42.5.5) and the smoothed Oja's algorithm (42.7.17) are shown in Figure 2, where the step size μ of the Oja's algorithm and the couple (μ, α) of the smoothed Oja's algorithm are chosen to provide the same value for $\mu\mathrm{Tr}(\mathbf{C}_P)$. We clearly see in this figure that the smoothed Oja's algorithm with $\alpha = 0.3$ provides faster convergence than the Oja's algorithm.

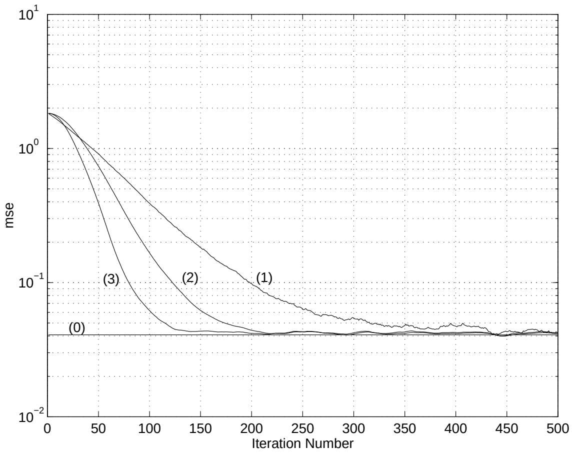

Figure 2 Learning curves of the mean square error $\mathrm{E}(\|\mathbf{P}(k)-\mathbf{P}_*\|^2_{\mathrm{Fro}})$ averaging 100 independent runs for the Oja's algorithm (1) and the smoothed Oja's algorithm with $\alpha = 1$ (2) and $\alpha = 0.3$ (3) compared with $\mu\mathrm{Tr}(\mathbf{C}_P)$ (0) in the same configuration $(\mathbf{C}_x, \mathbf{W}(0))$ that Figure 1.

Regarding the issue of asymptotic bias, note that there is a real methodological problem to apply the methodology of the end of Subsection 42.7.3.1. The trouble stems from the fact that the matrix $\mathbf{P}(k) = \mathbf{W}(k)\mathbf{W}^T(k)$ does not belong to a linear vector space because it is constrained to have fixed rank $r < n$. The set of such matrices is not invariant under addition; it is actually a smooth submanifold of $\mathcal{R}^{n\times n}$. This is not a problem in the first-order asymptotic analysis because this approach amounts to approximating this manifold by its tangent plane at a point of interest. This tangent plane is linear indeed. In order to refine the analysis by developing a higher order theory, it becomes necessary to take into account the curvature of the manifold. This is tricky business. As an example of these difficulties, one could show (under simple assumptions) that there exist no projection-valued estimators of a projection matrix that are unbiased at order $O(\mu)$; this can be geometrically pictured by representing the estimates as points on a curved manifold (here: the manifold of projection matrices).

Using a more involved expression of the covariance of the field (42.7.7), the previously described analysis can be extended to correlated data $\mathbf{x}(k)$. Expressions (42.7.16) and (42.7.19) extend provided that $\lambda_i\lambda_j$ is replaced by $\lambda_i\lambda_j + \lambda_{i,j}$ where $\lambda_{i,j}$ is defined in [25]. Note that when $\mathbf{x}(k) = (x_k, x_{k-1}, ..., x_{k-n+1})^T$ with x_k being an ARMA stationary process, the covariance of the field (42.7.7) and thus $\lambda_{i,j}$ can be expressed in closed form with the help of a finite sum [23].

The domain of learning rate μ for which the previously described asymptotic approach is valid and the performance criteria for which no analytical results could be derived from our first-order analysis, such as the speed of convergence and the deviation from orthonormality $d^2(\mu) \stackrel{\text{def}}{=} \|\mathbf{W}^T(k)\mathbf{W}(k) - \mathbf{I}_r\|^2_{\text{Fro}}$ can be derived from numerical experiments only. In order to compare Oja's and the smoothed Oja's algorithms, the associated parameters μ and (α, μ) must be constrained to give the same value of $\mu\text{Tr}(\mathbf{C}_P)$. In these conditions, it has been shown in [25] by numerical simulations that the smoothed Oja's algorithm provides faster convergence and a smaller deviation from orthonormality $d^2(\mu)$ than Oja's algorithm. More precisely, it has been shown that $d^2(\mu) \propto \mu^2$ [resp., $\propto \mu^4$] for Oja's [resp., the smoothed Oja's] algorithm. This result agrees with the presentation of Oja's algorithm given in Subsection 42.5.1 in which the term $O(\mu_k^2)$ was omitted from the orthonormalization of the columns of $\mathbf{W}(k)$.

Finally, using the theorem of continuity (e.g., [58, Th.6.2a]), note that the behavior of any differentiable function of $\mathbf{P}(k)$ can be obtained. For example, in DOA tracking from the MUSIC algorithm[12] (see e.g., Subsection 42.8.1), the MUSIC estimates $(\theta_i(k))_{i=1,...,r}$ of the DOAs at time k can be determined as the r deepest minima of the localization function $\mathbf{a}^H(\theta)[\mathbf{I}_n - \mathbf{P}(k)]\mathbf{a}(\theta)$. Using the mapping $\mathbf{P}(k) \longmapsto \boldsymbol{\theta}(k)$ where here $\boldsymbol{\theta}(k) \stackrel{\text{def}}{=} (\theta_1(k), ..., \theta_r(k))^T$, the Gaussian asymptotic distribution of the estimate $\boldsymbol{\theta}(k)$ can be derived [24] and compared to the batch estimate. For example for a single source, it has been proved [24] that

$$\text{Var}(\theta_1(k)) = \mu \frac{n\sigma_1^2}{2\alpha_1}\left(1 + \frac{\sigma_n^2}{n\sigma_1^2}\right)\frac{\sigma_n^2}{\sigma_1^2} + o(\mu).$$

where σ_1^2 is the source power and α_1 is a purely geometrical factor. Compared to the batch MUSIC estimate

$$\text{Var}(\theta_1(k)) = \frac{1}{k}\frac{1}{\alpha_1}\left(1 + \frac{\sigma_n^2}{n\sigma_1^2}\right)\frac{\sigma_n^2}{\sigma_1^2} + o(\frac{1}{k}),$$

the variances are similar provided $\mu n \sigma_1^2$ is replaced by $\frac{2}{k}$. This suggests that the step size μ of the adaptive algorithm must be normalized by $n\sigma_1^2$.

42.8 ILLUSTRATIVE EXAMPLES

Fast estimation and tracking of the principal (or minor) subspace or components of a sequence of random vectors is a major tool for parameter and signal estimation in many signal processing communications and RADAR applications (see e.g., [11] and the references therein). We can cite, for example, the Direction of Arrival (DOA) tracking and the blind channel estimation including CDMA and OFDM communications as illustrations.

Going back to the common observation model (42.3.1) introduced in Subsection 42.3.1

$$\mathbf{x}(k) = \mathbf{A}(k)\mathbf{r}(k) + \mathbf{n}(k), \tag{42.8.1}$$

where $\mathbf{A}(k)$ is an $n \times r$ full column rank matrix with $r < n$, the different applications are issued from specific deterministic parametrizations $\mathbf{A}(\boldsymbol{\phi}(k))$ of $\mathbf{A}(k)$ where $\boldsymbol{\phi}(k) \in \mathcal{R}^q$ is a slowly time-varying parameter compared to $\mathbf{r}(k)$. When this parametrization $\boldsymbol{\phi}(k) \longmapsto$

[12] Naturally in this application, the data are complex-valued, but using the conjugate transpose operator instead of transpose, and a complex parametrization based on the orthonormal basis $(\mathbf{H}_{i,j})_{1\leq i,j\leq n}$ where $\mathbf{H}_{i,j} = \mathbf{u}_i\mathbf{u}_i^H$ for $i = j$, $\frac{\mathbf{u}_i\mathbf{u}_j^H + \mathbf{u}_j\mathbf{u}_i^H}{\sqrt{2}}$ for $i < j$ and $\frac{\mathbf{u}_i\mathbf{u}_j^H - \mathbf{u}_j\mathbf{u}_i^H}{i\sqrt{2}}$ for $i > j$ instead of the orthonormal basis $(\mathbf{S}_{i,j})_{1\leq i\leq j\leq n}$, expressions (42.7.16) and (42.7.19) are still valid.

$\mathbf{A}(\boldsymbol{\phi}(k))$ is nonlinear, $\boldsymbol{\phi}(k)$ is assumed identifiable from the signal subspace $\mathrm{span}[\mathbf{A}(k)]$ or the noise subspace $\mathrm{null}[\mathbf{A}^T(k)]$ which is its orthogonal complement, i.e.,

$$\mathrm{span}\,[\mathbf{A}(\boldsymbol{\phi}(k))] = \mathrm{span}\,[\mathbf{A}(\boldsymbol{\phi}'(k))] \Rightarrow \boldsymbol{\phi}'(k) = \boldsymbol{\phi}(k),$$

and when this parametrization $\boldsymbol{\phi}(k) \longmapsto \mathbf{A}(\boldsymbol{\phi}(k))$ is linear, this identifiability is of course up to a multiplicative constant only.

42.8.1 Direction of arrival tracking

In the standard narrow-band array data model, $\mathbf{A}(\boldsymbol{\phi})$ is partitioned into r column vectors as $\mathbf{A}(\boldsymbol{\phi}) \stackrel{\mathrm{def}}{=} [\mathbf{a}(\boldsymbol{\phi}_1), \ldots, \mathbf{a}(\boldsymbol{\phi}_r)]$, where $(\boldsymbol{\phi}_i)_{i=1,\ldots,r}$ denotes different parameters associated with the r sources (azimuth, elevation, polarization,...). In this case, the parametrization is nonlinear. The simplest case corresponds to one parameter per source ($q = r$) (e.g., for a uniform linear array $\mathbf{a}(\phi_i) = (1, e^{i2\pi\frac{d}{\lambda}\sin\phi_i}, ..., e^{i2\pi\frac{d(n-1)}{\lambda}\sin\phi_i})^T$). For convenience and without loss of generality, we consider this case in the following. A simplistic idea to track the r DOAs would be to use an adaptive estimate $\widehat{\mathbf{\Pi}}_n(k)$ of the noise orthogonal projection matrix $\mathbf{\Pi}_n(k)$ given by $\mathbf{W}(k)\mathbf{W}^T(k)$ or $\mathbf{I}_n - \mathbf{W}(k)\mathbf{W}^T(k)$ where $\mathbf{W}(k)$ is, respectively, given by a minor or a dominant subspace adaptive algorithm introduced in Section 42.5,[13] and then to derive the estimated DOAs as the r minima of the cost function

$$\mathbf{a}^H(\phi)\widehat{\mathbf{\Pi}}_n(k)\mathbf{a}(\phi)$$

by a Newton-Raphson procedure

$$\begin{aligned}\phi_i(k+1) &= \phi_i(k)\\ &\quad - \frac{\Re[\mathbf{a}_i'^H(\phi(k))\widehat{\mathbf{\Pi}}_n(k+1)\mathbf{a}_i^H(\phi(k))]}{\Re[\mathbf{a}''^H(\phi_i(k))\widehat{\mathbf{\Pi}}_n(k+1)\mathbf{a}^H(\phi_i(k)) + \mathbf{a}'^H(\phi_i(k))\widehat{\mathbf{\Pi}}_n(k+1)\mathbf{a}'^H(\phi_i(k))]},\\ &\qquad i = 1, ..., r,\end{aligned}$$

where $\mathbf{a}_i' \stackrel{\mathrm{def}}{=} \frac{d\mathbf{a}_i}{d\phi}$ and $\mathbf{a}_i'' \stackrel{\mathrm{def}}{=} \frac{d^2\mathbf{a}_i}{d\phi^2}$. While this approach works for distant different DOAs, it breaks down when the DOAs of two or more sources are very close and particularly in scenarios involving targets with crossing trajectories. So the difficulty in DOA tracking is the association of the DOA estimated at different time points with the correct sources. To solve this difficulty, various algorithms for DOA tracking have been proposed in the literature (see e.g., [59] and the references therein). To maintain this correct association, a solution is to introduce the dynamic model governing the motion of the different sources

$$\boldsymbol{\phi}_i(k+1) \stackrel{\mathrm{def}}{=} \begin{bmatrix} \phi_i(k+1) \\ \phi_i'(k+1) \\ \phi_i''(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \phi_i(k) \\ \phi_i'(k) \\ \phi_i''(k) \end{bmatrix} + \begin{bmatrix} n_{1,i}(k) \\ n_{2,i}(k) \\ n_{3,i}(k) \end{bmatrix},$$

where T denotes the sampling interval and $\left(n_{j,i}(k)\right)_{j=1,2,3}$ are random process noise terms that account for random perturbations about the constant acceleration trajectory. This enables us to predict the state (position, velocity, and acceleration) of each source in any interval of time using the estimated state in the previous interval. An efficient and computationally simple heuristic procedure has been proposed in [65]. It consists of four

[13] Of course, adapted to complex-valued data.

steps by iteration k. First, a prediction $\widehat{\boldsymbol{\phi}}_i(k+1/k)$ of the state from the estimate $\widehat{\boldsymbol{\phi}}_i(k/k)$ is obtained. Second, an update of the estimated noise projection matrix $\widehat{\mathbf{\Pi}}_n(k)$ given by a subspace tracking algorithm introduced in Section 42.5 is derived from the new snapshot $\mathbf{x}(k)$. Third, for each source i, an estimate $\widehat{\phi}_i(k+1)$ given by a Newton-Raphson step initialized by the predicted DOA $\widehat{\phi}_i(k+1/k)$ given by a Kalman filter of the first step whose measurement equation is given by

$$\widehat{\phi}_i(k) = [1,0,0]\begin{bmatrix} \phi_i(k) \\ \phi'_i(k) \\ \phi''_i(k) \end{bmatrix} + n_{4,i}(k)$$

where the observation $\widehat{\phi}_i(k)$ is the DOA estimated by the Newton-Raphson step at iteration $k-1$. Finally, the DOA $\widehat{\phi}_i(k+1/k)$ predicted by the Kalman filter is also used to smooth the DOA $\widehat{\phi}_i(k+1)$ estimated by the Newton-Raphson step, to give the new estimate $\widehat{\boldsymbol{\phi}}_i(k+1/k+1)$ of the state whose its first component is used for tracking the r DOAs.

42.8.2 Blind channel estimation and equalization

In communication applications, the matched filtering followed by symbol rate sampling or oversampling yields an n-vector data $\mathbf{x}(k)$ which satisfies the model (42.8.1), where $\mathbf{r}(k)$ contains different transmitted symbols b_k. Depending on the context, (Single Input Multi Output (SIMO) channel, or Code Division Multiple Access (CDMA), Orthogonal Frequency Division Multiplexing (OFDM), Multi Carrier CDMA (MC CDMA) with or without intersymbol interference, different parametrizations of $\mathbf{A}(k)$ arise which are generally linear in the unknown parameter $\boldsymbol{\phi}(k)$. The latter represents different coefficients of the impulse response of the channel that are assumed slowly time-varying compared to the symbol rate. In these applications, two problems arise. First, the updating of the estimated parameters $\boldsymbol{\phi}(k)$, i.e., the adaptive identification of the channel can be useful to an optimal equalization based on an identified channel. Second, for particular models (42.8.1), a direct linear equalization $\mathbf{m}^T(k)\mathbf{x}(k)$ can be used from the adaptive estimation of the weight $\mathbf{m}(k)$. To illustrate subspace or component-based methods, two simple examples are given in the following.

For the two channel SIMO model, we assume that the two channels are of order m and that we stack the $m+1$ most recent samples of each channel to form the observed data $\mathbf{x}(k) = [\mathbf{x}_1(k), \mathbf{x}_2(k)]^T$. In this case we obtain the model (42.8.1) where $\mathbf{A}(k)$ is the following $2(m+1)\times(2m+1)$ Sylvester filtering matrix

$$\mathbf{A}(k) = \begin{pmatrix} \boldsymbol{\phi}_0(k) & \cdots & \cdots & \boldsymbol{\phi}_m(k) & & \\ & \ddots & & & \ddots & \\ & & \boldsymbol{\phi}_0(k) & & \cdots & \cdots & \boldsymbol{\phi}_m(k) \end{pmatrix},$$

and $\mathbf{r}(k) = (b_k, ..., b_{k-2m})^T$, with $\boldsymbol{\phi}_i(k) = (h_{i,1}(k), h_{i,2}(k))^T$, $i = 0, ..., m$ where $h_{i,j}$ represents the ith term of the impulse response of the j-th channel. These two channels do not share common zeros, guaranteeing their identifiability. In this specific two-channel case, the so called least square [79] and subspace [48] estimates of the impulse response $\boldsymbol{\phi}(k) = [\boldsymbol{\phi}_0^T(k), ..., \boldsymbol{\phi}_m^T(k)]^T$ defined up to a constant scale factor, coincide [80] and are given by $\boldsymbol{\phi}(k) = \mathbf{T}\mathbf{v}(k)$ with $\mathbf{v}(k)$ is the eigenvector associated with the unique smallest eigenvalue of $\mathbf{C}_x(k) = \mathrm{E}\left(\mathbf{x}(k)\mathbf{x}^T(k)\right)$ where $\mathbf{T}$ is the antisymmetric orthogonal matrix

$\mathbf{I}_{m+1} \otimes \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$. Consequently an adaptive estimation of the slowly time-varying impulse response $\boldsymbol{\phi}(k)$ can be derived from the adaptive estimation of the eigenvector $\mathbf{v}(k)$. Note that in this example, the rank r of the signal subspace is given by $r = 2m + 1$ whose order m of the channels that usually possess "small" leading and trailing terms is ill defined. For such channels it has been shown [45] that blind channel approximation algorithms should attempt to model only the significant part of the channel composed of the large impulse response terms because efforts toward modeling small leading and/or trailing terms lead to effective overmodeling, which is generically ill-conditioned and, thus, should be avoided. A detection procedure to detect the order of this significant part has been given in [44].

Consider now an asynchronous direct sequence CDMA system of r users without intersymbol interference. In this case, model (42.8.1) applies, where $\mathbf{A}(k)$ is given by

$$\mathbf{A}(k) = [a_1(k)\mathbf{s}_1, ..., a_r(k)\mathbf{s}_r]$$

where $a_i(k)$ and $\mathbf{s}_i$ are respectively the amplitude and the signature sequence of the i-th user and $\mathbf{r}(k) = (b_{k,1}, ..., b_{k,r})^T$ where $b_{k,i}$ is the symbol k of the i-th user. We assume that only the signature sequence of User 1, the user of interest, is known. Two linear multiuser detectors $\mathbf{m}^T(k)\mathbf{x}(k)$, namely, the decorrelation detector (i.e. that completely eliminates the multiple access interference caused by the other users) and the linear MMSE detector for estimating the symbol $b_{k,1}$, has been proposed in [75] in terms of the signal eigenvalues and eigenvectors. The scaled version of the respective weights $\mathbf{m}(k)$ of these detectors are given by

$$\begin{aligned} \mathbf{m}(k) &= \mathbf{U}_s(k)\left(\boldsymbol{\Delta}(k) - \sigma_n^2(k)\mathbf{I}_r\right)^{-1}\mathbf{U}_s^T(k)\mathbf{s}_1 \\ \mathbf{m}(k) &= \mathbf{U}_s(k)\boldsymbol{\Delta}^{-1}(k)\mathbf{U}_s^T(k)\mathbf{s}_1, \end{aligned}$$

where $\mathbf{U}_s(k) = [\mathbf{v}_1(k), ..., \mathbf{v}_r(k)]$, $\boldsymbol{\Delta}(k) = \mathrm{Diag}(\lambda_1(k), ..., \lambda_r(k))$ and $\sigma_n^2(k) = \lambda_{r+1}(k)$ issued from the adaptive EVD of $\mathbf{C}_x(k) = \mathrm{E}\left(\mathbf{x}(k)\mathbf{x}^T(k)\right)$ including the detection of the number r of user that can change by a rank tracking procedure (e.g., [71]).

42.9 CONCLUDING REMARKS

Although adaptive subspace and component-based algorithms were introduced in signal processing three decades ago, a rigorous convergence analysis has been only derived for the celebrated Oja's algorithm, whose Oja's neuron is a particular case, in stationary environment. In general all these techniques are derived heuristically from standard iterative computational techniques issued from numerical methods of linear algebra. So a theoretical convergence and performance analysis of these algorithms is necessary, but seem very challenging. Furthermore, such analysis is not sufficient because these algorithms may present numerical instabilities due to rounding errors. Consequently, a comprehensive comparison of the different algorithms that have appeared in the literature from the performance (convergence speed, mean square error, distance to the orthonormality, tracking capabilities), computational complexity and numerical stability points of view, that are out the scope of this chapter, would be be very useful for practitioners.

The interest of the signal processing community in adaptive subspace and component-based schemes remains strong as it is evident from the numerous articles and reports published in this area each year. But we note that these contributions mainly consist in the application of standard adaptive subspace and component-based algorithms in new applications and in refinements of well known subspace/component-based algorithms, principally to reduce their computational complexity and to numerically stabilize the minor subspace/component-based algorithms, whose literature is much more limited than the principal subspace and component-based algorithms.

Problems

42.1 Let λ_0 be a simple eigenvalue of a real symmetric $n \times n$ matrix $\mathbf{C}_0$, and let $\mathbf{u}_0$ be a unit 2-norm associated eigenvector, so that $\mathbf{C}\mathbf{u}_0 = \lambda_0 \mathbf{u}_0$. Then a real-valued function $\lambda(.)$ and a vector function $\mathbf{u}(.)$ are defined for all $\mathbf{C}$ in some neighborhood (e.g., among the real symmetric matrices) of $\mathbf{C}_0$ such that

$$\lambda(\mathbf{C}_0) = \lambda_0, \;\; \mathbf{u}(\mathbf{C}_0) = \mathbf{u}_0 \;\text{ and }\; \mathbf{C}\mathbf{u} = \lambda\mathbf{u} \;\text{ under the constraint }\; \|\mathbf{u}\|_2 = 1.$$

Using simple perturbations algebra manipulations, prove that the functions $\lambda(.)$ and $\mathbf{u}(.)$ are differentiable on some neighborhood of $\mathbf{C}_0$ and that the differentials at $\mathbf{C}_0$ are given by

$$\delta\lambda = \mathbf{u}_0^T(\delta\mathbf{C})\mathbf{u}_0 \quad \text{and} \quad \delta\mathbf{u} = -(\mathbf{C} - \lambda_0\mathbf{I}_n)^{\#}(\delta\mathbf{C})\mathbf{u}_0, \tag{42.9.2}$$

where $\#$ stands for the Moore Penrose inverse. Prove that if the constraint $\|\mathbf{u}\|_2 = 1$ is replaced by $\mathbf{u}_0^T\mathbf{u} = 1$, the differential $\delta\mathbf{u}$ given by (42.9.2) remains valid.

Now consider the same problem where $\mathbf{C}_0$ is a Hermitian matrix. To fix the perturbed eigenvector $\mathbf{u}$, the condition $\|\mathbf{u}\|^2 = 1$ is not sufficient. So suppose now that $\mathbf{u}_0^H\mathbf{u} = 1$. Note that in this case $\mathbf{u}$ no longer has unit 2-norm. Using the same approach as for the real symmetric case, prove that the functions $\lambda(.)$ and $\mathbf{u}(.)$ are differentiable on some neighborhood of $\mathbf{C}_0$ and that the differentials at $\mathbf{C}_0$ are now given by

$$\delta\lambda = \mathbf{u}_0^H(\delta\mathbf{C})\mathbf{u}_0 \quad \text{and} \quad \delta\mathbf{u} = -(\mathbf{C} - \lambda_0\mathbf{I}_n)^{\#}(\mathbf{I}_n - \mathbf{u}_0\mathbf{u}_0^H)(\delta\mathbf{C})\mathbf{u}_0. \tag{42.9.3}$$

In practice, different constraints are used to fix $\mathbf{u}$. For example, the SVD function of MATLAB forces all eigenvectors to be unit 2-norm with a real first element. Specify in this case the new expression of the differential $\delta\mathbf{u}$ given by (42.9.3). Finally, show that the differential $\delta\mathbf{u}$ given by (42.9.2) would be obtained with the condition $\mathbf{u}_0^H\delta\mathbf{u} = 0$, which is no longer derived from the constraint $\|\mathbf{u}\|_2 = 1$.

42.2 Consider an $n \times n$ real symmetric or complex Hermitian matrix $\mathbf{C}_0$ whose the r smallest eigenvalues are equal to σ^2 with $\lambda_{n-r} > \lambda_{n-r+1}$. Let $\mathbf{\Pi}_0$ the projection matrix onto the invariant subspace associated with σ^2. Then a matrix-valued function $\mathbf{\Pi}(.)$ is defined as the projection matrix onto the invariant subspace associated with the r smallest eigenvalues of $\mathbf{C}$ for all $\mathbf{C}$ in some neighborhood of $\mathbf{C}_0$ such that $\mathbf{\Pi}(\mathbf{C}_0) = \mathbf{\Pi}_0$. Using simple perturbations algebra manipulations, prove that the functions $\mathbf{\Pi}(.)$ is two times differentiable on some neighborhood of $\mathbf{C}_0$ and that the differentials at $\mathbf{C}_0$ are given by

$$\begin{aligned} \delta\mathbf{\Pi} \;\; &= \;\; -\Big(\mathbf{\Pi}_0(\delta\mathbf{C})\mathbf{S}_0^{\#} + \mathbf{S}_0^{\#}(\delta\mathbf{C})\mathbf{\Pi}_0\Big) \\ &+ \;\; \mathbf{S}_0^{\#}(\delta\mathbf{C})\mathbf{\Pi}_0(\delta\mathbf{C})\mathbf{S}_0^{\#} - \mathbf{\Pi}_0(\delta\mathbf{C})\mathbf{S}_0^{\#2}(\delta\mathbf{C})\mathbf{\Pi}_0 + \mathbf{S}_0^{\#}(\delta\mathbf{C})\mathbf{S}_0^{\#}(\delta\mathbf{C})\mathbf{\Pi}_0 \\ &+ \;\; \mathbf{\Pi}_0(\delta\mathbf{C})\mathbf{S}_0^{\#}(\delta\mathbf{C})\mathbf{S}_0^{\#} - \mathbf{S}_0^{\#2}(\delta\mathbf{C})\mathbf{\Pi}_0(\delta\mathbf{C})\mathbf{\Pi}_0 - \mathbf{\Pi}_0(\delta\mathbf{C})\mathbf{\Pi}_0(\delta\mathbf{C})\mathbf{S}_0^{\#2}, \end{aligned}$$

where $\mathbf{S}_0 \overset{\text{def}}{=} \mathbf{C}_0 - \sigma^2\mathbf{I}_n$.

42.3 Consider a Hermitian matrix $\mathbf{C}$ whose real and imaginary parts are denoted by $\mathbf{C}_r$ and $\mathbf{C}_i$ respectively. Prove that each eigenvalue eigenvector pair $(\lambda, \mathbf{u})$ of $\mathbf{C}$ is associated with the eigenvalue eigenvector pairs $(\lambda, \begin{pmatrix} \mathbf{u}_r \\ \mathbf{u}_i \end{pmatrix})$ and $(\lambda, \begin{pmatrix} -\mathbf{u}_i \\ \mathbf{u}_r \end{pmatrix})$ of the real symmetric matrix $\begin{bmatrix} \mathbf{C}_r & -\mathbf{C}_i \\ \mathbf{C}_i & \mathbf{C}_r \end{bmatrix}$ where $\mathbf{u}_r$ and $\mathbf{u}_i$ denote the real and imaginary parts of $\mathbf{u}$.

42.4 Consider what happens when the orthogonal iteration method (42.2.11) is applied with $r = n$ and under the assumption that all the eigenvalues of $\mathbf{C}$ are simple. The QR algorithm arises by considering how to compute the matrix $\mathbf{T}_i \overset{\text{def}}{=} \mathbf{W}_i^T \mathbf{C} \mathbf{W}_i$ directly from this predecessor $\mathbf{T}_{i-1}$. Prove that the following iterations

$$\begin{array}{rcl} \mathbf{T}_0 & = & \mathbf{Q}_0^T \mathbf{C} \mathbf{Q}_0 \;\; \text{where } \mathbf{Q}_0 \text{ is an arbitrary orthonormal matrix} \\ \text{for } i = 1,2,... \;\; \mathbf{T}_{i-1} & = & \mathbf{Q}_i \mathbf{R}_i \;\; \text{QR factorization} \\ \mathbf{T}_i & = & \mathbf{R}_i \mathbf{Q}_i, \end{array}$$

produce a sequence $(\mathbf{T}_i, \mathbf{Q}_0\mathbf{Q}_i ... \mathbf{Q}_i)$ that converges to $(\text{Diag}(\lambda_1, ..., \lambda_n), [\pm\mathbf{u}_1, ..., \pm\mathbf{u}_n])$.

42.5 Specify what happens to the convergence and the convergence speed, if the step $\mathbf{W}_i =$ orthonorm$\{\mathbf{C}\mathbf{W}_{i-1}\}$ of the orthogonal iteration algorithm (42.2.11) is replaced by the following $\{\mathbf{W}_i =$ orthonorm$\{(\mathbf{I}_n + \mu\mathbf{C})\mathbf{W}_{i-1}\}$. Same questions, for the step $\{\mathbf{W}_i =$ orthonormalization of $\mathbf{C}^{-1}\mathbf{W}_{i-1}\}$, then $\{\mathbf{W}_i =$ orthonormalization of $(\mathbf{I}_n - \mu\mathbf{C})\mathbf{W}_{i-1}\}$. Specify the conditions that must satisfy the eigenvalues of $\mathbf{C}$ and μ for these latter two steps. Examine the specific case $r = 1$.

42.6 Using the EVD of $\mathbf{C}$, prove that the solutions $\mathbf{W}$ of the maximizations and minimizations (42.2.7) are given by $\mathbf{W} = [\mathbf{u}_1, ..., \mathbf{u}_r]\mathbf{Q}$ and $\mathbf{W} = [\mathbf{u}_{n-r+1}, ..., \mathbf{u}_n]\mathbf{Q}$ respectively, where $\mathbf{Q}$ is an arbitrary $r \times r$ orthogonal matrix.

42.7 Consider the scalar function (42.2.14) $J(\mathbf{W}) \overset{\text{def}}{=} \text{E}(\|\mathbf{x} - \mathbf{W}\mathbf{W}^T\mathbf{x}\|^2)$ of $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_r]$ with $\mathbf{C} \overset{\text{def}}{=} \text{E}(\mathbf{x}\mathbf{x}^T)$. Let $\boldsymbol{\nabla}_\mathbf{W} = [\boldsymbol{\nabla}_1, ..., \boldsymbol{\nabla}_r]$ where $(\boldsymbol{\nabla}_k)_{k=1,..,r}$ is the gradient operator with respect to $\mathbf{w}_k$. Prove that

$$\boldsymbol{\nabla}_\mathbf{W} J = 2\left(-2\mathbf{C} + \mathbf{C}\mathbf{W}\mathbf{W}^T + \mathbf{W}\mathbf{W}^T\mathbf{C}\right)\mathbf{W}. \tag{42.9.4}$$

Then, prove that the stationary points of $J(\mathbf{W})$ are given by $\mathbf{W} = \mathbf{U}_r\mathbf{Q}$ where the r columns of $\mathbf{U}_r$ denote arbitrary r distinct unit-2 norm eigenvectors among $\mathbf{u}_1, ..., \mathbf{u}_n$ of $\mathbf{C}$ and where $\mathbf{Q}$ is an arbitrary $r \times r$ orthogonal matrix. Finally, prove that at each stationary point, $J(\mathbf{W})$ equals the sum of eigenvalues whose eigenvectors are not involved in $\mathbf{U}_r$.

Consider now the complex valued case where $J(\mathbf{W}) \overset{\text{def}}{=} \text{E}(\|\mathbf{x} - \mathbf{W}\mathbf{W}^H\mathbf{x}\|^2)$ with $\mathbf{C} \overset{\text{def}}{=} \text{E}(\mathbf{x}\mathbf{x}^H)$ and use the complex gradient operator (see e.g., [35]) defined by $\boldsymbol{\nabla}_\mathbf{W} = \frac{1}{2}[\boldsymbol{\nabla}_R + i\boldsymbol{\nabla}_I]$ where $\boldsymbol{\nabla}_R$ and $\boldsymbol{\nabla}_I$ denote the gradient operators with respect to the real and imaginary parts. Show that $\boldsymbol{\nabla}_\mathbf{W} J$ has the same form as the real gradient (42.9.4) except for a factor 1/2 and changing the transpose operator by the conjugate transpose one. By noticing that $\boldsymbol{\nabla}_\mathbf{W} J = \mathbf{O}$ is equivalent to $\boldsymbol{\nabla}_R J = \boldsymbol{\nabla}_I J = \mathbf{O}$, extend the previous results to the complex valued case.

42.8 With the notations of Exercice 42.7, suppose now that $\lambda_r > \lambda_{r+1}$ and consider first the real valued case. Show that the (i,j)th block $\boldsymbol{\nabla}_i \boldsymbol{\nabla}_j^T J$ of the block Hessian matrix $\mathbf{H}$ of $J(\mathbf{W})$ with respect to the nr-dimensional vector $[\mathbf{w}_1^T, ..., \mathbf{w}_r^T]^T$ is given by

$$\frac{1}{2}\boldsymbol{\nabla}_i \boldsymbol{\nabla}_j^T J \;\; = \;\; \delta_{ij}\left(-2\mathbf{C} + \mathbf{C}\mathbf{W}\mathbf{W}^T + \mathbf{W}\mathbf{W}^T\mathbf{C}\right)$$

$$+ \quad (\mathbf{w}_j^T\mathbf{C}\mathbf{w}_i)\mathbf{I}_n + (\mathbf{w}_j^T\mathbf{w}_i)\mathbf{C} + \mathbf{C}\mathbf{w}_j\mathbf{w}_i^T + \mathbf{w}_j\mathbf{w}_i^T\mathbf{C}.$$

After evaluating the EVD of the block Hessian matrix $\mathbf{H}$ at the stationary points $\mathbf{W} = \mathbf{U}_r\mathbf{Q}$, prove that $\mathbf{H}$ is nonnegative if $\mathbf{U}_r = [\mathbf{u}_1, ..., \mathbf{u}_r]$. Interpret in this case the zero eigenvalues of $\mathbf{H}$. Prove that when $\mathbf{U}_r$ contains an eigenvector different from $\mathbf{u}_1, ..., \mathbf{u}_r$, some eigenvalues of $\mathbf{H}$ are strictly negative. Deduce that all stationary points of $J(\mathbf{W})$ are saddle points except the points $\mathbf{W}$ whose associated matrix $\mathbf{U}_r$ contains the r dominant eigenvectors $\mathbf{u}_1, ..., \mathbf{u}_r$ of $\mathbf{C}$ which are global minima of the cost function (42.2.14).

Extend the previous results by considering the $2nr \times 2nr$ real Hessian matrix $\mathbf{H} = \boldsymbol{\nabla}\boldsymbol{\nabla}J$ with $\boldsymbol{\nabla} \stackrel{\text{def}}{=} [\boldsymbol{\nabla}_{R,1}^T, ..., \boldsymbol{\nabla}_{R,r}^T, \boldsymbol{\nabla}_{I,1}^T, ..., \boldsymbol{\nabla}_{I,r}^T]^T$.

42.9 With the notations of the NP3 algorithm described in Subsection 42.5.1.3, write (42.5.15) in the form

$$\mathbf{G}(k+1) = \frac{1}{\beta}\left[\mathbf{G}^{-1/2}(k)(\mathbf{I}_n + \mathbf{a}\mathbf{b}^T + \mathbf{b}\mathbf{a}^T + \alpha\mathbf{a}\mathbf{a}^T)\mathbf{G}^{-T/2}(k)\right]^{-1/2}$$

with $\mathbf{a} \stackrel{\text{def}}{=} \frac{1}{\beta}\mathbf{G}(k)\mathbf{y}(k)$, $\mathbf{b} \stackrel{\text{def}}{=} \frac{1}{\beta}\mathbf{G}(k)\mathbf{z}(k)$ and $\alpha \stackrel{\text{def}}{=} \|\mathbf{x}(k)\|^2$. Then, using the EVD $\nu_1\mathbf{e}_1\mathbf{e}_1^T + \nu_2\mathbf{e}_2\mathbf{e}_2^T$ of the symmetric rank two matrix $\mathbf{a}\mathbf{b}^T + \mathbf{b}\mathbf{a}^T + \alpha\mathbf{a}\mathbf{a}^T$, prove equalities (42.5.16) and (42.5.17) where $\tau_i \stackrel{\text{def}}{=} 1 - 1/\sqrt{\nu_i + 1}$, $i = 1, 2$.

42.10 Consider the following stochastic approximation algorithm derived from Oja's algorithm (42.5.5) where the sign of the step size can be reversed and where the estimate $\mathbf{W}(k)$ is forced to be orthonormal at each time step

$$\begin{aligned}\mathbf{W}'(k+1) &= \mathbf{W}(k) \pm \mu_k[\mathbf{I}_n - \mathbf{W}(k)\mathbf{W}^T(k)]\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k) \\ \mathbf{W}(k+1) &= \mathbf{W}'(k+1)[\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)]^{-1/2}, \qquad (42.9.5)\end{aligned}$$

where $[\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)]^{-1/2}$ denotes the symmetric inverse square root of $\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)$. To compute the later, use the updating equation of $\mathbf{W}'(k+1)$ and keeping in mind that $\mathbf{W}(k)$ is orthonormal, prove that $\mathbf{W}'^T(k+1)\mathbf{W}'(k+1) = \mathbf{I}_r \pm \mathbf{z}\mathbf{z}^T$ with $\mathbf{z} \stackrel{\text{def}}{=} \mu\|\mathbf{x}(k) - \mathbf{W}(k)\mathbf{y}(k)\|\mathbf{y}(k)$ where $\mathbf{y}(k) \stackrel{\text{def}}{=} \mathbf{W}^T(k)\mathbf{x}(k)$. Using identity (42.5.9), prove that $[\mathbf{W}'^T(k+1)\mathbf{W}'(k+1)]^{-1/2} = \mathbf{I}_r \pm \tau_k\mathbf{y}(k)\mathbf{y}^T(k)$ with $\tau_k \stackrel{\text{def}}{=} (1/\|\mathbf{y}(k)\|^2)\left((1/(1+\mu^2\|\mathbf{x}(k) - \mathbf{W}(k)\mathbf{y}(k)\|^2\|\mathbf{y}(k)\|^2)^{1/2}) - 1\right)$. Finally, using the update equation of $\mathbf{W}(k+1)$, prove that algorithm (42.9.5) leads to $\mathbf{W}(k+1) = \mathbf{W}(k) \pm \mu_k\mathbf{p}(k)\mathbf{y}^T(k)$ with $\mathbf{p}(k) \stackrel{\text{def}}{=} \pm\tau_k/\mu_k\mathbf{W}(k)\mathbf{y}(k) + (1 + \tau_k\|\mathbf{y}(k)\|^2)\left(\mathbf{x}(k) - \mathbf{W}(k)\mathbf{y}(k)\right)$.

Alternatively, prove that algorithm (42.9.5) leads to $\mathbf{W}(k+1) = \mathbf{H}(k)\mathbf{W}(k)$ where $\mathbf{H}(k)$ is the Householder transform given by $\mathbf{H}(k) = \mathbf{I}_n - 2\mathbf{u}(k)\mathbf{u}^T(k)$ where $\mathbf{u}(k) \stackrel{\text{def}}{=} \mathbf{p}(k)/\|\mathbf{p}(k)\|$.

42.11 Consider the scalar function (42.5.21) $J(\mathbf{W}) \stackrel{\text{def}}{=} \text{Tr}[\ln(\mathbf{W}^T\mathbf{C}\mathbf{W})] - \text{Tr}(\mathbf{W}^T\mathbf{W})$. Using the notations of Exercice 42.7, prove that

$$\boldsymbol{\nabla}_{\mathbf{W}}J = 2\left(\mathbf{C}\mathbf{W}(\mathbf{W}^T\mathbf{C}\mathbf{W})^{-1} - \mathbf{W}\right). \qquad (42.9.6)$$

Then, prove that the stationary points of $J(\mathbf{W})$ are given by $\mathbf{W} = \mathbf{U}_r\mathbf{Q}$ where the r columns of $\mathbf{U}_r$ denotes arbitrary r distinct unit-2 norm eigenvectors among $\mathbf{u}_1, ..., \mathbf{u}_n$ of $\mathbf{C}$ and where $\mathbf{Q}$ is an arbitrary $r \times r$ orthogonal matrix. Finally, prove that at each stationary point, $J(\mathbf{W}) = \sum_{i=1}^r \ln(\lambda_{s_i}) - r$, where the r eigenvalues λ_{s_i} are associated with the eigenvectors involved in $\mathbf{U}_r$.

42.12 With the notations of Exercice 42.11 and using the matrix differential method [46, Chap. 6], prove that the Hessian matrix $\mathbf{H}$ of $J(\mathbf{W})$ with respect to the nr-dimensional vector $[\mathbf{w}_1^T, ..., \mathbf{w}_r^T]^T$ is given by

$$\begin{aligned}\frac{1}{2}\mathbf{H} &= -\mathbf{I}_{nr} - (\mathbf{W}^T\mathbf{C}\mathbf{W})^{-1} \otimes [\mathbf{C}\mathbf{W}(\mathbf{W}^T\mathbf{C}\mathbf{W})^{-1}\mathbf{W}^T\mathbf{C}] \\ &- \mathbf{K}_{rn}[\mathbf{C}\mathbf{W}(\mathbf{W}^T\mathbf{C}\mathbf{W})^{-1}] \otimes [(\mathbf{W}^T\mathbf{C}\mathbf{W})^{-1}\mathbf{W}^T\mathbf{C}] + (\mathbf{W}^T\mathbf{C}\mathbf{W})^{-1} \otimes \mathbf{C},\end{aligned}$$

where $\mathbf{K}_{rn}$ is the $nr \times rn$ commutation matrix [46, Chap. 2]. After evaluating this Hessian matrix $\mathbf{H}$ at the stationnary points $\mathbf{W} = \mathbf{U}_r\mathbf{Q}$ of $J(\mathbf{W})$ (42.5.21), subtituting the EVD of $\mathbf{C}$ and deriving the EVD of $\mathbf{H}$, prove that when $\lambda_r > \lambda_{r+1}$, $\mathbf{H}$ is nonnegative if $\mathbf{U}_r = [\mathbf{u}_1, ..., \mathbf{u}_r]$. Interpret in this case the zero eigenvalues of $\mathbf{H}$. Prove that when $\mathbf{U}_r$ contains an eigenvector different from $\mathbf{u}_1, ..., \mathbf{u}_r$, some eigenvalues of $\mathbf{H}$ are strictly positive. Deduce that all stationary points of $J(\mathbf{W})$ are saddle points except the points $\mathbf{W}$ whose associated matrix $\mathbf{U}_r$ contains the r dominant eigenvectors $\mathbf{u}_1, ..., \mathbf{u}_r$ of $\mathbf{C}$ which are global maxima of the cost function (42.5.21).

42.13 Suppose the columns $[\mathbf{w}_1(k), ..., \mathbf{w}_r(k)]$ of the $n \times r$ matrix $\mathbf{W}(k)$ are orthonormal and let $\mathbf{W}'(k+1)$ be the matrix $\mathbf{W}(k) \pm \mu_k \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)$. If the matrix $\mathbf{S}(k+1)$ performs a Gram-Schmidt orthonormalization on the columns of $\mathbf{W}'(k+1)$, write this in explicit form for the columns of matrix $\mathbf{W}(k+1) = \mathbf{W}'(k+1)\mathbf{S}(k+1)$ as a power series expansion in μ_k and prove that

$$\begin{aligned}\mathbf{w}_i(k+1) &= \mathbf{w}_i(k) + \mu_k\left[\mathbf{I}_n - \mathbf{w}_i(k)\mathbf{w}_i^T(k) - 2\sum_{j=1}^{i-1}\mathbf{w}_j(k)\mathbf{w}_j^T(k)\right] \\ &\qquad \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}_i(k) + O(\mu_k^2) \text{ for } i = 1, \ldots, r.\end{aligned}$$

Following the same approach with now $\mathbf{W}'(k+1) = \mathbf{W}(k) \pm \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{W}(k)\mathbf{\Gamma}(k)$ where $\mathbf{\Gamma}(k) = \mu_k\mathrm{Diag}(1, \alpha_2, \ldots, \alpha_r)$, prove that

$$\begin{aligned}\mathbf{w}_i(k+1) &= \mathbf{w}_i(k) + \alpha_i\mu_k\left[\mathbf{I}_n - \mathbf{w}_i(k)\mathbf{w}_i^T(k) - \sum_{j=1}^{i-1}(1 + \frac{\alpha_j}{\alpha_i})\mathbf{w}_j(k)\mathbf{w}_j^T(k)\right] \\ &\qquad \mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}_i(k) + O(\mu_k^2) \text{ for } i = 1, \ldots, r.\end{aligned}$$

42.14 Specify the stationary points of the ODE associated with algorithm (42.6.10). Using the eigenvalues of the derivative of the mean field of this algorithm, prove that if $\lambda_{n-r+1} < 1$ and and $\beta > \frac{\lambda_{n-r+1}}{\lambda_n} - 1$, the only asymptotically stable points of the associated ODE are the eigenvectors $\pm\mathbf{v}_{n-r+1}, \ldots, \pm\mathbf{v}_n$.

42.15 Prove that the set of the $n \times r$ orthogonal matrices $\mathbf{W}$ (denoted the Stiefel manifold $\mathbf{St}_{n,r}$) is given by the set of matrices of the form $e^{\mathbf{A}}\mathbf{W}$ where $\mathbf{W}$ is an arbitrary $n \times r$ fixed orthogonal matrix and $\mathbf{A}$ is a skew-symmetric matrix ($\mathbf{A}^T = -\mathbf{A}$).

Prove the following relation

$$J(\mathbf{W} + \delta\mathbf{W}) = J(\mathbf{W}) + \mathrm{Tr}[\delta\mathbf{A}^T(\mathbf{H}_2\mathbf{W}\mathbf{H}_1\mathbf{W}^T - \mathbf{W}\mathbf{H}_1\mathbf{W}^T\mathbf{H}_2)] + o(\delta\mathbf{W}),$$

where $J(\mathbf{W}) = \mathrm{Tr}[\mathbf{W}\mathbf{H}_1\mathbf{W}^T\mathbf{H}_2]$ (where $\mathbf{H}_1$ and $\mathbf{H}_2$ are arbitrary $r \times r$ and $n \times n$ symmetric matrices) defined on the set of $n \times r$ orthogonal matrices. Then, give the differential dJ of the cost function $J(\mathbf{W})$ and deduce the gradient of $J(\mathbf{W})$ on this set of $n \times r$ orthogonal matrices

$$\boldsymbol{\nabla}_{\mathbf{W}} J = [\mathbf{H}_2\mathbf{W}\mathbf{H}_1\mathbf{W}^T - \mathbf{W}\mathbf{H}_1\mathbf{W}^T\mathbf{H}_2]\mathbf{W}. \tag{42.9.7}$$

42.16 Prove that if $\bar{f}(\boldsymbol{\theta}) = -\boldsymbol{\nabla}_{\boldsymbol{\theta}} J$, where $J(\boldsymbol{\theta})$ is a positive scalar function, $J(\boldsymbol{\theta}(t))$ tends to a constant as t tends to ∞, and consequently all the trajectories of the ODE (42.7.2) converge to the set of the stationary points of the ODE.

42.17 Let $\boldsymbol{\theta}_*$ be a stationary point of the ODE (42.7.2). Consider a Taylor series expansion of $\bar{f}(\boldsymbol{\theta})$ about the point $\boldsymbol{\theta} = \boldsymbol{\theta}_*$

$$\bar{f}(\boldsymbol{\theta}) = \bar{f}(\boldsymbol{\theta}_*) + \frac{d\bar{f}(\boldsymbol{\theta})}{d\boldsymbol{\theta}}_{|\boldsymbol{\theta}=\boldsymbol{\theta}_*}(\boldsymbol{\theta} - \boldsymbol{\theta}_*) + O[(\boldsymbol{\theta} - \boldsymbol{\theta}_*)](\boldsymbol{\theta} - \boldsymbol{\theta}_*).$$

By admitting that the behavior of the trajectory $\boldsymbol{\theta}(t)$ of the ODE (42.7.2) in the neighborhood of $\boldsymbol{\theta}_*$ is identical to those of the associated linearized ODE $\frac{d\boldsymbol{\theta}(t)}{dt} = \mathbf{D}\,(\boldsymbol{\theta}(t) - \boldsymbol{\theta}_*)$ (with $\mathbf{D} \overset{\text{def}}{=} \frac{d\bar{f}(\boldsymbol{\theta})}{d\boldsymbol{\theta}}_{|\boldsymbol{\theta}=\boldsymbol{\theta}_*}$) about the point $\boldsymbol{\theta}_*$, relate the stability of the stationary point $\boldsymbol{\theta}_*$ to the behavior of the eigenvalues of the matrix $\mathbf{D}$.

42.18 Consider the general stochastic approximation algorithm (42.7.1) in which the field $f(\boldsymbol{\theta}(k), \mathbf{x}(k)\mathbf{x}^T(k))$ and the residual perturbation term $h(\boldsymbol{\theta}(k), \mathbf{x}(k)\mathbf{x}^T(k))$ depend on the data $\mathbf{x}(k)$ through $\mathbf{x}(k)\mathbf{x}^T(k)$ and are linear in $\mathbf{x}(k)\mathbf{x}^T(k)$. The data $\mathbf{x}(k)$ are independent. The estimated parameter is here denoted $\boldsymbol{\theta}_\mu(k) \overset{\text{def}}{=} \boldsymbol{\theta}_* + \delta\boldsymbol{\theta}_k$. We suppose that the Gaussian approximation result (42.7.4) applies and that the convergence of the second-order moments allows us to write $\mathrm{E}[(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)]\,(\boldsymbol{\theta}_\mu(k) - \boldsymbol{\theta}_*)^T] = \mu\mathbf{C}_{\boldsymbol{\theta}} + o(\mu)$. Taking the expectation of both sides of (42.7.1), provided $\mu_k = \mu$ and $\boldsymbol{\theta}_\mu(k)$ stationary, gives that

$$\mathbf{0} = \mathrm{E}(f(\boldsymbol{\theta}_* + \delta\boldsymbol{\theta}_k, \mathbf{C}_x) = \frac{\partial f}{\partial\boldsymbol{\theta}}_{|\boldsymbol{\theta}=\boldsymbol{\theta}_*}\mathrm{E}(\delta\boldsymbol{\theta}_k) + \frac{\mu}{2}\frac{\partial^2 f}{\partial\boldsymbol{\theta}^2}_{|\boldsymbol{\theta}=\boldsymbol{\theta}_*}\mathrm{vec}(\mathbf{C}_{\boldsymbol{\theta}}) + o(\mu).$$

Deduce a general expression of the asymptotic bias $\lim_{k\to\infty}\mathrm{E}[\boldsymbol{\theta}(k)] - \boldsymbol{\theta}_*$.

REFERENCES

1. K. Abed Meraim, A. Chkeif, and Y. Hua, “Fast orthogonal PAST algorithm,” *IEEE Signal Process. letters*, vol. 7, no. 3, pp. 60-62, March 2000.
2. K. Abed Meraim, S. Attallah, A. Chkeif, and Y. Hua, “Orthogonal Oja algorithm,” *IEEE Signal Process. letters*, vol. 7, no. 5, pp. 116-119, May 2000.
3. T.W. Anderson, *An introduction to multivariate statistical analysis*, Second Edition, Wiley and Sons, 1984.
4. S. Attallah and K. Abed Meraim, “Fast algorithms for subspace tracking,” *IEEE Signal Process. letters*, vol. 8, no. 7, pp. 203-206, July 2001.
5. S. Attallah and K. Abed Meraim, “Low cost adaptive algorithm for noise subspace estimation,” *Electron. letters*, vol. 48, no. 12, pp. 609-611, June 2002.
6. S. Attallah, “The generalized Rayleigh’s quotient adaptive noise subspace algorithm: a Householder transformation-based implementation,” *IEEE Trans. on circ. and syst. II*, vol. 53, no. 81, pp. 3-7, January 2006.

7. S. Attallah, J.H. Manton and K. Abed Meraim, “Convergence analysis of the NOJA algorithm using the ODE approach,” *Signal Processing*, vol. 86, pp. 3490-3495, 2006.
8. R. Badeau, B. David and G. Richard, “Fast approximated power iteration subspace tracking,” *IEEE Trans. on Signal Process.*, vol. 53, no. 8, pp. 2931-2941, October 2005.
9. S. Bartelmaos, K. Abed Meraim and S. Attallah, “Fast algorithms for minor component analysis,” *Proc. ICASSP*, Philadelphia, March 2005.
10. S. Bartelmaos, “Subspace tracking and mobile localization in UMTS,” *PhD Thesis, Telecom Paris and University Pierre et Marie Curie*, France, 2008.
11. S. Beau and S. Marcos, “Range dependent clutter rejection using range recursive space time adaptive processing (STAP) algorithms,” Submitted to *Signal Processing*, March 2008.
12. R. Bellman, *Stability theory of differential equations*, McGraw Hill, New York, 1953.
13. A. Benveniste, M. Métivier and P. Priouret, *Adaptive algorithms and stochastic approximations*, Springer Verlag, New York, 1990.
14. A. Cantoni and P. Butler, “Eigenvalues and eigenvectors of symmetric centrosymmetric matrices,” *Linear Algebra and its Applications* 13, pp. 275-288, 1976.
15. T. Chen, “Modified Oja’s algorithms for principal subspace and minor subspace extraction,” *Neural Processing letters*, vol. 5, pp. 105-110, 1997.
16. T. Chen, Y. Hua and W.Y. Yan, “Global convergence of Oja’s subspace algorithm for principal component extraction,” *IEEE Trans. on Neural Networks*, vol. 9, no. 1, pp. 58-67, January 1998.
17. T. Chen and S. Amari, “Unified stabilization approach to principal and minor components extraction algorithms,” *Neural Networks*, vol. 14, no. 10, pp. 1377-1387, 2001.
18. T. Chonavel, B. Champagne and C. Riou, “Fast adaptive eigenvalue decomposition: a maximum likelihood approach,” *Signal Processing*, vol. 83, pp. 307-324, 2003.
19. A. Cichocki and S. Amari, *Adaptive blind signal and image processing, learning algorithms and applications*, John Wiley and Sons, 2002.
20. P. Comon and G.H. Golub, “Tracking a few extreme singular values and vectors in signal processing,” *Proc. IEEE*, vol. 78, no. 8, pp. 1327-1343, Aug. 1990.
21. J. Dehaene, *Continuous-time matrix algorithms, systolic algorithms and adaptive neural networks*, Ph.D. dissertation, Katholieke Univ. Leuven, Belgium, Oct. 1995.
22. J.P. Delmas and F. Alberge, “Asymptotic performance analysis of subspace adaptive algorithms introduced in the neural network literature,” *IEEE Trans. on Signal Process.*, vol. 46, no. 1, pp. 170-182, January 1998.
23. J.P. Delmas, “Performance analysis of a Givens parameterized adaptive eigenspace algorithm,” *Signal Processing*, vol. 68, no. 1, pp. 87-105, July 1998.
24. J.P. Delmas and J.F. Cardoso, “Performance analysis of an adaptive algorithm for tracking dominant subspace,” *IEEE Trans. on Signal Process.*, vol. 46, no. 11, pp. 3045-3057, November 1998.
25. J.P. Delmas and J.F. Cardoso, “Asymptotic distributions associated to Oja’s learning equation for Neural Networks,” *IEEE Trans. on Neural Networks*, vol. 9, no. 6, pp. 1246-1257, November 1998.
26. J.P. Delmas, “On eigenvalue decomposition estimators of centro-symmetric covariance matrices,” *Signal Processing*, vol. 78, no. 1, pp. 101-116, October 1999.
27. S.C. Douglas, S.Y. Kung and S. Amari, “A self-stabilized minor subspace rule,” *IEEE Signal Process. letters*, vol. 5, no. 12, pp. 328-330, December 1998.
28. X.G. Doukopoulos, “Power techniques for blind channel estimation in wireless communications systems, *PhD Thesis, IRISA-INRIA*, University of Rennes, France, 2004.

29. X.G. Doukopoulos and G.V. Moustakides, “The fast data projection method for stable subspace tracking,” *Proc. 13th European Signal Proc. Conf.*, Antalya, Turkey, September 2005.

30. X.G. Doukopoulos and G.V. Moustakides, “Fast and Stable Subspace Tracking,” *IEEE Trans. on Signal Process*, vol. 56, no. 4, pp. 1452-1465, April 2008.

31. J.C. Fort and G. Pagès, “Sur la convergence presque sure d’algorithmes stochastiques: le théoreme de Kushner-Clark theorem revisité,” *Technical report*, University Paris 1, 1994. Preprint SAMOS.

32. J.C. Fort and G. Pagès, “Convergence of stochastic algorithms: from the Kushner and Clark theorem to the Lyapunov functional method,” *Advances in Applied Probability*, no. 28, pp. 1072-1094, December 1996.

33. B. Friedlander and A.J. Weiss, “On the second-order of the eigenvectors of sample covariance matrices,” *IEEE Trans. on Signal Process.*, vol. 46, no. 11, pp. 3136-3139, November 1998.

34. G.H. Golub and C.F. Van Loan, *Matrix computations*, 3rd Edition, the Johns Hopkins University Press, 1996.

35. S. Haykin, *Adaptive filter theory*, Englewoods Cliffs, NJ: Prentice Hall, 1991.

36. R.A. Horn and CR. Johnson, *Matrix analysis*, Cambridge University Press, 1985.

37. Y. Hua, Y. Xiang, T. Chen, K. Abed Meraim and Y. Miao, "A new look at the power method for fast subspace tracking,” *Digital Signal Process.*, vol. 9, no. 2, pp. 297-314, October 1999.

38. B.H. Juang, S.Y. Kung, and C.A. Kamm (Eds.), *Proc. IEEE Workshop on neural networks for signal processing*, Princeton, NJ, September 1991.

39. I. Karasalo, “Estimating the covariance matrix by signal subspace averaging,” *IEEE Trans. on on ASSP*, vol. 34, no.1, pp. 8-12, February 1986.

40. J. Karhunen and J. Joutsensalo, “Generalizations of principal componant analysis, optimizations problems, and neural networks,” *Neural Networks*, vol. 8, pp. 549-562, 1995.

41. S.Y. Kung and K.I. Diamantaras “Adaptive principal component extraction (APEX) and applications,” *IEEE Trans. on ASSP*, vol. 42, no. 5, pp. 1202-1217, May 1994.

42. H.J. Kushner and D.S. Clark, *Stochastic approximation for constrained and unconstrained systems*, Applied math. science, no. 26, Springer Verlag, New York, 1978.

43. H.J. Kushner, *Weak convergence methods and singular perturbed stochastic control and filtering problems*, vol. 3 of Systems and Control: Foundations and applications, Birkhäuser, 1989.

44. A.P. Liavas, P.A. Regalia, J.P. Delmas, “Blind channel approximation: Effective channel order determination,” *IEEE Transactions on Signal Process.*, vol. 47, no. 12, pp. 3336-3344, December 1999

45. A.P. Liavas, P.A. Regalia, J.P. Delmas, “On the robustness of the linear prediction method for blind channel identification with respect to effective channel undermodeling / overmodeling,” *IEEE Transactions on Signal Process.*, vol. 48, no. 5, pp. 1477-1481, May 2000.

46. J.R. Magnus and H. Neudecker, *Matrix differential calculus with applications in statistics and econometrics*, Wiley series in probability and statistics, 1999.

47. Y. Miao and Y. Hua, “Fast subspace tracking and neural learning by a novel information criterion,” *IEEE Trans. on Signal Process.*, vol. 46, no. 7, pp. 1967-1979, July 1998.

48. E. Moulines, P. Duhamel, J.F. Cardoso and S. Mayrargue, “Subspace methods for the blind identification of multichannel FIR filters,” *IEEE Trans. Signal Process.*, vol. 43, no. 2, pp. 516-525, Feb. 1995.

49. E. Oja, “A simplified neuron model as a principal components analyzer,” *J. Math. Biol.*, vol. 15, pp. 267-273, 1982.

50. E. Oja and J. Karhunen, “On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix,” *J. Math. anal. Applications*, vol. 106, pp. 69-84, 1985.

51. E. Oja, *Subspace methods of pattern recognition*, Letchworth, England, Research Studies Press and John Wiley and Sons, 1983.

52. E. Oja, “Principal components, minor components and linear neural networks,” *Neural networks*, vol. 5, pp. 927-935, 1992.

53. E. Oja, H. Ogawa and J. Wangviwattana, “Principal component analysis by homogeneous neural networks, Part I: The weighted subspace criterion,” *IEICE Trans. Inform. and Syst.*, vol.E75-D, pp. 366-375, 1992.

54. E. Oja, H. Ogawa and J. Wangviwattana, “Principal component analysis by homogeneous neural networks, Part II: Analysis and extensions of the learning algorithms,” *IEICE Trans. Inform. Syst.*, vol.E75-D, pp. 376-382, 1992.

55. N. Owsley, “Adaptive data orthogonalization,” in *Proc. Conf. ICASSP*, pp. 109-112, 1978.

56. B.N. Parlett, *The symmetric eigenvalue problem*, Prentice Hall, Englewood Cliffs, N.J. 1980.

57. B. Picinbono, “Second-order complex random vectors and normal distributions,” *IEEE Trans. on Signal Process.*, vol. 44, no. 10, pp. 2637-2640, October 1996.

58. C.R. Rao, *Linear statistical inference and its applications*, New York, Wiley, 1973.

59. C.R Rao, C.R. Sastry and B. Zhou, “Tracking the direction of arrival of moving targets,” *IEEE Trans. on Signal Process.*, vol. 472, no. 5, pp. 1133-1144, May 1994.

60. P.A. Regalia, “An adaptive unit norm filter with applications to signal analysis and Karhunen Loéve tranformations,” *IEEE Trans. on Circuits and Systems*, vol. 37, no. 5, pp. 646-649, May 1990.

61. C. Riou, T. Chonavel and P.Y. Cochet, “Adaptive subspace estimation - Application to moving sources localization and blind channel identification,” in *Proc. ICASSP* Atlanta, GA, May 1996.

62. C. Riou, “Estimation adaptative de sous espaces et applications, *PhD Thesis, ENSTB*, University of Rennes, France, 1997.

63. H. Rutishauser, “Computational aspects of F.L. Bauer’s simultaneous iteration method,” *Numer. Math*, vol. 13, pp. 3-13, 1969.

64. H. Sakai and K. Shimizu, “A new adaptive algorithm for minor component analysis,” *Signal Processing*, vol. 71, pp. 301-308, 1998.

65. J. Sanchez-Araujo and S. Marcos, “A efficient PASTd algorithm implementation for multiple direction of arrival tracking,” *IEEE Trans. on Signal Process.*, vol. 47, no. 8, pp. 2321-2324, August 1999.

66. T.D. Sanger, “Optimal unsupervised learning in a single-layer linear feedforward network,” *Neural Networks*, vol. 2, pp. 459-473, 1989.

67. A.H. Sayed, *Fundamentals of adaptive filtering*, IEEE Press, Wiley-Interscience, 2003.

68. W.Y. Yan, U. Helmke, J.B. Moore, “Global analysis of Oja’s flow for neural networks,” *IEEE Trans. on Neural Networks*, vol. 5, no. 5, pp. 674-683, Sep. 1994.

69. J.F. Yang and M. Kaveh, “Adaptive eigensubspace algorithms for direction or frequency estimation and tracking,” *IEEE Trans. on ASSP*, vol. 36, no. 2, pp. 241-251, February 1988.

70. B. Yang, “Projection approximation subspace tracking,” *IEEE Trans. on Signal Process.*, vol. 43, no. 1, pp. 95-107, January 1995.

71. B. Yang, “An extension of the PASTd algorithm to both rank and subspace tracking,” *IEEE Signal Process. Letters*, vol. 2, no. 9, pp. 179-182, September 1995.

72. B. Yang, “Asymptotic convergence analysis of the projection approximation subspace tracking algorithms,” *Signal Processing*, vol. 50, pp. 123-136, 1996.

73. B. Yang and F. Gersemsky “Asymptotic distribution of recursive subspace estimators,” in *Proc. ICASSP* Atlanta, GA, May 1996.

74. L. Yang, S. Attallah, G. Mathew and K. Abed-Meraim, “Analysis of orthogonality error propagation for FRANS an HFRANS algorithms,” accepted to *IEEE Trans. on Signal Process.*, 2007.

75. X. Wang and H.V. Poor, “Blind multiuser detection,” *IEEE Trans. on Inform. Theory*, vol. 44, no. 2, pp. 677-689, March 1998.

76. J.H. Wilkinson, *The algebraic eigenvalue problem*, New York: Oxford University Press, 1965.

77. R. Williams, “Feature discovery through error-correcting learning,” *Technical Report 8501*, San Diego, CA: University of California, Institute of Cognitive Science, 1985.

78. L. Xu, E. Oja, C. Suen, “Modified Hebbian learning for curve and surface fitting,” *Neural Networks*, vol. 5, no.3, pp. 441-457, 1992.

79. G. Xu, H. Liu, L. Tong and T. Kailath, “A least-squares approach to blind channel identification,” *IEEE Trans. Signal Process.*, vol. 43, pp. 2982-2993, Dec. 1995.

80. H. Zeng and L. Tong, “Connections between the least squares and subspace approaches to blind channel estimation,” *IEEE Trans. Signal Process.*, vol. 44, pp. 1593-1596, June 1996.