# Asymptotic Distributions associated to Oja's Learning Equation for Neural Networks

Jean-Pierre Delmas, Jean-François Cardoso

# Asymptotic Distributions associated to Oja's Learning Equation for Neural Networks

Jean-Pierre Delmas * Jean-François Cardoso †

**Abstract**

In this paper, we perform a complete asymptotic performance analysis of the stochastic approximation algorithm (denoted Subspace Network Learning algorithm) derived from Oja's learning equation, in the case where the learning rate is constant and a large number of patterns is available. This algorithm drives the connection weight matrix $\mathbf{W}$ to an orthonormal basis of a dominant invariant subspace of a covariance matrix. Our approach consists in associating to this algorithm a second stochastic approximation algorithm that governs the evolution of $\mathbf{W}\mathbf{W}^T$ to the projection matrix onto this dominant invariant subspace. Then, using a general result of Gaussian approximation theory, we derive the asymptotic distribution of the estimated projection matrix. Closed form expressions of the asymptotic covariance of the projection matrix estimated by the SNL algorithm, and by the smoothed SNL algorithm that we introduce, are given in case of independent or correlated learning patterns and are further analyzed. It is found that the structures of these asymptotic covariance matrices are similar to those describing batch estimation techniques. The accuracy or our asymptotic analysis is checked by numerical simulations and it is found to be valid not only for a "small" learning rate but in a very large domain. Finally, improvements brought by our smoothed SNL algorithm are shown, such as the learning speed/misadjustment tradeoff and the deviation from orthonormality.

*Institut National des Télécommunications. 9 rue Charles Fourier, 91011 Evry Cedex, France. tel: (33)-01-60 76 46 32. fax: (33)-01-60 76 42 84. Email: delmas@int-evry.fr

†Ecole Nationale Supérieure des Télécommunications. 46 rue Barrault, 75634 Paris Cedex, France. tel: (33)-1-45 81 78 59. fax: (33)-1-45 88 79 35. Email: cardoso@sig.enst.fr

**Paper accepted to IEEE Transactions on Neural Networks**

# 1 Introduction

Over the past decade, adaptive estimation of subspaces of covariance matrices has been applied successfully in different fields of signal processing, such as high resolution spectral analysis and source localization, see [1] and the references therein, and more recently in the subspace approach used in blind identification of multichannel finite impulse response filters [2]. At the same time, and independently many neural network realizations have been proposed for the statistical technique of principal component analysis in data compression and feature extraction and for optimal fitting in the total least squares sense [3]. Among these realizations, several stochastic approximation algorithms have been proposed by many authors of the neural network community.

To understand the performance of these neural network unsupervised learning algorithms, it is of fundamental importance to investigate how they behave in the case where a large number of training samples is available. It was rigorously established for constant [4] and for decreasing [5] [3] learning rates that the behaviour of these algorithms is intimately related to the properties of an ordinary differential equation (ODE) which is obtained by suitably averaging over the training patterns. More precisely, if $\Theta_k$, $\mathbf{x}_k$ and $\gamma_k$ denote, respectively, the vector of network weights to be learned, the training patterns and the learning rate at time $k$, these stochastic approximation algorithms can be written in the form

$$\Theta_{k+1} = \Theta_k + \gamma_k f(\Theta_k, \mathbf{x}_k). \tag{1.1}$$

The key tool in the analysis of the sequence $\Theta_k$ is the so-called interpolated process $(\Theta_t), t \in \mathcal{R}^+$, usually defined by

$$\Theta(t) = \frac{t_{k+1} - t}{\gamma_{k+1}}\Theta_k + \frac{t - t_k}{\gamma_{k+1}}\Theta_{k+1}, \qquad t_k \leq t < t_{k+1} \tag{1.2}$$

where

$$t_0 = 0, \qquad t_k = \gamma_1 + \ldots + \gamma_k.$$

If $\gamma_k$ tends to zero at a suitable rate, the interpolated process of $\Theta_k$ eventually follows a trajectory which is a solution of the associated ODE with probability one [6], [7]. As such, the study of the local or global

stability of the equilibria of the ODE is of great importance [3]. If the sequence of learning rates is a small constant $\gamma$, the estimates $\Theta_k$ usually fail to stabilize, and the analysis of the interpolated processes cannot be carried out for fixed $\gamma$. Nevertheless, interesting asymptotic behavior may be obtained by letting $\gamma$ tend to 0 because for $\gamma$ "small enough," these algorithms will oscillate around the theoretical limit of the decreasing learning rate scheme. In particular the corresponding interpolated processes (1.2) converge weakly to the solution of the associated ODE [8] when $\gamma$ tends to 0. In practice, as $\gamma$ is necessarily small, the stochastic approximation algorithm (1.1) follows its associated ODE from the start in a first approximation. This transient phase is followed by an asymptotic phase where the random aspect of the fluctuations becomes prominent with respect to the evolution of the ODE. This second phase constitutes a second approximation. Naturally, if the learning rate $\gamma$ is chosen larger [resp. smaller], the learning speed increases [resp. decreases], but the fluctuations of the asymptotic phase increase [resp. decrease]. So a tradeoff naturally arises between the learning speed and the variances of the estimated network weights, often called misadjustment. In stationary random input environments, it is desirable to keep $\gamma$ large at the beginning, to achieve fast learning, and subsequently to decrease its value in order to reduce the variance of the estimates $\Theta_k$. So, it is of great importance to specify these variances. A good tool for evaluating these variances is a general Gaussian approximation result [9] which gives the limiting distribution of the estimates $\Theta_k$ when $k$ and $\gamma$ tend respectively to $+\infty$ and 0. The purpose of this paper is to determine the asymptotic distribution of the estimates by using the approach developed in [10], [11] [12] and [13], for two algorithms: the so-called SNL stochastic approximation algorithm [3], derived from Oja's learning equation, and the smoothed SNL algorithm that we introduce. However, since these stochastic approximation algorithms converge to any orthonormal basis of the considered eigenspace of the covariance matrix of the training patterns, and not to the eigenvectors themselves, we need to develop a special methodology, obtained by considering the stochastic approximation algorithm governed by the associated projection matrix.

This paper is organized as follows. In Section 2, we give an overview of Oja's learning equation and

of its associated stochastic approximation algorithm. Connections to very similar algorithms are enlightened and a modification of this stochastic approximation algorithm, denoted *smoothed* SNL algorithm, is introduced to improve the learning speed versus misadjustment tradeoff. In Section 3, after presenting a brief review of a general Gaussian approximation result, we consider the stochastic approximation algorithm that governs the associated projection matrix. This enables us to derive a closed form expression of the covariance of the limiting distribution of the projection matrix estimator computed by the SNL and by the smoothed SNL algorithms. These expressions are further analysed and compared to those obtained in batch estimation, and some by-products such as mean square errors are derived. The case of time-correlated training patterns is studied in Section 4. Finally we present in Section 5 some simulations with two purposes. On the one hand, we examine the accuracy of the expressions of the mean square error of the subspace projection matrix estimators and investigate the domain of learning rate for which our asymptotic approach is valid. On the other hand, we examine performance criteria for which no analytic results were obtained in the preceding sections. We thus show (by simulation) that the smoothed SNL algorithm is better than the SNL algorithm as concerns the learning speed/misadjusment tradeoff. Furthermore, it is showed that the deviation from orthonormality is proportional to $\gamma^2$ and to $\gamma^4$ for the SNL and the smoothed SNL algorithms, respectively.

The following notations are used in the paper. Matrices and vectors are represented by bold upper case and bold lower case characters, respectively. Vectors are by default in column orientation. $T$ stands transpose and $\mathbf{I}$ is the identity matrix. $\mathrm{E}(.), \mathrm{Cov}(.), \mathrm{Tr}(.)$ and $\|\,.\,\|_{\mathrm{Fro}}$ denote the expectation, the covariance, the trace operator and the Frobenius matrix norm, respectively. Vec(.) is the "vectorization" operator that turns a matrix into a vector consisting of the columns of the matrix stacked one below another and $\mathrm{Vec}^{-1}(.)$ is the inverse of the "vectorization" operator that turns an $n^2$-vector into an $n \times n$ matrix. They are used in conjunction with the Kronecker product $\mathbf{A} \otimes \mathbf{B}$ as the block matrix whose $(i, j)$ block element is $a_{i,j}\mathbf{B}$. For a projection matrix $\mathbf{P}$, $\mathbf{P}^{\perp}$ denotes the complementary projector $\mathbf{I} - \mathbf{P}$. $\mathrm{Diag}(a_1, \ldots, a_n)$ is a diagonal matrix consisting of the diagonal elements $a_i$. The symbol $1_A$ denotes

the indicator function of the condition $A$, which assumes the value 1 if the condition is satisfied and 0 otherwise.

## 2 The SNL and smoothed SNL algorithms

### 2.1 The algorithm associated to Oja's learning equation

For a given $n \times n$ covariance matrix $\mathbf{R}_x = E(\mathbf{x}\mathbf{x}^T)$ of a Gaussian distributed, zero mean real random training pattern vector $\mathbf{x}$, let $\lambda_1 \geq \ldots \geq \lambda_r > \lambda_{r+1} \geq \ldots \geq \lambda_n$ denote the eigenvalues of $\mathbf{R}_x$ and $\mathbf{v}_1, \ldots, \mathbf{v}_n$ the corresponding eigenvectors. We consider the recursive updating of an (approximately) orthonormal basis $\mathbf{W}_k$ of the $r$-dimensional dominant invariant subspace of $\mathbf{R}_x$. In neural networks, the integer $r$ stands for the number of neurons, $n$ the number of inputs and $\mathbf{W}_k$ the connection weight matrix.

The algorithm that we consider was introduced independently by Williams [14], Baldi [15] and Oja [16]. It was reformulated in [3] and [17] as a stochastic approximation counterpart of the "simultaneous iteration method" of numerical analysis [18]. This stochastic approximation algorithm reads:

$$\mathbf{W}'_{k+1} = \mathbf{W}_k + \gamma_k \mathbf{R}_k \mathbf{W}_k \tag{2.3}$$

$$\mathbf{W}_{k+1} = \mathbf{W}'_{k+1} \mathbf{S}_{k+1}^{-1} \tag{2.4}$$

in which $\mathbf{W}_k = (\mathbf{w}_{k,1}, \ldots, \mathbf{w}_{k,r}) \in \mathcal{R}^{n \times r}$ is a matrix whose columns $\mathbf{w}_{k,i} \in \mathcal{R}^n$ are orthonormal and approximate $r$ dominant eigenvectors of $\mathbf{R}_x$. We suppose that the learning rate sequence $\gamma_k$ satisfies the conditions:

$$\sum_{k=1}^{\infty} \gamma_k = +\infty \qquad \text{and} \qquad \lim_{k \to +\infty} \gamma_k = 0.$$

The matrix $\mathbf{R}_k$ in (2.3) is an estimate of the covariance matrix $\mathbf{R}_x$. In (2.4), $\mathbf{S}_{k+1}$ is a matrix depending on $\mathbf{W}'_{k+1}$ which orthonormalizes the columns of $\mathbf{W}'_{k+1}$. Depending on the form of $\mathbf{S}_{k+1}$ and on the choice of the estimate of $\mathbf{R}_k$, variants of the basic stochastic algorithm are obtained. In the algorithm that we consider, the instantaneous estimate $\mathbf{x}_k \mathbf{x}_k^T$ is used for $\mathbf{R}_k$ and the matrix $\mathbf{S}_{k+1}$ orthonormalizes

the columns of $\mathbf{W}'_{k+1}$ in (2.4) in a symmetrical way. Since $\mathbf{W}_k$ has orthonormal columns, for small $\gamma_k$ the columns of $\mathbf{W}'_{k+1}$ in (2.3) 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 $\mathbf{S}_{k+1} = (\mathbf{W}'^T_{k+1}\mathbf{W}'_{k+1})^{1/2}$. When, assuming $\gamma_k$ is small, $\mathbf{S}^{-1}_{k+1}$ is expanded and when the term $O(\gamma_k^2)$ is neglected from its expansion, the algorithm reads:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \gamma_k[\mathbf{I}_n - \mathbf{W}_k\mathbf{W}_k^T]\mathbf{x}_k\mathbf{x}_k^T\mathbf{W}_k. \tag{2.5}$$

The ODE associated to (2.5), called *Oja's learning equation*, enables us to study the convergence of the stochastic approximation algorithm (2.5). It reads:

$$\frac{d\mathbf{W}_t}{dt} = [\mathbf{I}_n - \mathbf{W}_t\mathbf{W}_t^T]\mathbf{R}_x\mathbf{W}_t. \tag{2.6}$$

If $r = 1$, in which case $\mathbf{W}_t$ is a vector, (2.5) gives the simplified neuron model of Oja [19] and $\pm\mathbf{v}_1$ is the only global asymptotically stable solution of (2.6). Furthermore, in [17], it is shown that if the algorithm (2.5) is used with uniformly bounded inputs $\mathbf{x}_k$, $\mathbf{W}_k$ remains inside some bounded subset. Thus, applying Kushner's ODE method [7], $\mathbf{W}_k$ converges almost surely either to $-\mathbf{v}_1$ or $+\mathbf{v}_1$ under these conditions. For $r > 1$, Oja conjectured in [17] similar properties: namely, $\mathbf{W}_k$ tends to an orthonormal basis of the eigenspace generated by $\mathbf{v}_1, \ldots, \mathbf{v}_r$. Following Oja's work, there has been considerable interest generated in understanding equation (2.6). For exemple, Baldi and Hornik [20] found the general form of equilibria $\mathbf{W} = [\mathbf{v}_{i_1}, \ldots, \mathbf{v}_{i_r}]\mathbf{Q}$ where $1 \leq i_1 < \ldots < i_r \leq n$ and $\mathbf{Q}$ is an orthogonal $r \times r$ matrix. Krogh and Hertz [21] examined the local properties of these equilibria and show that only $\mathbf{W} = [\mathbf{v}_1, \ldots, \mathbf{v}_r]\mathbf{Q}$ are locally stable. More recently it is proved in [22] that if $\mathbf{R}_x$ is positive definite and if the initial condition $\mathbf{W}_0$ is of rank $r$, the solution of (2.6) converges to an orthonormal basis of the $r$-dominant eigenspace of $\mathbf{R}_x$. Although this last result is a global asymptotic analysis of (2.6), the question of the theoretical study of the stochastic approximation algorithm (2.5) appears to be extremely challenging.

### 2.2 Connections with other algorithms

Written in the form $\mathbf{W}_{k+1} = \mathbf{W}_k + \gamma_k[\mathbf{x}_k\mathbf{x}_k^T - \mathbf{W}_k\mathbf{W}_k^T\mathbf{x}_k\mathbf{x}_k^T]\mathbf{W}_k$, the SNL algorithm is quite similar to the algorithm presented independently by Russo [23] and Yang [24] and further analyzed in [25]. This latter algorithm, which we will call the Yang algorithm, is a stochastic gradient algorithm based on the unconstrained minimization of $\mathrm{E}\| \mathbf{x}_k - \mathbf{W}\mathbf{W}^T\mathbf{x}_k \|_{\mathrm{Fro}}^2$, and it reads:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \gamma_k[2\mathbf{x}_k\mathbf{x}_k^T - \mathbf{x}_k\mathbf{x}_k^T\mathbf{W}_k\mathbf{W}_k^T - \mathbf{W}_k\mathbf{W}_k^T\mathbf{x}_k\mathbf{x}_k^T]\mathbf{W}_k, \tag{2.7}$$

in which the term between brackets is the symmetrization of the term $\mathbf{x}_k\mathbf{x}_k^T - \mathbf{W}_k\mathbf{W}_k^T\mathbf{x}_k\mathbf{x}_k^T$ of the SNL algorithm. In [24], it is shown that the Yang algorithm globally converges, almost surely, to the set of the orthonormal bases of the $r$-dominant invariant subspace of $\mathbf{R}_x$. Based on this observation, the matrix $\mathbf{W}_k^T\mathbf{W}_k$ that appears in (2.7) can be approximated by $\mathbf{I}_r$. We note in this case that the Yang algorithm gives the SNL algorithm. Connected to the SNL algorithm, Oja *et al* [26] proposed an algorithm denoted *Weighted Subspace Algorithm* (WSA) similar to the SNL algorithm (2.5) except for the diagonal matrix $\mathbf{\Delta} \stackrel{\mathrm{def}}{=} \mathrm{Diag}(\beta_1, \ldots, \beta_r)$. It reads:

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \gamma_k[\mathbf{I}_n - \mathbf{W}_k\mathbf{\Delta}^{-1}\mathbf{W}_k^T]\mathbf{x}_k\mathbf{x}_k^T\mathbf{W}_k\mathbf{\Delta}. \tag{2.8}$$

If $\beta_i = 1$ for all $i$, this algorithm reduces to the SNL algorithm. However, if all of them are chosen different and positive: $0 < \beta_1 < \ldots < \beta_r$, then it has been shown by Oja *et al* [27] that the eigenvectors $\pm\mathbf{v}_1, \ldots, \pm\mathbf{v}_r$ are the global asymptotically stable solutions of the ODE associated to (2.8). Thus Oja *et al* [27] conjectured that $\mathbf{w}_{k,1}, \ldots, \mathbf{w}_{k,r}$ converge almost surely to the eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$.

To improve the learning speed and misadjustment tradeoff, we propose in this paper to use the following recursive estimate for $\mathbf{R}_k$:

$$\mathbf{R}_{k+1} = \mathbf{R}_k + \gamma_k(\mathbf{x}_k\mathbf{x}_k^T - \mathbf{R}_k), \tag{2.9}$$

so that the modified SNL algorithm, which we call the *smoothed SNL algorithm*, reads:

$$\mathbf{R}_{k+1} = \mathbf{R}_k + \alpha\gamma_k(\mathbf{x}_k\mathbf{x}_k^T - \mathbf{R}_k), \tag{2.10}$$

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \gamma_k[\mathbf{I}_n - \mathbf{W}_k\mathbf{W}_k^T]\mathbf{R}_k\mathbf{W}_k. \tag{2.11}$$

$\alpha$ is introduced in order to normalize both algorithms because if the learning rate of eq.(2.10) has no dimension, the learning rate of eq.(2.11) must have the dimension of the inverse of the power of $\mathbf{x}_k$. Furthermore $\alpha$ can take into account a better tradeoff between the misadjustments and the learning speed, as we will see in section 5. We note that such a recursive estimator was introduced by Owsley [28] in his *Orthogonal Iteration* algorithm.

## 3 Asymptotic performance analysis

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 $\mathbf{v}_1, \ldots, \mathbf{v}_r$, 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 [14] did when he studied the stability of $\mathbf{P}_t \stackrel{\text{def}}{=} \mathbf{W}_t\mathbf{W}_t^T$ in the dynamics induced by Oja's learning equation (2.6), viz

$$\frac{d\mathbf{P}_t}{dt} = (\mathbf{I}_n - \mathbf{P}_t)\mathbf{R}_x\mathbf{P}_t + \mathbf{P}_t\mathbf{R}_x(\mathbf{I}_n - \mathbf{P}_t), \tag{3.1}$$

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

$$\mathbf{P}_{k+1} = \mathbf{P}_k + \gamma_k f(\mathbf{P}_k, \mathbf{x}_k\mathbf{x}_k^T) + \gamma_k^2 h(\mathbf{P}_k, \mathbf{x}_k\mathbf{x}_k^T) \tag{3.2}$$

with

$$f(\mathbf{P}, \mathbf{M}) \stackrel{\text{def}}{=} (\mathbf{I}_n - \mathbf{P})\mathbf{M}\mathbf{P} + \mathbf{P}\mathbf{M}(\mathbf{I}_n - \mathbf{P}) \tag{3.3}$$

$$h(\mathbf{P}, \mathbf{M}) \stackrel{\text{def}}{=} (\mathbf{I}_n - \mathbf{P})\mathbf{M}\mathbf{P}\mathbf{M}(\mathbf{I}_n - \mathbf{P}). \tag{3.4}$$

A remarkable feature of (3.2) 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 (3.1) has a unique global asymptotically stable point $\mathbf{P}_* \stackrel{\text{def}}{=} [\mathbf{v}_1, \ldots, \mathbf{v}_r][\mathbf{v}_1, \ldots, \mathbf{v}_r]^T$ [22], (3.2) converges almost surely to $\mathbf{P}_*$ if $\mathbf{P}_k$ remains inside a bounded subset. To evaluate the asymptotic distributions of the subspace projection matrix estimators given by the previous algorithms, we shall use a general Gaussian approximation result ([9, theorem 2, p. 108]) which we now recall for convenience of the reader.

### 3.1 A short review of a general Gaussian approximation result

Consider a constant learning rate recursive stochastic algorithm (we write $\Theta_k^\gamma$ for the sequence of estimates to emphasize the dependence on $\gamma$):

$$\Theta_{k+1}^\gamma = \Theta_k^\gamma + \gamma f(\Theta_k^\gamma, \mathbf{x}_k) + \gamma^2 h_k(\Theta_k^\gamma, \mathbf{x}_k) \tag{3.5}$$

with $\mathbf{x}_k = g(\xi_k)$, where $\xi_k$ is a Markov chain independent of $\Theta_k^\gamma$ and with $h_k(\Theta, \mathbf{x})$ a uniformly bounded function for $(\Theta, \mathbf{x})$ in some fixed compact set. Suppose that the parameter vector $\Theta_k^\gamma$ converges almost surely to the unique asymptotically stable point $\Theta_*$ in the corresponding decreasing learning rate algorithm. Consider the continuous Lyapunov equation:

$$\mathbf{D}\mathbf{C}_\Theta + \mathbf{C}_\Theta \mathbf{D}^T + \mathbf{G} = \mathbf{O} \tag{3.6}$$

and where $\mathbf{D}$ and $\mathbf{G}$ are respectively the derivative of the mean field and the covariance of the field of the algorithm (3.5):

$$\mathbf{D} \stackrel{\text{def}}{=} \mathrm{E}[\frac{\partial f}{\partial \Theta}(\Theta, \mathbf{x}_k)]_{\Theta=\Theta_*} \tag{3.7}$$

$$\mathbf{G} \stackrel{\text{def}}{=} \sum_{k=-\infty}^{\infty} \mathrm{Cov}[f(\Theta_*, \mathbf{x}_k), f(\Theta_*, \mathbf{x}_0)] \tag{3.8}$$

If all the eigenvalues of the derivative of the mean field $\mathbf{D}$ have strictly negative real parts, then, in a stationary situation, when $\gamma \to 0$ and $k \to \infty$, we have:

$$\frac{1}{\sqrt{\gamma}}(\Theta_k^\gamma - \Theta_*) \stackrel{\mathcal{L}}{\to} \mathcal{N}(0, \mathbf{C}_\Theta) \tag{3.9}$$

where $\mathbf{C}_\Theta$ is the unique symmetric solution of the Lyapunov equation (3.6).

## 3.2 Asymptotic distributions of projection matrix estimators

### 3.2.1 Local characterization of the field

According to the previous section and following the methodology explained in [13], one needs to characterize two local properties of the field $f(\mathbf{P}, \mathbf{x}\mathbf{x}^T)$: the mean value of its derivative, and its covariance, both evaluated at the point $\mathbf{P} = \mathbf{P}_*$. To proceed, it will be convenient to define the following orthonormal basis for the $n \times n$ symmetric matrices ($\mathbf{v}_i$ is defined in section 2.1 and the inner product under consideration is $(\mathbf{A}, \mathbf{B}) \stackrel{\text{def}}{=} \text{Tr}\mathbf{A}^T\mathbf{B}$):

$$\mathbf{S}_{ij} = \begin{cases} \mathbf{v}_i\mathbf{v}_j^T & i = j \\ \frac{\mathbf{v}_i\mathbf{v}_j^T + \mathbf{v}_j\mathbf{v}_i^T}{\sqrt{2}} & i < j. \end{cases} \tag{3.10}$$

With this definition, a first order approximation in the neighborhood of $\mathbf{P}_*$ of the mean field, and the eigenstructure of the covariance matrix of the field, are given by the following lemma:

**Lemma 1** *For $1 \leq i \leq j \leq n$, in case of independent learning patterns,*

$$\text{E}f(\mathbf{P}_* + \epsilon\ \mathbf{S}_{ij},\ \mathbf{x}_k\mathbf{x}_k^T) = \epsilon\ \mu_{ij}\ \mathbf{S}_{ij}\ +\ O(\epsilon^2), \tag{3.11}$$

$$\text{Cov}(\text{Vec}(f(\mathbf{P}_*, \mathbf{x}_k\mathbf{x}_k^T)))\ \text{Vec}(\mathbf{S}_{ij}) = \nu_{ij}\text{Vec}(\mathbf{S}_{ij}), \tag{3.12}$$

*with, respectively,*

$$\mu_{ij} \stackrel{\text{def}}{=} \lambda_i(1_{i>r} - 1_{i\leq r}) + \lambda_j(1_{j>r} - 1_{j\leq r}) \quad \textit{and} \quad \nu_{ij} \stackrel{\text{def}}{=} 2(1_{i\leq r} - 1_{j\leq r})^2\lambda_i\lambda_j. \tag{3.13}$$

### 3.2.2 Real parameterization.

To apply the Benveniste results recalled in section 3.1, we must check that the required conditions on $f$ and $h$ hold. Since $\|h(\mathbf{P}_k, \mathbf{x}_k\mathbf{x}_k^T)\| \leq 4\|\mathbf{I}_n - \mathbf{P}_k\|^2\|\mathbf{P}_k\|\|\mathbf{x}_k\|^4$, the required condition A3 (ii) for the complementary term mentioned in [9, p. 216] is fulfilled. As for the field $f$, we note from (3.13) that some eigenvalues of the derivative of the mean field are positive real, whereas the Benveniste results require strictly negative real parts for these eigenvalues. To adapt these results to our needs, the $n \times n$

rank-$r$ symmetric matrix $\mathbf{P}$ should be parameterized by a vector $\Theta$ of real parameters. Counting degrees of freedom, for example from the singular value decomposition, shows that the set of $n \times n$ rank-$r$ symmetric matrices is a $\frac{r}{2}(2n - r + 1)$-dimensional manifold. Let us now consider the parameterization of $\mathbf{P}_k$ in a neighborhood of $\mathbf{P}_*$. If $\{\theta_{ij}(\mathbf{P}) | 1 \leq i \leq j \leq n\}$ are the coordinates of $\mathbf{P} - \mathbf{P}_*$ in the basis $\mathbf{S}_{i,j}$, then,

$$\theta_{ij}(\mathbf{P}) = \mathrm{Tr}\{\mathbf{S}_{ij}(\mathbf{P} - \mathbf{P}_*)\} \qquad \text{for} \quad 1 \leq i \leq j \leq n, \tag{3.14}$$

$$\mathbf{P} = \mathbf{P}_* + \sum_{1 \leq i \leq j \leq n} \theta_{ij}(\mathbf{P})\, \mathbf{S}_{ij}. \tag{3.15}$$

The relevance of these parameters is shown by the following lemma:

**Lemma 2** *If* $\mathbf{P}$ *is an* $n \times n$ *rank-*$r$ *symmetric matrix, then*

$$\mathbf{P} = \mathbf{P}_* + \sum_{(i,j) \in P_s} \theta_{ij}(\mathbf{P})\, \mathbf{S}_{ij} \;+\; O(\|\mathbf{P} - \mathbf{P}_*\|^2) \tag{3.16}$$

*where* $P_s$ *is the complement of* $\{(i,j) \,|\, r < i \leq j \leq n\}$, *i.e.* $P_s \stackrel{\text{def}}{=} \{(i,j) \mid 1 \leq i \leq j \leq n \text{ and } i \leq r\}$.

There are $\frac{r}{2}(2n - r + 1)$ pairs in $P_s$ and this is exactly the dimension of the manifold of $n \times n$ rank-$r$ symmetric matrices. This point, together with eq. (3.16), 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 $\mathbf{P}_*$. 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 $\Theta(\mathbf{P})$ defined by: $\Theta(\mathbf{P}) \stackrel{\text{def}}{=} \mathcal{S}^T \mathrm{Vec}(\mathbf{P} - \mathbf{P}_*)$, where $\mathcal{S}$ denotes the following $n^2 \times \frac{r}{2}(2n - r + 1)$ matrix:

$$\mathcal{S} \stackrel{\text{def}}{=} [\ldots, \mathrm{Vec}(\mathbf{S}_{ij}), \ldots], \quad (i,j) \in P_s. \tag{3.17}$$

We note that the particular ordering of the pairs in the set $P_s$ is irrelevant if this ordering is preserved for all the forthcoming diagonal matrices indiced by $(i,j)$. If $\mathcal{P}(\Theta)$ denotes the unique (for $\|\Theta\|$ small enough) $n \times n$ rank-$r$ symmetric matrix such that $\mathcal{S}^T \mathrm{Vec}(\mathcal{P}(\Theta) - \mathbf{P}_*) = \Theta$, the following one-to-one mapping is exhibited for small enough $\|\Theta_t\|$:

$$\mathrm{Vec}(\mathcal{P}(\Theta_k)) = \mathrm{Vec}(\mathbf{P}_*) + \mathcal{S}\Theta_k + O(\|\Theta_k\|^2) \leftrightarrow \Theta_k = \mathcal{S}^T \mathrm{Vec}(\mathbf{P}_k - \mathbf{P}_*) \tag{3.18}$$

### 3.2.3 Solution of the Lyapunov equation

We are now in position to solve the Lyapunov equation in the new parameter $\Theta$ defined in the previous section. The stochastic equation governing the evolution of this vector parameter is obtained by applying the transformation $\mathbf{P}_k \to \Theta_k = \mathcal{S}^T \mathrm{Vec}(\mathbf{P}_k - \mathbf{P}_*)$ to the original equation (3.2), thereby giving

$$\Theta_{k+1} = \Theta_k + \gamma_k \phi(\Theta_k, \mathbf{x}_k) + \gamma_k^2 \psi(\Theta_k, \mathbf{x}_k) \tag{3.19}$$

where the functions $\phi$ and $\psi$ turn out to be

$$\phi(\Theta, \mathbf{x}) \stackrel{\text{def}}{=} \mathcal{S}^T \mathrm{Vec}(f(\mathcal{P}(\Theta), \mathbf{x}\mathbf{x}^T)), \tag{3.20}$$

$$\psi(\Theta, \mathbf{x}) \stackrel{\text{def}}{=} \mathcal{S}^T \mathrm{Vec}(h(\mathcal{P}(\Theta), \mathbf{x}\mathbf{x}^T)), \tag{3.21}$$

where, like $h$, $\psi$ verifies the condition A3(ii) of [9, p. 216]. We need to evaluate the derivative matrix $\mathbf{D}$ of $E\phi(\Theta, \mathbf{x})$ at point $\Theta = \mathbf{0}$, and since we consider only the case of independent learning patterns, the covariance matrix $\mathbf{\Gamma}$ of $\phi(\mathbf{0}, \mathbf{x})$. With these notations, the results of section 3.2.1 are recycled as follows:

$$\begin{aligned} E\phi(\Theta, \mathbf{x}) &= \mathcal{S}^T \mathrm{Vec}\mathrm{E} f(\mathcal{P}(\Theta), \mathbf{x}\mathbf{x}^T) = \mathcal{S}^T \mathrm{Vec}\mathrm{E} f\left(\mathbf{P}_* + \sum \theta_{ij}\mathbf{S}_{ij} + O(\|\Theta\|^2),\ \mathbf{x}\mathbf{x}^T\right) \\ &= \mathcal{S}^T \mathrm{Vec}\left(\sum \theta_{ij}\mu_{ij}\mathbf{S}_{ij} + O(\|\Theta\|^2)\right) = \mathcal{S}^T(\mathcal{S}\mathbf{\Delta}_\mu\Theta + O(\|\Theta\|^2) = \mathbf{\Delta}_\mu\Theta + O(\|\Theta\|^2), \end{aligned} \tag{3.22}$$

where the above summations are over $(i, j) \in P_s$. The first equality uses definition (3.20) and the linearity of the Vec operation; the second equality stems from property (3.18) of the reparameterization; the third equality uses lemma 1 and the differentiability of $f$; the fourth equality is induced by definitions (3.13) and (3.23). The final equality is due to the orthonormality of the basis $\{\mathbf{S}_{ij}\}$, and enables us to conclude that

$$\mathbf{D} \stackrel{\text{def}}{=} \left.\frac{\partial E\phi(\Theta, \mathbf{x})}{\partial \Theta}\right|_{\Theta=\mathbf{0}} = \mathbf{\Delta}_\mu, \quad \text{with} \quad \mathbf{\Delta}_\mu \stackrel{\text{def}}{=} \mathrm{Diag}(\ldots, \mu_{ij}, \ldots) \quad (i,j) \in P_s \quad \text{and now} \quad \mu_{ij} < 0 \quad (i,j) \in P_s. \tag{3.23}$$

We now proceed with evaluating the covariance of the field at $\Theta = \mathbf{0}$:

$$\mathrm{Cov}(\phi(\mathbf{0}, \mathbf{x})) = \mathrm{Cov}(\mathcal{S}^T \mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}\mathbf{x}^T))) = \mathcal{S}^T \mathrm{Cov}(\mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}\mathbf{x}^T)))\mathcal{S} = \mathcal{S}^T\mathcal{S}\mathbf{\Delta}_\nu = \mathbf{\Delta}_\nu. \tag{3.24}$$

The first equality holds by definition of $\phi$; the second equality is due to the bilinearity of the Cov operator; the third equality is obtained by noting that (3.12) also reads $\mathrm{Cov}(\mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}\mathbf{x}^T)))\mathcal{S} = \mathcal{S}\mathbf{\Delta}_\nu$, with $\mathbf{\Delta}_\nu$ defined by (3.25). The final equality is due to the orthonormality of the basis $\{\mathbf{S}_{ij}\}$, and it enables us to conclude that for independent learning patterns:

$$\mathbf{G} \stackrel{\text{def}}{=} \mathrm{Cov}(\phi(0, \mathbf{x})) = \mathbf{\Delta}_\nu, \quad \text{with} \quad \mathbf{\Delta}_\nu \stackrel{\text{def}}{=} \mathrm{Diag}(\ldots, \nu_{ij}, \ldots) \qquad (i,j) \in P_s. \tag{3.25}$$

Thus both $\mathbf{G}$ and $\mathbf{D}$ are diagonal matrices. In this case, the Lyapunov equation (3.6) reduces to $\frac{r}{2}(2n - r + 1)$ uncoupled scalar equations. Thus the solution is clearly

$$\mathbf{C}_\Theta = -\frac{1}{2}\mathbf{\Delta}_\nu \mathbf{\Delta}_\mu^{-1}. \tag{3.26}$$

According to (3.9), $\gamma^{-1/2}\Theta_k \to_{\mathcal{L}} \mathcal{N}(\mathbf{0}, -\frac{1}{2}\mathbf{\Delta}_\nu\mathbf{\Delta}_\mu^{-1})$. By eq. (3.18), we have $\mathrm{Vec}(\mathbf{P}_k) = \mathrm{Vec}(\mathbf{P}_*) + \mathcal{S}\Theta_k + O(\|\Theta_k\|^2)$. We conclude that for $\gamma \to 0$ and $k \to +\infty$,

$$\frac{1}{\sqrt{\gamma}}(\mathrm{Vec}(\mathbf{P}_k) - \mathrm{Vec}(\mathbf{P}_*)) \to_{\mathcal{L}} \mathcal{N}(\mathbf{0}, \mathbf{C}_P) \quad \text{with} \quad \mathbf{C}_P = \mathcal{S}\mathbf{C}_\Theta\mathcal{S}^T = -\frac{1}{2}\mathcal{S}\mathbf{\Delta}_\nu\mathbf{\Delta}_\mu^{-1}\mathcal{S}^T. \tag{3.27}$$

The expression (3.27) of the covariance matrix $\mathbf{C}_P$ in the asymptotic distribution of $\mathrm{Vec}(\mathbf{P}_k)$ may be written as an explicit sum:

$$\mathbf{C}_P = \sum_{(i,j)\in P_s} \frac{\nu_{ij}}{-2\mu_{ij}} \mathrm{Vec}(\mathbf{S}_{ij})\mathrm{Vec}(\mathbf{S}_{ij})^T. \tag{3.28}$$

From the definitions (3.13) of $\mu_{ij}$ and $\nu_{ij}$, and noting that $\nu_{ij} = 0$ for $i \le j \le r$ and $\mathrm{Vec}(\mathbf{v}_i\mathbf{v}_j^T) = \mathbf{v}_j \otimes \mathbf{v}_i$, the expression (3.28) is finally rewritten as

$$\mathbf{C}_P = \sum_{1\le i\le r<j\le n} \frac{\lambda_i\lambda_j}{2(\lambda_i - \lambda_j)}(\mathbf{v}_i \otimes \mathbf{v}_j + \mathbf{v}_j \otimes \mathbf{v}_i)(\mathbf{v}_i \otimes \mathbf{v}_j + \mathbf{v}_j \otimes \mathbf{v}_i)^T \tag{3.29}$$

This expression coincides with the expression of the covariance matrix $\mathbf{C}_P$ of the Yang algorithm (2.7) given in [13], despite some differences in the expression of $\mu_{ij}$ and $\nu_{ij}$. In fact the "symmetrization" of the SNL algorithm implies that the terms $\frac{\nu_{ij}}{\mu_{ij}}$ remain invariant for $(i,j) \in P_s$. Furthermore, we note that the expression (3.29) is the limit when $\beta_i$ tends to 1 for all $i$ of the expression of the covariance matrix $\mathbf{C}_P$ of the WSA algorithm given in [12].

### 3.3 Study of the smoothed SNL algorithm

To study the smoothed SNL algorithm, we note that eqs.(2.10) and (2.11) take globally the form (3.5) if we set $\mathbf{\Theta}_k \stackrel{\text{def}}{=} \begin{bmatrix} \text{Vec}(\mathbf{R}_k) \\ \\ \text{Vec}(\mathbf{W}_k) \end{bmatrix}$. Then, if we consider the trajectory of the associated matrix $\mathbf{R}_k$, as $\mathbf{P}_k$ remains symmetric (when the initial condition $\mathbf{R}_0$ is symmetric), it is natural to use the parameter $\mathbf{\Theta}_k = \begin{bmatrix} \Theta_{1,k} \\ \\ \Theta_{2,k} \end{bmatrix}$, i.e., the respective coordinates of $\mathbf{R}_k$ in the basis $\mathbf{S}_{ij}$, $1 \leq i \leq j \leq n$ and of $\mathbf{P}_k$ in the basis $\mathbf{S}_{ij}$, $(i,j) \in P_s$. So, $\Theta_{1,k} = {\mathcal{S}'}^T \text{Vec}(\mathbf{R}_k)$, in which

$$\mathcal{S}' \stackrel{\text{def}}{=} [\ldots, \text{Vec}(\mathbf{S}_{ij}), \ldots], \quad (i,j) \in P_{s'}, \quad \text{with} \quad P_{s'} \stackrel{\text{def}}{=} \{(i,j) | 1 \leq i \leq j \leq n\},$$

and $\Theta_{2,k} = \mathcal{S}^T \text{Vec}(\mathbf{P}_k - \mathbf{P}_*)$. As such, $\mathbf{\Theta}_k$ follows a stochastic equation of the form (3.19). In this equation $\phi(\mathbf{\Theta}_k, \mathbf{x}_k) \stackrel{\text{def}}{=} \begin{bmatrix} \phi_1(\mathbf{\Theta}_k, \mathbf{x}_k) \\ \\ \phi_2(\mathbf{\Theta}_k, \mathbf{x}_k) \end{bmatrix}$ and $\psi(\mathbf{\Theta}_k, \mathbf{x}_k) \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{0} \\ \\ \psi_2(\mathbf{\Theta}_k, \mathbf{x}_k) \end{bmatrix}$, where

$$\phi_1(\mathbf{\Theta}, \mathbf{x}) \stackrel{\text{def}}{=} \alpha({\mathcal{S}'}^T \text{Vec}(\mathbf{x}\mathbf{x}^T) - \Theta_1), \tag{3.30}$$

$$\phi_2(\mathbf{\Theta}, \mathbf{x}) \stackrel{\text{def}}{=} \mathcal{S}^T \text{Vec}\left( f(\mathcal{P}(\Theta_2), \text{Vec}^{-1}(\mathcal{S}'\Theta_1)) \right), \tag{3.31}$$

$$\psi(\mathbf{\Theta}, \mathbf{x}) \stackrel{\text{def}}{=} \mathcal{S}^T \text{Vec}\left( h(\mathcal{P}(\Theta_2), \text{Vec}^{-1}(\mathcal{S}'\Theta_1)) \right). \tag{3.32}$$

Note firstly that $\mathbf{R}_k^\gamma$ converges almost surely to $\mathbf{R}_* = \mathbf{R}_x = \sum_{1 \leq i \leq n} \lambda_i \mathbf{S}_{ii}$ when $\gamma \to 0$ and $k \to \infty$. So $\Theta_{1,k}$ converges almost surely to $\Theta_{1,*} = \{\theta_{1,ij,*}\}_{(ij) \in P_{s'}}$ with $\theta_{1,ii,*} = \lambda_i$ for $1 \leq i \leq n$ and $\theta_{1,ij,*} = 0$ elsewhere. This type of coupled algorithm introduces a form of relaxation: the solution of the first equation is fed directly back into the second. According to section 3.1, we must check that the required condition on $\phi$ and $\psi$ holds and we need to characterize the mean value of the derivative and the covariance of the field $\phi(\mathbf{\Theta}, \mathbf{x}_k)$. Like $h$, $\psi$ verifies the condition mentioned in [9, A3 (ii), p. 216]. To evaluate the derivative matrix $\mathbf{D}$ of $\text{E}\phi(\mathbf{\Theta}, \mathbf{x}_k)$ at point $\mathbf{\Theta}_* = \begin{bmatrix} \Theta_{1,*} \\ \\ \mathbf{0} \end{bmatrix}$, we need the following lemma:

**Lemma 3** *For $1 \leq i \leq j \leq n$,*

$$f(\mathbf{P}_*, \mathbf{R}_* + \epsilon\ \mathbf{S}_{ij}) = \epsilon\ \kappa_{ij}\ \mathbf{S}_{ij} \tag{3.33}$$

*with*

$$\kappa_{ij} \stackrel{\text{def}}{=} 1_{i\leq r\leq j} + 1_{j\leq r\leq i}. \tag{3.34}$$

So, in the neighborhood of $\mathbf{\Theta}_*$, we have

$$\mathrm{E}\phi_1(\mathbf{\Theta}, \mathbf{x}) = \alpha(\Theta_{1,*} - \Theta_1) \tag{3.35}$$

and

$$\begin{aligned}
\mathrm{E}\phi_2(\mathbf{\Theta}, \mathbf{x}) &= \mathcal{S}^T \mathrm{Vec}\left( f(\mathbf{P}_* + \sum_{(i,j)\in P_s} \theta_{2,ij}\mathbf{S}_{ij} + O(\|\Theta_2\|^2), \mathbf{R}_* + \sum_{(i,j)\in P_{s'}} (\theta_{1,ij} - \theta_{1,ij,*})\mathbf{S}_{ij}) \right) \\
&= \mathcal{S}^T \mathrm{Vec}\left( \sum_{(i,j)\in P_s} \theta_{2,ij}\mu_{ij}\mathbf{S}_{ij} + O(\|\Theta_2\|^2) + \sum_{(i,j)\in P_{s'}} (\theta_{1,ij} - \theta_{1,ij,*})\kappa_{ij}\mathbf{S}_{ij} \right) \\
&= \mathcal{S}^T(\mathcal{S}\mathbf{\Delta}_\mu\Theta_2 + O(\|\Theta_2\|^2) + \mathcal{S}'\mathbf{\Delta}'_\kappa(\Theta_1 - \Theta_{1,*})) \\
&= \mathbf{\Delta}_\mu\Theta_2 + O(\|\Theta_2\|^2) + \left[\mathbf{I}_{r(2n-r+1)/2}, \mathbf{0}\right] \mathbf{\Delta}'_\kappa(\Theta_1 - \Theta_{1,*}),
\end{aligned} \tag{3.36}$$

where the second equality uses the differentiability of $f$ with lemmas 2 and 3, the third equality uses the diagonal matrix $\mathbf{\Delta}'_\kappa \stackrel{\text{def}}{=} \mathrm{Diag}(\ldots, \kappa_{ij}, \ldots)$ for $(i,j) \in P_{s'}$, and the last equality is due to the orthonormality of the basis $\mathbf{S}_{ij}$. Eq. (3.36) enables us to conclude that:

$$\mathbf{D} \stackrel{\text{def}}{=} \left.\frac{\partial \mathrm{E}\phi(\mathbf{\Theta}, \mathbf{x})}{\partial \mathbf{\Theta}}\right|_{\mathbf{\Theta}=\mathbf{\Theta}_*} = \begin{bmatrix} -\alpha\mathbf{I}_{n(n-1)/2} & \mathbf{0} \\ \mathbf{\Delta}_\kappa \;\; \mathbf{0} & \mathbf{\Delta}_\mu \end{bmatrix} \tag{3.37}$$

with $\mathbf{\Delta}_\kappa \stackrel{\text{def}}{=} \mathrm{Diag}(\ldots, \kappa_{ij}, \ldots)$ for $(i,j) \in P_s$. We note that like $\mathbf{\Delta}_\mu$, the eigenvalues of $\mathbf{D}$ are real and strictly negative. We proceed with evaluating the covariance of $\phi(\mathbf{\Theta}, \mathbf{x})$ at $\mathbf{\Theta} = \mathbf{\Theta}_*$:

$$\mathbf{G} \stackrel{\text{def}}{=} \mathrm{Cov}\phi(\mathbf{\Theta}_*, \mathbf{x}) = \begin{bmatrix} \mathrm{Cov}\phi_1(\mathbf{\Theta}_*, \mathbf{x}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{3.38}$$

with

$$\begin{aligned}
\mathrm{Cov}\phi_1(\mathbf{\Theta}_*, \mathbf{x}) &= \alpha^2 \mathrm{Cov}(\mathcal{S}'^T \mathrm{Vec}(\mathbf{x}\mathbf{x}^T) - \Theta_{1,*}) = \alpha^2 \mathcal{S}'^T \mathrm{Cov}(\mathrm{Vec}(\mathbf{x}\mathbf{x}^T))\mathcal{S}' \\
&= \alpha^2 \mathcal{S}'^T (\mathbf{R}_x \otimes \mathbf{R}_x)(\mathbf{I}_{n^2} + \mathbf{K})\mathcal{S}' \\
&= 2\alpha^2 \mathcal{S}'^T \mathcal{S}' \mathbf{\Delta}_\xi = 2\alpha^2 \mathbf{\Delta}_\xi.
\end{aligned} \tag{3.39}$$

The third equality uses (A.6), and the fourth equality stems from (A.7), (A.10) and the definition $\mathbf{\Delta}_\xi \stackrel{\text{def}}{=} \text{Diag}(\ldots, \lambda_i\lambda_j, \ldots)$ for $(i,j) \in P_{s'}$. Thus,

$$\mathbf{G} \stackrel{\text{def}}{=} \text{Cov}\phi(\mathbf{\Theta}_*, \mathbf{x}) = \begin{bmatrix} 2\alpha^2\mathbf{\Delta}_\xi & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}. \tag{3.40}$$

The Lyapunov equation (3.6) then has a block triangular form, the unique symmetric solution of which is:

$$\mathbf{C}_\Theta = \begin{bmatrix} \mathbf{C}_{\Theta_1} & \mathbf{C}^T_{\Theta_1,\Theta_2} \\ \mathbf{C}_{\Theta_1,\Theta_2} & \mathbf{C}_{\Theta_2} \end{bmatrix}, \tag{3.41}$$

with

$$\mathbf{C}_{\Theta_2} = -\alpha(\alpha\mathbf{I}_{n(2n-r+1)/2} - \mathbf{\Delta}_\mu)^{-1}\mathbf{\Delta}_\kappa^2\mathbf{\Delta}_\mu^{-1}\mathbf{\Delta}_\xi. \tag{3.42}$$

Then, as in section 3.2.3, we deduce $\mathbf{C}_P = \mathcal{S}\mathbf{C}_{\Theta_2}\mathcal{S}^T$. From the definition of $\mu_{ij}, \kappa_{ij}$ and $\xi_{ij}$ and noting that $\kappa_{ij} = 0$ for $i \leq j \leq r$, the matrix $\mathbf{C}_P$ is finally written in a similar form to (3.29), except for the term $\alpha_{ij} \stackrel{\text{def}}{=} \frac{\alpha}{\alpha+\lambda_i-\lambda_j} < 1$, yielding

$$\mathbf{C}_P = \sum_{1\leq i\leq r<j\leq n} \frac{\alpha_{ij}\lambda_i\lambda_j}{2(\lambda_i-\lambda_j)}(\mathbf{v}_i\otimes\mathbf{v}_j + \mathbf{v}_j\otimes\mathbf{v}_i)(\mathbf{v}_i\otimes\mathbf{v}_j + \mathbf{v}_j\otimes\mathbf{v}_i)^T. \tag{3.43}$$

### 3.4 Analysis of the results

Firstly, the expressions (3.29) and (3.43) can be compared to the covariances of the asymptotic distributions obtained in batch estimation. If $\mathbf{P}_k = \sum_{1\leq i\leq r} \mathbf{w}_{k,i}\mathbf{w}_{k,i}^T$ denotes the batch estimated orthogonal projection matrix, we have from [13]

$$\sqrt{k}\ (\text{Vec}(\mathbf{P}_k) - \text{Vec}(\mathbf{P}_*)) \stackrel{\mathcal{L}}{\to} \mathcal{N}(\mathbf{O}, \mathbf{C}_P) \tag{3.44}$$

when $k$ tends to $+\infty$ with

$$\mathbf{C}_P = \sum_{1\leq i\leq r<j\leq n} \frac{\lambda_i\lambda_j}{(\lambda_i-\lambda_j)^2}(\mathbf{v}_i\otimes\mathbf{v}_j + \mathbf{v}_j\otimes\mathbf{v}_i)(\mathbf{v}_i\otimes\mathbf{v}_j + \mathbf{v}_j\otimes\mathbf{v}_i)^T \tag{3.45}$$

which is also in close similarity with (3.29) and (3.43).

Secondly, a simple global measure of performance is the MSE between $\mathbf{P}_k$ and $\mathbf{P}_*$. Indeed, since the projection matrix $\mathbf{P}_k$ characterizes the estimated subspace, $\mathrm{E}\| \mathbf{P}_k - \mathbf{P}_* \|_{\mathrm{Fro}}^2$ is a measure of the distance between the estimated and the desired principal component subspaces.

To give a MSE expression, we assume, as is customary, that the first and second asymptotic moments of $\mathbf{P}_k$ are those of its asymptotic distribution. This implies:

$$\| \mathrm{E}(\mathbf{P}_k - \mathbf{P}_*) \|_{\mathrm{Fro}}^2 = o(\gamma), \qquad \mathrm{Cov}(\mathrm{Vec}(\mathbf{P}_k)) = \gamma \mathbf{C}_P + o(\gamma). \tag{3.46}$$

In particular, the MSE between $\mathbf{P}_k$ and $\mathbf{P}_*$ is given by the trace of the covariance matrix of the asymptotic distribution of $\mathbf{P}_k$. Since trace is invariant under an orthonormal change of basis with $\{\mathrm{Vec}(\mathbf{S}_{ij}) | 1 \leq i \leq j \leq n\}$ being an orthonormal basis, we obtain from eq. (3.28) and (3.43) that

$$\mathrm{E}\| \mathbf{P}_k - \mathbf{P}_* \|_{\mathrm{Fro}}^2 = \gamma \sum_{1 \leq i \leq r < j \leq n} \frac{\alpha_{ij} \lambda_i \lambda_j}{\lambda_i - \lambda_j} + o(\gamma) \tag{3.47}$$

where $\alpha_{ij} < 1$ for the smoothed SNL algorithm and $\alpha_{ij} \stackrel{\mathrm{def}}{=} 1$ for the SNL algorithm.

Finally, following the methodology explained in [13], a finer picture of the MSE of $\mathbf{C}_P$ can be derived from the regular structure (3.29) and (3.43) of the covariance matrix $\mathbf{C}_P$ by decomposing the error $\mathbf{P}_k - \mathbf{P}_*$ into three orthogonal terms. Furthermore, we note that as for the Yang algorithm, our first-order analysis does not provide the order of deviation from orthonormality. We show in section 5 that this MSE of orthonormality is, to a first-order approximation, proportional to $\gamma^2$ for the SNL algorithm, and to $\gamma^4$ for the smoothed SNL algorithm.

## 4 Extension to correlated training patterns

This section gives explicit solutions for the case of real correlated training patterns for the SNL algorithm; the extension to the modified SNL algorithm is straightforword. The covariance of the field has a more involved expression: from (A.5) we have

$$\mathrm{Cov}[\mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}_k \mathbf{x}_k^T)), \mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}_0 \mathbf{x}_0^T))] = \mathbf{Q} \mathrm{Cov}(\mathrm{Vec}(\mathbf{x}_k \mathbf{x}_k^T), \mathrm{Vec}(\mathbf{x}_0 \mathbf{x}_0^T)) \mathbf{Q} \tag{4.1}$$

According to the following property ([29] p. 57) for Gaussian real signals

$$\mathrm{Cov}(\mathrm{Vec}(\mathbf{x}_k\mathbf{x}_k^T), \mathrm{Vec}(\mathbf{x}_0\mathbf{x}_0^T)) = \mathbf{R}_{k,0} \otimes \mathbf{R}_{k,0} + (\mathbf{R}_{k,0} \otimes \mathbf{R}_{k,0})\mathbf{K} \tag{4.2}$$

where $\mathbf{R}_{k,0} \stackrel{\text{def}}{=} \mathrm{E}(\mathbf{x}_k\mathbf{x}_0^T)$, we have

$$\mathrm{Cov}[\mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}_k\mathbf{x}_k^T)), \mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}_0\mathbf{x}_0^T))] = \mathbf{Q}(\mathbf{R}_{k,0} \otimes \mathbf{R}_{k,0})(\mathbf{I} + \mathbf{K})\mathbf{Q}. \tag{4.3}$$

Thus, we can write

$$\sum_{k=-\infty}^{+\infty} \mathrm{Cov}[\mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}_k\mathbf{x}_k^T)), \mathrm{Vec}(f(\mathbf{P}_*, \mathbf{x}_0\mathbf{x}_0^T))] = \mathbf{Q}\mathcal{R}(\mathbf{I} + \mathbf{K})\mathbf{Q} \tag{4.4}$$

where we define

$$\mathcal{R} \stackrel{\text{def}}{=} \sum_{k=-\infty}^{\infty} \mathbf{R}_{k,0} \otimes \mathbf{R}_{k,0}. \tag{4.5}$$

To solve the Lyapunov equation for the asymptotic covariance of $\mathbf{P}_k$, we resort to the parameterization of $\mathbf{P}_k$ by a vector $\Theta = \{\theta_{ij}\}_{(i,j)\in P_s}$ as in section 3.2.2. However, as the matrix $\mathbf{\Gamma}$ is no longer diagonal, we must use a component-wise expression for the asymptotic covariance matrix $\mathbf{C}_\Theta$. This is

$$(\mathbf{C}_\Theta)_{ij,i'j'} = \frac{\mathrm{Vec}(\mathbf{S}_{ij})^T\mathbf{Q}\mathcal{R}(\mathbf{I} + \mathbf{K})\mathbf{Q}\mathrm{Vec}(\mathbf{S}_{i'j'})}{-(\mu_{ij} + \mu_{i'j'})} \quad (i,j) \in P_s, \quad (i',j') \in P_s \tag{4.6}$$

This may be simplified using the following properties: for any pair $(i,j)$, we have $\mathbf{K}\mathrm{Vec}(\mathbf{S}_{ij}) = \mathrm{Vec}(\mathbf{S}_{ij})$ from (A.7), eq. (A.9) and finally the particular expressions of $\mu_{ij}$ and $\mu_{i'j'}$. This results in:

$$(\mathbf{C}_\Theta)_{ij,i'j'} = \begin{cases} \frac{2(1_{i\le r} - 1_{j\le r})^2(1_{i'\le r} - 1_{j'\le r})^2}{(\lambda_i - \lambda_j) + (\lambda_{i'} - \lambda_{j'})}\ \mathrm{Vec}(\mathbf{S}_{ij})^T\mathcal{R}\mathrm{Vec}(\mathbf{S}_{i'j'}) & \text{for } \ 1 \le i, i' \le r < j, j' \le n \\ 0 & \text{elsewhere} \end{cases} \tag{4.7}$$

Unfortunately, no significantly simpler expressions seem to be available for $\mathbf{C}_\Theta$ in the correlated case.

In order to proceed, we focus on the total MSE for $\mathbf{P}_k$. As above, this is closely related to $\mathrm{Tr}\mathbf{C}_P$. Since $\mathbf{C}_P = \mathcal{S}\mathbf{C}_\Theta\mathcal{S}^T$, we have

$$\mathrm{Tr}\mathbf{C}_P = \mathrm{Tr}\mathbf{C}_\Theta = \sum_{(i,j)\in P_s} (\mathbf{C}_\Theta)_{ij,ij} = \sum_{1\le i\le r<j\le n} \frac{\mathrm{Vec}(\mathbf{S}_{ij})^T\mathcal{R}\mathrm{Vec}(\mathbf{S}_{ij})}{\lambda_i - \lambda_j} \tag{4.8}$$

Thus, for correlated learning patterns, expression (3.47) generalizes to

$$E\| \mathbf{P}_k - \mathbf{P}_* \|_{\text{Fro}}^2 = \gamma\mathrm{Tr}(\mathbf{C}_P) + o(\gamma) = \gamma \sum_{1\le i\le r<j\le n} \frac{\lambda_i\lambda_j + \lambda_{i,j}}{\lambda_i - \lambda_j} + o(\gamma) \tag{4.9}$$

where the additional (with respect to the independent case) terms $\lambda_{i,j}$ are

$$\lambda_{i,j} \stackrel{\text{def}}{=} 2\sum_{k=1}^{\infty}(\mathbf{v}_i^T\mathbf{R}_{k,0}\mathbf{v}_i)(\mathbf{v}_j^T\mathbf{R}_{k,0}\mathbf{v}_j) + (\mathbf{v}_i^T\mathbf{R}_{k,0}\mathbf{v}_j)(\mathbf{v}_j^T\mathbf{R}_{k,0}\mathbf{v}_i). \tag{4.10}$$

When $\mathbf{x}_k = (x_k, x_{k-1}, \ldots, x_{k-n+1})^T$ with $x_k$ being an MA($q$), an AR($p$) or an ARMA($p,q$) stationary process, we note that $\lambda_{i,j}$ can be expressed as a finite closed form sum, as shown in [10]. This particular case has practical implications in system identification and in Karhunen Loève decomposition of time series.

# 5 Simulations

We now examine the accuracy of expressions (3.47) and (4.9) of the mean square error of the projection matrix and investigate the domain of learning rate for which our asymptotic approach is valid. Furthermore, we examine some 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.

In the first experiment, we consider the case $n = 4$, $r = 2$ associated to $\mathbf{R}_x = \text{Diag}(1.75, 1.5, 0.5, 0.25)$. Clearly, the eigenvalues of $\mathbf{R}_x$ are 1.75, 1.5, 0.5 and 0.25 and the associated eigenvectors are the unit vectors in $\mathcal{R}^4$. $\mathbf{R}_0 = \mathbf{O}$ and the entries of the initial value $\mathbf{W}_0$ are chosen randomly uniformly in [0,1], then $\mathbf{w}_{0,i}, i = 1, 2$ are normalized, and all the learning curves are averaged over 100 independent runs. First of all, in order to compare the SNL and the smoothed SNL algorithm, we consider different values of $(\alpha, \gamma)$ that provide the same value of $\gamma\text{Tr}(\mathbf{C}_P)$. Fig. 1 shows the learning curves of the mean square error of $\mathbf{P}_k$ for the SNL and the smoothed SNL algorithms. We see that the smoothed SNL algorithm with $\alpha = 0.3$ provides faster convergence than the SNL algorithm. Fig. 2 shows the associated learning curves of the deviation from orthonormality $d^2(\gamma) \stackrel{\text{def}}{=} \text{E}\| \mathbf{W}_k^T\mathbf{W}_k - \mathbf{I}_r \|_{\text{Fro}}^2$. As can be seen, the smoothed SNL algorithm provides faster convergence as well, and a smaller deviation from orthonormality. Fig. 3 shows the ratio of the estimated mean square error $\text{E}\| \mathbf{P}_k - \mathbf{P}_* \|_{\text{Fro}}^2$ over the theoretical asymptotic mean square error $\gamma\text{Tr}(\mathbf{C}_P)$ as a function of $\gamma$, for both the SNL and the smoothed SNL algorithms and with $\alpha = 1$. Our present asymptotic analysis is seen to be valid over a large range of $\gamma$ ($\gamma < 0.02$ for the SNL algorithm

and $\gamma < 0.2$ for the smoothed SNL algorithm), and the domain of "stability" is $\gamma < 0.09$ for the SNL algorithm and $\gamma < 0.25$ for the smoothed SNL algorithm, for which this ratio is closed to 1. Fig. 4 reveals something which could not be determined from our first-order analysis: the true order of deviation from orthonormality. Indeed, our analysis yields only $\mathrm{E}\| \mathbf{W}_k^T\mathbf{W}_k - \mathbf{I}_r \|_{\mathrm{Fro}}^2 = O(\gamma)$. In this figure, we plot on a log-log scale $\mathrm{E}\| \mathbf{W}_k^T\mathbf{W}_k - \mathbf{I}_r \|_{\mathrm{Fro}}^2$ as a function of $\gamma$. We find a slope equal to 2 [1] for the SNL algorithm and of 4 for the smoothed SNL algorithm, which means that, experimentally, $\mathrm{E}\| \mathbf{W}_k^T\mathbf{W}_k - \mathbf{I}_r \|_{\mathrm{Fro}}^2 \propto \gamma^2$ [resp., $\propto \gamma^4$] for the SNL [resp., the smoothed SNL] algorithm. Finally the learning speed is investigated through the iteration number until "convergence" is achieved (the convergence is considered achieved if the ratio of the estimated mean square error $\mathrm{E}\| \mathbf{P}_k - \mathbf{P}_* \|_{\mathrm{Fro}}^2$ over the theoretical asymptotic mean square error $\gamma\mathrm{Tr}(\mathbf{C}_P)$ is smaller than 1.1). Fig. 5 plots this iteration number as a function of the asymptotic mean square error $\gamma\mathrm{Tr}(\mathbf{C}_P)$ (the learning rate $\gamma$ is adjusted so that $\gamma\mathrm{Tr}(\mathbf{C}_P)$ keeps the same value for the different algorithms). As can be seen, the smoothed SNL algorithm provides a much better tradeoff between the learning speed and the misadjustment $\gamma\mathrm{Tr}(\mathbf{C}_P)$. So the various merits (deviation from orthonormality and tradeoff between learning speed and misadjustment) of the smoothed SNL algorithm can counterbalance its more computationally demanding in some applications.

In the second experiment, we compare in Fig. 6 the learning curves of the mean square error of the projection matrix on the eigenspace generated by the first two eigenvectors, for independent and then AR(1) learning patterns, produced from the same covariance matrix $\mathbf{R}_x = \begin{bmatrix} 1 & a & a^2 \\ a & 1 & a \\ a^2 & a & 1 \end{bmatrix}$ with $a = 0.3$ or 0.9 and $\gamma = 0.005$. We see that the convergence speed of these mean square errors does not seem to be affected by the correlation between the learning patterns $\mathbf{x}_k$, and that the misadjusments tend to values that agree with the theoretical values (3.47) and (4.9) respectively. Fig. 7 shows, for the same covariance matrix $\mathbf{R}_x$, the theoretical mean square error of $\mathbf{P}_k$ (normalized by the learning rate $\gamma$) for

[1] This result agrees with the presentation of the SNL algorithn given in subsection 2.1 in which the term $O(\gamma_k^2)$ was omitted from the orthonormalization of the columns of $\mathbf{W}_k$.

independent or AR(1) learning patterns, as a function of the parameter $a$ of the AR model of unit power. We observe that these errors decrease when $a$ increases, that is when the eigenvalue spread increases. We see that these errors are about 12 dB worse for independent learning patterns than for correlated learning patterns. This result was previously observed in parameterized adaptive algorithms [10].

# 6 Conclusion

We have performed in this paper a complete asymptotic performance analysis of the SNL algorithm and of a smoothed SNL algorithm that we have introduced, assuming a constant learning rate, and in the case where a large number of patterns is available. A closed form expression of the covariance in distribution of the projection matrices onto the principal component subspace estimators has been given in case of independent or correlated learning patterns. We showed that the misadjustment effects are sensitive to the temporal correlation between successive learning patterns. The tradeoff between the speed of convergence and misadjustment, as well as the deviation from orthonormality, have also been investigated. Naturally the covariance of the limiting distribution and consequently the mean square errors of any function of the projection matrix $\mathbf{P}_k$ could be obtained, such as the DOAs [1] or the finite impulse response [2] estimated by the MUSIC algorithm.

# Appendix

**Proof of lemma 1** As the field $f$ in definition (3.3) is linear in its second argument, the mean field at any point $\mathbf{P}$ is

$$\mathrm{E}f(\mathbf{P}, \mathbf{x}_k\mathbf{x}_k^T) = f(\mathbf{P}, \mathrm{E}(\mathbf{x}_k\mathbf{x}_k^T)) = f(\mathbf{P}, \mathbf{R}_x) = (\mathbf{I}_n - \mathbf{P})\mathbf{R}_x\mathbf{P} + \mathbf{P}\mathbf{R}_x(\mathbf{I}_n - \mathbf{P}). \tag{A.1}$$

Using $\mathbf{P}_*\mathbf{v}_i = 1_{i\leq r}\mathbf{v}_i$ and $(\mathbf{I}_n - \mathbf{P}_*)\mathbf{v}_i = 1_{i>r}\mathbf{v}_i$, a substitution $\mathbf{P} = \mathbf{P}_* + \epsilon\ \mathbf{v}_i\mathbf{v}_j^T$ in (A.1) yields after simplification

$$\mathrm{E}f(\mathbf{P}_* + \epsilon\ \mathbf{v}_i\mathbf{v}_j^T, \mathbf{x}_k\mathbf{x}_k^T) = \epsilon\ \mu_{ij}\mathbf{v}_i\mathbf{v}_j^T\ +\ O(\epsilon^2) \tag{A.2}$$

where $\mu_{ij}$ is defined in eq. (3.13). The lemma follows by using the symmetry $\mu_{ij} = \mu_{ji}$.

At point $\mathbf{P} = \mathbf{P}_*$, definition (3.3) of the field reduces to

$$f(\mathbf{P}_*, \mathbf{x}\mathbf{x}^T) = \mathbf{P}_*^\perp \mathbf{x}\mathbf{x}^T \mathbf{P}_* + \mathbf{P}_* \mathbf{x}\mathbf{x}^T \mathbf{P}_*^\perp \tag{A.3}$$

which, by vectorization and using $\mathbf{Q} \stackrel{\text{def}}{=} \mathbf{P}_*^\perp \otimes \mathbf{P}_* + \mathbf{P}_* \otimes \mathbf{P}_*^\perp$ and thanks to the property

$$\text{Vec}(\mathbf{ABC}) = (\mathbf{A} \otimes \mathbf{C}^T)\text{Vec}(\mathbf{B}) \tag{A.4}$$

also reads

$$\text{Vec}(f(\mathbf{P}_*, \mathbf{x}\mathbf{x}^T)) = \mathbf{Q}\text{Vec}(\mathbf{x}\mathbf{x}^T). \tag{A.5}$$

Now, for a Gaussian vector $\mathbf{x}$, we have ([29, p. 57]):

$$\text{Cov}(\text{Vec}(\mathbf{x}\mathbf{x}^T)) = \mathbf{R}_x \otimes \mathbf{R}_x + (\mathbf{R}_x \otimes \mathbf{R}_x)\mathbf{K}. \tag{A.6}$$

where $\mathbf{K}$ is an $n^2 \times n^2$ block matrix acting as a permutation matrix, *i.e.*

$$\mathbf{K}\text{Vec}(\mathbf{x}\mathbf{y}^T) = \text{Vec}(\mathbf{y}\mathbf{x}^T) \tag{A.7}$$

for any vectors $\mathbf{x}$ and $\mathbf{y}$. Combining (A.5) and (A.6), we obtain

$$\text{Cov}(\text{Vec}(f(\mathbf{P}_*, \mathbf{x}\mathbf{x}^T))) = \mathbf{Q}\text{Cov}(\text{Vec}(\mathbf{x}\mathbf{x}^T))\mathbf{Q}^T = \mathbf{Q}(\mathbf{R}_x \otimes \mathbf{R}_x)(\mathbf{I}_{n^2} + \mathbf{K})\mathbf{Q}^T. \tag{A.8}$$

For any pair $1 \leq i, j \leq n$, by simple substitution, we find

$$\mathbf{Q}\text{Vec}(\mathbf{v}_i\mathbf{v}_j^T) = (1_{i\leq r} - 1_{j\leq r})^2 \text{Vec}(\mathbf{v}_i\mathbf{v}_j^T) \tag{A.9}$$

$$(\mathbf{R}_x \otimes \mathbf{R}_x)\text{Vec}(\mathbf{v}_i\mathbf{v}_j^T) = \lambda_i\lambda_j \text{Vec}(\mathbf{v}_i\mathbf{v}_j^T) \tag{A.10}$$

by using (A.4) and the properties $\mathbf{R}_x\mathbf{v}_i\mathbf{v}_j^T\mathbf{R}_x = \lambda_i\lambda_j\mathbf{v}_i\mathbf{v}_j^T$ and $\mathbf{P}_*\mathbf{v}_i\mathbf{v}_j^T\mathbf{P}_*^\perp = 1_{i\leq r}(1 - 1_{j\leq r})\mathbf{v}_i\mathbf{v}_j^T$ and the identity $1_{i\leq r}(1 - 1_{j\leq r}) + 1_{j\leq r}(1 - 1_{i\leq r}) = (1_{i\leq r} - 1_{j\leq r})^2$. Combining (A.8),(A.9),(A.10) and (A.7), it follows that

$$\text{Cov}(\text{Vec}(f(\mathbf{P}_*, \mathbf{x}\mathbf{x}^T)))\ \text{Vec}(\mathbf{v}_i\mathbf{v}_j^T) = \frac{1}{2}\ \nu_{ij}\ (\text{Vec}(\mathbf{v}_i\mathbf{v}_j^T) + \text{Vec}(\mathbf{v}_j\mathbf{v}_i^T)) \tag{A.11}$$

where the scalars $\nu_{ij}$ are defined in the lemma. Using $\nu_{ij} = \nu_{ji}$, symmetrization of eq. (A.11) completes the proof.

**Proof of lemma 2** Denote $\mathbf{P} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ the singular value decomposition of $\mathbf{P}$. This one is not differentiable at point $\mathbf{P} = \mathbf{P}_*$ because the eigenvalues of $\mathbf{P}_*$ are degenerate. However, results from ([30, theorem 5.4 p. 111]) are available for the perturbation of the orthogonal projector $\mathbf{V}\mathbf{V}^T$ onto the range of $\mathbf{P}$ and of the eigenvalues. This is

$$\mathbf{V}\mathbf{V}^T = \mathbf{P}_* + \mathbf{P}_*(\mathbf{P}-\mathbf{P}_*)\mathbf{P}_*^\perp + \mathbf{P}_*^\perp(\mathbf{P}-\mathbf{P}_*)\mathbf{P}_* + O(\|\mathbf{P}-\mathbf{P}_*\|^2) \quad (A.12)$$

$$\mathbf{\Lambda} = \mathbf{I}_r + O(\|\mathbf{P}-\mathbf{P}_*\|) \quad (A.13)$$

Based on this, we derive thanks to (A.12) and $\mathbf{P}_*\mathbf{P}_*^\perp = \mathbf{0}$, that $\|\mathbf{P}_*^\perp\mathbf{V}\|^2 = \mathrm{Tr}(\mathbf{P}_*^\perp\mathbf{V}\mathbf{V}^T\mathbf{P}_*^\perp) = O(\|\mathbf{P}-\mathbf{P}_*\|^2)$, and thus

$$\mathbf{P}_*^\perp\mathbf{V} = O(\|\mathbf{P}-\mathbf{P}_*\|). \quad (A.14)$$

It follows from (A.14) and (A.13) that: $\mathbf{P}_*^\perp\mathbf{P}\mathbf{P}_*^\perp = \mathbf{P}_*^\perp\mathbf{V}\mathbf{\Lambda}\mathbf{V}^T\mathbf{P}_*^\perp = O(\|\mathbf{P}-\mathbf{P}_*\|^2)$. And since $\mathbf{P}_*^\perp\mathbf{P}\mathbf{P}_*^\perp = \sum_{r<i\leq j\leq n}\theta_{ij}(\mathbf{P})\mathbf{S}_{ij}$, this completes the proof of the lemma.

**Proof of lemma 3** As the field $f$ in definition (3.3) is linear in its second argument, we obtain

$$f(\mathbf{P}_*, \mathbf{R}_* + \epsilon\ \mathbf{v}_i\mathbf{v}_j^T) = \epsilon\ (\mathbf{P}_*^\perp\mathbf{v}_i\mathbf{v}_j^T\mathbf{P}_* + \mathbf{P}_*\mathbf{v}_i\mathbf{v}_j^T\mathbf{P}_*^\perp) = \epsilon\ \kappa_{ij}\ \mathbf{v}_i\mathbf{v}_j^T \quad (A.15)$$

where we have used $\mathbf{P}_*\mathbf{v}_i = 1_{i\leq r}\mathbf{v}_i$ and $\mathbf{P}_*^\perp\mathbf{v}_i = 1_{i>r}\mathbf{v}_i$, and where $\kappa_{ij}$ is defined in eq. (3.34). The lemma follows thanks to the symmetry of $\kappa_{ij} = \kappa_{ji}$.

## References


[1] H. Krim, M. Viberg, "Two decades of array signal processing research," *IEEE Signal Processing Magazine*, pp. 67-94, July 1996.

[2] E. Moulines, P. Duhamel, J.F. Cardoso, S. Mayrargue, "Subspace methods for blind identification of multichannel FIR filters," *IEEE Trans. on Signal Processing*, vol. 43, no. 2, pp. 516-525, Feb. 1995.

[3] E. Oja, "Principal components, minor components and linear neural networks," *Neural Networks*, vol. 5, pp. 927-935, 1992.

[4] C.M. Kuan, K. Hornik, "Convergence of learning algorithms with constant learning rates," *IEEE Trans. on Neural Networks*, vol. 2, no. 5, pp. 484-489, Sep. 1991.

[5] K. Hornik, C.M. Kuan, "Convergence analysis of local feature extraction algorithms," *Neural Networks*, vol. 5, pp. 229-240, 1992.

[6] L. Ljung, "Analysis of recursive stochastic algorithms," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 551-575, 1977.

[7] H.J. Kushner, D.S. Clark, *Stochastic approximation methods for constrainted and unconstrainted systems*. New York: Springer Verlag, 1978.

[8] H.J. Kushner, *Approximation and weak convergence methods for random processes*. Cambridge, MA: MIT Press, 1984.

[9] A. Benveniste, M. Métivier, P. Priouret, *Adaptive algorithms and stochastic approximations*, Springer Verlag, 1990.

[10] J.P. Delmas, "Performance analysis of parametrized adaptive eigensubpace algorithms," in *Proc. ICASSP Detroit*, pp. 2056-2059, May 1995.

[11] B. Yang, F. Gersemsky, "Asymptotic distribution of recursive subspace estimators," *Proc. ICASSP Atlanta*, pp. 1764-1767, May 1996.

[12] J.P. Delmas, F. Alberge, "Asymptotic Performance Analysis of Subspace Adaptive Algorithms Introduced in the Neural Network Literature," to appear in *IEEE Trans. on Signal Processing*, Jan. 1998.

[13] J.P. Delmas, J.F. Cardoso "Performance analysis of an adaptative algorithm for tracking dominant subspaces," submitted to *IEEE Trans. on Signal Processing*.

[14] R. Williams, "Feature discovery through error-correcting learning," *Technical Report 8501*, San Diego, CA: University of California, Institute of Cognitive Science, 1985.

[15] P. Baldi, "Linear learning: Landscapes and algorithms," *in Proc. NIPS, Denver*, 1988.

[16] E. Oja, "Neural Networks, principal components and subspaces," *International Journal of Neural Systems*, vol.1, no.1 pp. 61-68, 1989.

[17] E. Oja, J. Karhunen, "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix," *Journal of math. analysis and applications*, 106, pp. 69-84, 1985.

[18] H. Rutishauser, "Computational aspects of F.L.Bauer's simultaneous iteration method," *Numer. Math.* vol. 13 pp. 4-13, 1969.

[19] E. Oja, "A simplified neuron model as a principal components analyzer," *Journal of Math. Biology*, vol. 15, pp. 267-273, 1982.

[20] P. Baldi, K. Hornik, "Back-propagation and unsupervised learning in linear networks," *In Y. Chauvin and D.E. Rummelhart, Back Propagation: Theory, Architectures and Applications, Earlbaum Associates*, 1991.

[21] A. Krogh, J.A. Hertz, "Hebbian learning of principal components," *In R. Eckmiller, G. Hartmann and G. Hauske, Parallel Processing in Neural Systems and Computers. Amsterdam. Elsevier Science Publishers B.V. North-Holland*, 1990.

[22] 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.

[23] L. Russo, "An outer product neural network for extracting principal components from a time series," *In B.H. Juang et al. Neural networks for signal processing*, pp. 161-170. New York IEEE Press, 1991.

[24] B. Yang, "Projection approximation subspace tracking," *IEEE, Trans. on Signal Processing*, vol. 43, no. 1, pp. 95-107, Jan. 1995.

[25] B. Yang, "Convergence analysis of the subspace tracking algorithms PAST and PASTd," *Proc. ICASSP Atlanta*, pp. 1760-1763, May 1996.

[26] E. Oja, H. Ogawa and J. Wangviwattana "Principal component analysis by homogeneous neural networks, Part I: The weighted subspace criterion," *IEICE Trans. on Information and Systems*, E75-D,3, pp. 366-375, 1992.

[27] 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. on Information and Systems*, E75-D,3, pp. 376-382, 1992.

[28] N.L. Owsley, "Adaptive data orthogononalization," *in Proc. IEEE Int. Conf. ASSP*, pp. 109-112, April 1978.

[29] T.W. Anderson, *An introduction to multivariate statistical analysis*. Second Edition, Wiley and Sons, 1984.

[30] T.Kato, *Perturbation Theory for Linear Operators*. Springer Berlin, 1995.

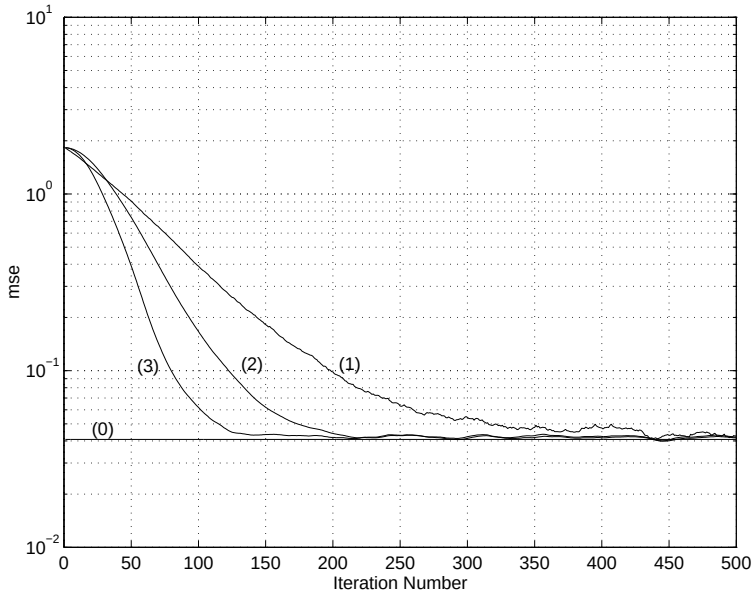


Figure 1: Learning curves of the mean square error $\mathrm{E}\| \mathbf{P}_k - \mathbf{P}_* \|^2_{\mathrm{Fro}}$, averaged over 100 independent runs, for the SNL algorithm (1), and the smoothed SNL algorithm for the following different values of the parameter $\alpha$: $\alpha = 1$ (2), $\alpha = 0.3$ (3), compared to the theoretical value $\gamma \mathrm{Tr}(\mathbf{C}_P)$ (0).

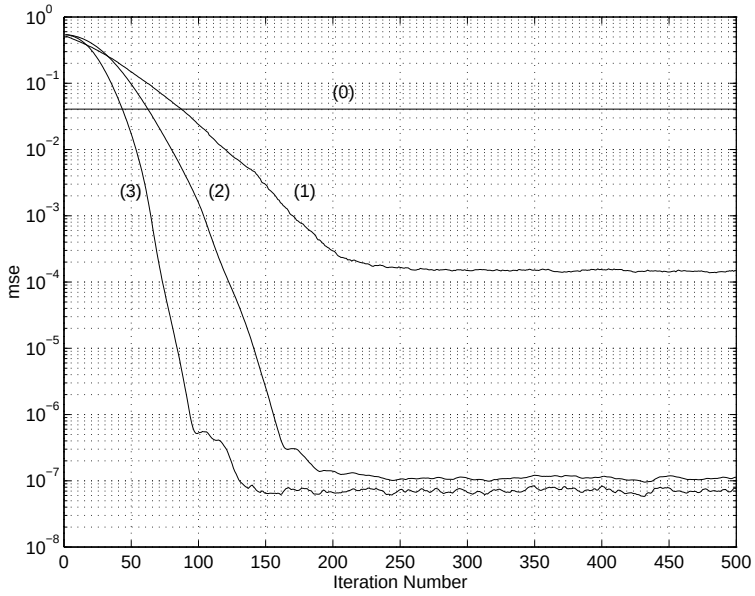


Figure 2: Learning curves of the deviation from orthonormality $\mathrm{E}\| \mathbf{W}_k^T \mathbf{W}_k - \mathbf{I}_r \|^2_{\mathrm{Fro}}$, averaged over 100 independent runs, for the SNL algorithm (1), and the smoothed SNL algorithm for the following different values of the parameter $\alpha$: $\alpha = 1$ (2), $\alpha = 0.3$ (3), compared to $\gamma \mathrm{Tr}(\mathbf{C}_P)$ (0).

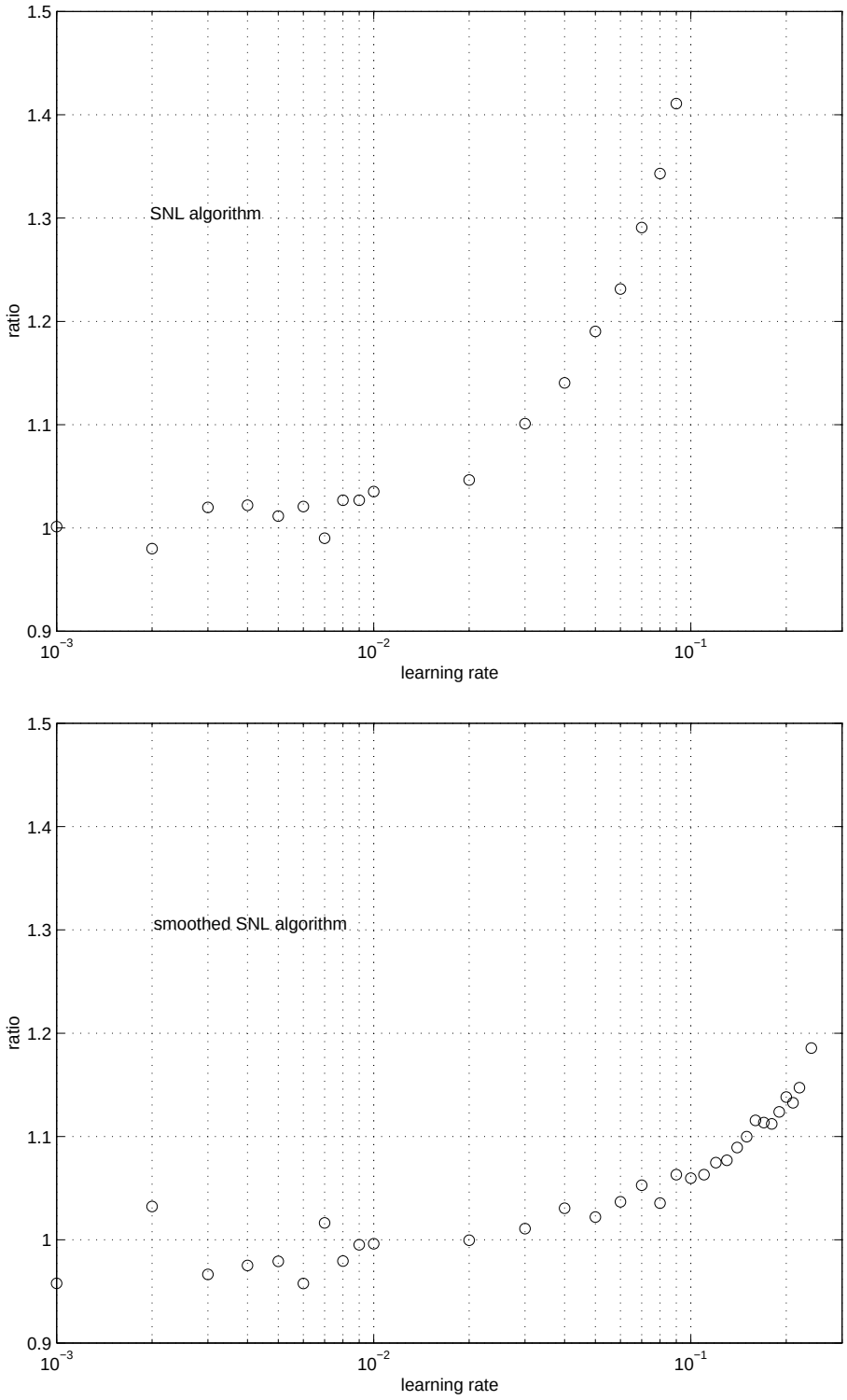


Figure 3: Ratio of the estimated mean square error $\mathrm{E}\| \mathbf{P}_k - \mathbf{P}_* \|^2_{\mathrm{Fro}}$, averaged over 400 independent runs, over the theoretical asymptotic mean square error $\gamma \mathrm{Tr}(\mathbf{C}_P)$, as a function of the learning rate $\gamma$, for both the SNL algorithm and the smoothed SNL algorithm with $\alpha = 1$.

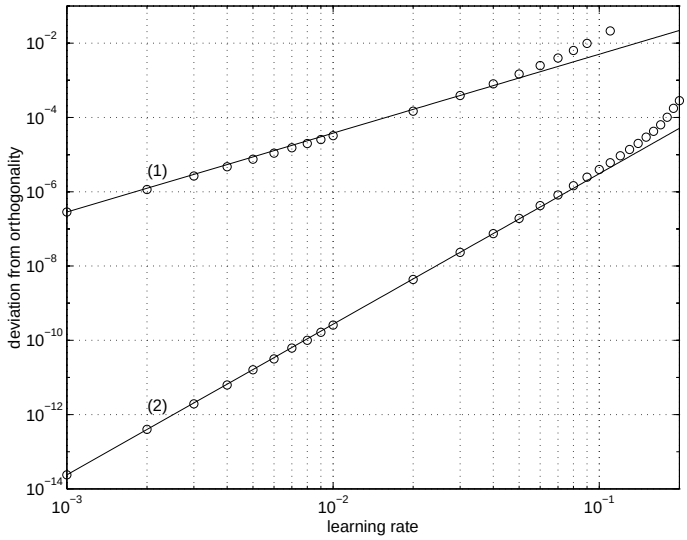


Figure 4: Deviation from orthonormality $d^2(\gamma) \stackrel{\text{def}}{=} \mathrm{E}\| \mathbf{W}_k^T \mathbf{W}_k - \mathbf{I}_r \|_{\mathrm{Fro}}^2$ at "convergence", estimated by averaging 100 independent runs, as a function of the learning rate $\gamma$ in log-log scales, for the SNL (1) and the smoothed SNL algorithm with $\alpha = 1$ (2).

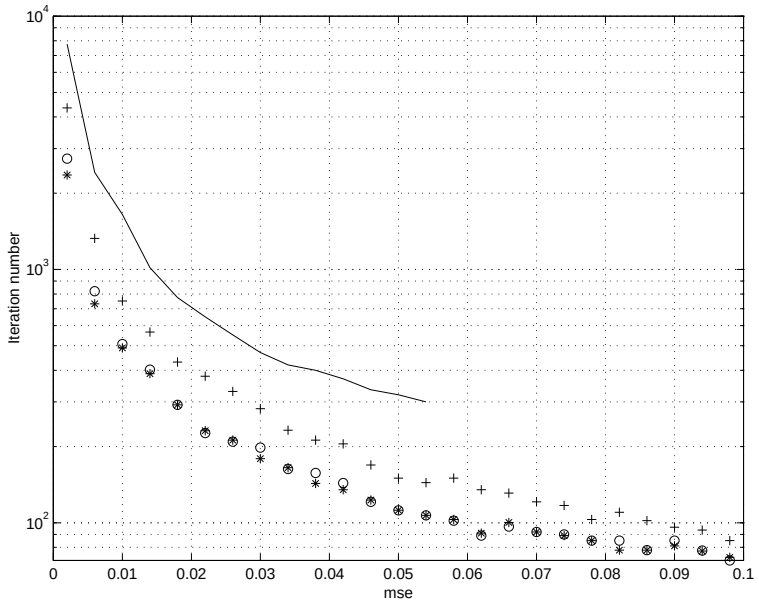


Figure 5: Iteration number until "convergence" is achieved, versus theoretical asymptotic mean square error $\gamma \mathrm{Tr}(\mathbf{C}_P)$, of the SNL algorithm (-), and of the smoothed SNL algorithm with $\alpha = 1$ (+), $\alpha = 0.2$ (*) and $\alpha = 0.3$ (o).

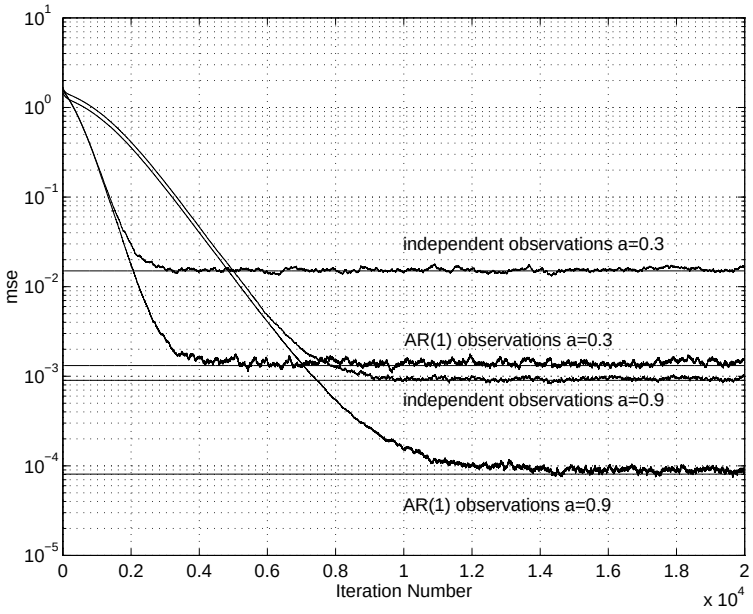


Figure 6: Learning curves of $\mathrm{E}\| \mathbf{P}_k - \mathbf{P}_* \|^2_{\mathrm{Fro}}$ compared to $\gamma \mathrm{Tr}(\mathbf{C}_P)$ averaging 100 independent runs for real independent or AR(1) learning patterns and for the parameter $a = 0.3$ and $a = 0.9$ for the SNL algorithm with $\gamma = 0.005$.

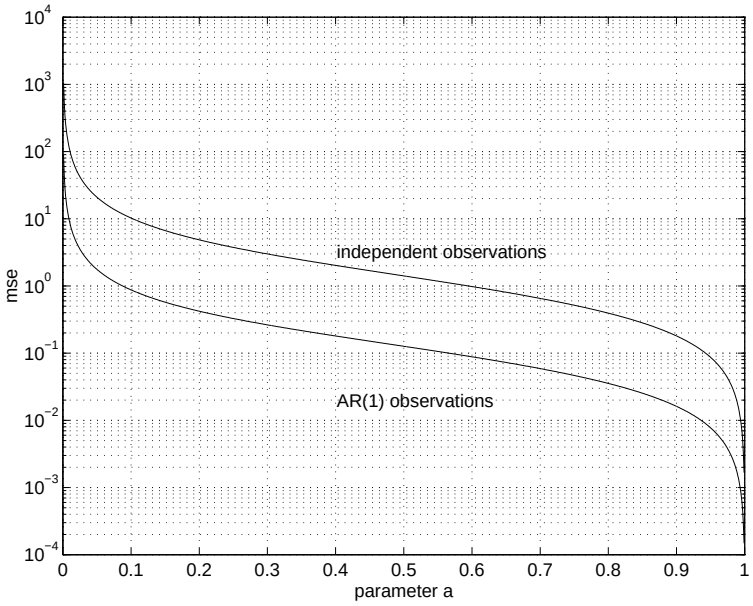


Figure 7: Mean square error of the projection matrix (normalized by the gain factor $\gamma$) on the eigenspace generated by the first two eigenvectors for independent or AR(1) consecutive learning patterns $\mathbf{x}_k$ for the same covariance matrix $\mathbf{R}_x$ as a function of the parameter $a$.