A Bias-Variance Dilemma in Joint Diagonalization and Blind Source Separation

Bijan Afsari
Department of Applied Mathematics
University of Maryland, College Park
Email: bijan@umd.edu

***Abstract*—We identify and explain a bias-variance dilemma which exists in the problem of approximate matrix Joint Diagonalization (JD) as well as in many related Blind Source Separation (BSS) problems. We consider solving a blind identification problem based on JD, where at least one of the matrices under JD is positive definite. We then compare two methods to solve the problem: The first method consists of the so-called Hard-Whitening (HW) followed by Orthogonal JD (OJD), and the second method is based on Non-Orthogonal JD (NOJD). We identify a bias-variance trade-off in this problem, and argue that there is a region depending on the noise level, the number of sources and the number of (statistics) matrices used in the JD process, where the method based on OJD can have less estimation error than the one based on NOJD, while the former always has higher estimation bias than the latter. Simulations support the arguments presented. We also report a constraint proposed in the literature which might be helpful in finding a good trade-off point between bias and variance.**

I. Introduction

Approximate Matrix Joint Diagonalization or more briefly Joint Diagonalization (JD) has found many applications in blind signal processing methodology, and many algorithms for this problem have been proposed in the past several years. Simply yet vaguely phrased, a JD algorithm tries to find a non-singular $n \times n$ matrix B such that it jointly diagonalizes a set of N given $n \times n$ symmetric matrices $\{C_i\}_{i=1}^N$ "as much as possible". Here diagonalization is meant in the sense of congruence i.e., BC_iB^T are ought to be "as diagonal as possible", where B^T is the transpose of B. The matrices to be jointly diagonalized are usually statistics matrices (e.g., covariance matrices or cumulant slice matrices) formed from the observed data. The joint diagonalizaton problem can be divided in two categories with remarkably different properties: Orthogonal Joint Diagonalization or OJD and Non-Orthogonal Joint Diagonalization or NOJD. Historically, OJD was introduced before NOJD as a part of the JADE algorithm [1]. In JADE and the later algorithms such as SOBI [2], first a Hard-Whitening (HW) step is applied to the data and the remaining orthogonal part of the un-mixing matrix is found through an OJD step. It is well-known that this approach leaves an unresolved bias. To combat the bias the idea of NOJD was proposed in [3]. Here, we shall describe a scenario where NOJD due to high variance, and despite its low bias, can result in a high total estimation error in the underlying BSS problem. Our arguments are based on the sensitivity analysis developed in [4]. We mention that another bias-variance dilemma whose cause is the large errors in estimating cumulants in small sample sizes has been observed in [3].

The organization of the paper is as follows: In Section II, we give a brief and rather abstract explanation of the JD problem as an estimation tool in the context of BSS. In Section III, we recall the main differences between OJD and NOJD in terms of sensitivity or variance in estimation. In Section IV we explain the underlying bias-variance dilemma. In Section V, we perform some simulations which support the arguments in Section IV. We conclude the paper in Section VI with conclusions and suggestions for future work.

II. Why We Do Joint Diagonalization?

In many blind identification problems the underlying physics leads to a model in which we have N symmetric matrices $\{C_i\}_{i=1}^N$ of the form

$$C_i = A\Lambda_i A^T, \quad 1 \leq i \leq N, \tag{1}$$

where A is the non-singular $n \times n$ mixing matrix and Λ_i is a diagonal matrix. In a typical problem A and Λ_i's are unknown, and C_i's are statistics matrices associated with an observable signal. The goal is to estimate A or its inverse based on the set $\{C_i\}_{i=1}^N$ in order to achieve separation of sources. Note that (1) means that $A^{-1}C_iA^{-T} =$ diagonal for $1 \leq i \leq N$. This suggests that if we could find a matrix B, such that $B^{-1}C_iB^{-T} =$ diagonal for $1 \leq i \leq N$, then we might conclude that $B = A^{-1}$ up to row permutation and scaling which is usually an acceptable ambiguity. If that is the only ambiguity, then we say that A^{-1} and B are essentially equal. On the other hand, the mentioned conclusion cannot be valid always and one needs certain uniqueness conditions to guarantee the essential equality of A^{-1} and B. In practice, due to noise or estimation errors or because our underlying model is not accurate we only have

$$C_i \approx A\Lambda_i A^T, \quad 1 \leq i \leq N. \tag{2}$$

The main idea behind approximate joint diagonalization is that if B is such that $BC_iB^T \approx$ diagonal for $1 \leq i \leq N$, then B is a good (essential) estimate of A^{-1} and the more matrices we use the better the estimate will be. We emphasize that the JD problem is defined in the context of an estimation problem for which bias and variance can be defined. The exact analysis of this estimation problem is very difficult, and we are not aware of any accurate such analysis. As mentioned earlier OJD and

NOJD are two different forms of JD. In the OJD problem A and B are assumed to be orthogonal, whereas in the NOJD problem more freedom is allowed and they are assumed to be only non-singular. As one expects NOJD can be much more difficult than OJD both in algorithmic sense (as evidenced by so many different algorithms proposed for NOJD) and in the sense of the behavior of the solution (see the next section and [5] for more details). Also for recent trends on NOJD algorithms see [6], [7] and [8].

III. Why Are NOJD and OJD Different?

Working with an orthogonal matrix (because its condition number is 1) is easier than a non-orthogonal matrix. This can be one source of the difference between OJD and NOJD; however, the main difference between OJD and NOJD is that OJD is generically a well-defined problem when we have only $N = 1$ symmetric matrix and adding more matrices helps combat the noise in an averaging process. By a generic matrix here we mean a matrix whose eigenvalues are distinct. The diagonalization of a single symmetric matrix by an arbitrary matrix is not a well-defined problem, unless we assume the diagonalizer is an orthogonal matrix. However, by inclusion of $N = 2$ symmetric matrices we can define a well-defined simultaneous or joint diagonalization problem. Although, with only two matrices the NOJD problem can have a unique solution but still it might be ill-conditioned, by which we mean that a small change in the matrices can change the joint diagonalizer significantly. As many other problems the uniqueness and sensitivity issues of the JD problem are closely related. If the underlying noise-free problem has a unique solution, then the noisy version of the problem can be robust to noise; but this is not an abrupt process, i.e., there is a region where the underling noiseless problem has unique solution while the solution is not "unique enough". This leads to a no-robust solution for the noisy problem. For the OJD this issue of ill-conditioning is not as severe as is for the NOJD problem (see [5] for discussion on this issue). To measure the ill-conditioning of the NOJD problem an important parameter is the so-called modulus of uniqueness introduced in [4]. Let $\{C_i\}_{i=1}^N$ be as in (1). Denote by Λ an $N \times n$ matrix whose i^{th} row is the diagonal of Λ_i. The modulus of uniqueness for $\{C_i\}_{i=1}^N$ is defined as the absolute value of the cosine of the angle between the columns of Λ and is denoted by ρ. Obviously, ρ measures the co-linearity of the columns of Λ. One can show that if $\rho < 1$, then $BC_iB^T = \text{diagonal}(1 \leq i \leq N)$ implies that BA is essentially a diagonal matrix; hence, the underlying noiseless NOJD problem has an essentially unique solution. As one expects and is shown in [4], if ρ is close to 1, then the noisy NOJD problem will be ill-conditioned and finding a B which approximately diagonalizes all the statistics matrices will not yield a good estimate of the unknown un-mixing matrix A^{-1}. The generic behavior of ρ in terms of n and N is quite interesting. If N is small and n is large, then ρ will be very close to 1 (e.g., if $N = 2$ and $n = 40$, then $\rho > 0.997$ [4]). However, as N increases ρ drops very fast in such a way that a logarithmic relation between N and n will ensure acceptable ρ or well-conditioned NOJD problem.

IV. A Bias-Variance Dilemma

In many occasions, we can assume that at least C_1 is positive definite, e.g., if C_1 is a covariance matrix. Now, assume that our matrices in (1) are contaminated with noise according to this model

$$C_i(t) = A\Lambda_i A^T + tE_i, \quad t \in [-\delta, \delta], \delta > 0, \tag{3}$$

where t measures noise amplitude or gain. We assume that E_1 and δ are such that $C_1(t)$ remains positive definite. One can use this model to model noise, finite sample effects or error in the original model. Two approaches based on JD have been proposed to estimate A (or A^{-1}): First approach which is a part of the JADE [1] and SOBI algorithms [2], is based on the idea of the so-called Hard-Whitening (HW). For this purpose first we find $C_1^{\frac{1}{2}}(t)$, the square root of C_1, and then find a new set of matrices of the form

$$\hat{C}_i(t) = C_1^{-\frac{1}{2}}(t)C_i(t)C_1^{-\frac{T}{2}}(t) = \hat{A}\Lambda_i\hat{A}^T + t\hat{E}_i, \tag{4}$$

where $\hat{A} = C_1(t)^{-\frac{1}{2}}A$, and $\hat{E}_i$ is also found accordingly. Note that $\hat{C}_1 = I$ where I is the identity matrix. If noise is not very strong (i.e., δ is small), then $\hat{A}$ is close to an orthogonal matrix. This step is called Hard-Whitening. In the SOBI or the JADE algorithm this step is applied to the data rather than to the statistics matrices directly; however, this will not have significant effect on our discussion. The next step in estimating A^{-1} or an un-mixing matrix is to jointly diagonalize the set $\{\hat{C}_i\}_{i=1}^N$ in an OJD process. As mentioned in [9] and [3], with the HW we are introducing a bias in the estimation which is not removable even when we add more matrices, i.e., increase N or add more samples. The reason is that we are assuming $\hat{A}$ to be orthogonal while it is not. To avoid this bias, Yeredor suggested to use the other method which is the NOJD of $\{C_i(t)\}_{i=1}^N$ [3]. In this method we try to jointly diagonalize $\{C_i(t)\}_{i=1}^N$. One can show that if $\rho < 1$ and if E_i's (for $2 \leq i \leq N$) are i.i.d. with zero mean, then the estimation error will go to zero as N increases. Also when all E_i's are positive definite even the estimate produced by the NOJD process will be biased; however, the source of the bias is the E_i's being of non-zero mean. To see these facts please consult [4]. A vulnerability of NOJD is that when the modulus of uniqueness ρ is close to unity, then the NOJD problem becomes very sensitive or ill-conditioned, and the power or variance of the noise will be amplified in the estimation process. In particular, as mentioned before, when n is large and N is small, ρ can be very close to unity and may result in an ill-conditioned NOJD problem. On the other hand the OJD process, as mentioned before, does not generically suffer from this form of ill-conditioning, and it is much more robust and has less estimation variance. Therefore, when N is small and n is large we have a bias-variance dilemma in estimating A^{-1} in the model (3). Recall that any estimation error has two components due to the bias and variance of the estimation. Note that if noise is not very weak (i.e., t is not close to zero),

as N increases the bias-variance dilemma dilutes very fast since the modulus of uniqueness, in a generic case, improves very fast by N; and NOJD will become the favorable method. Moreover, as t increases the bias in the HW method also increases; therefore, most likely for strong levels of noise also NOJD should be the favorable method, even in the case of large ρ. In summary, sensitivity analysis of the NOJD and OJD problems predicts that there must be a region in terms of n, N and t in which HW followed by OJD will outperform NOJD in estimating A in the model (3). The exact investigation of this bias-variance dilemma requires an accurate bias-variance analysis of the related estimation problem. However, in the next section we perform some experiments which support the qualitative analysis we gave.

V. Experimental Observations

Now, we perform some numerical simulations to observe the mentioned bias-variance trade=off. We are interested in the situation where the method based on OJD outperforms the method based on NOJD. For this we generate some matrices according to (3) and apply the two methods to estimate A^{-1}. We should mention that as many other numerical problems the observed performance depends both on the conditioning of the problem and the algorithm used; and for this reason it can be rather hard to distinguish the source of a low quality of the performance. This is especially true for NOJD algorithms which show very different behavior in different scenarios. For this reason we choose three different NOJD algorithms QRJ2D, FFDIAG and UWEDGE introduced in [10], [11] and [6], respectively. Due to space limitations we do not include more experiments which deal with actual data rather than just the estimated matrices.

Example 1: In the first example we generate matrices according to model (3), change t and N, and find the matrix B with three methods: First, HW applied to the matrices followed by OJD and we denote this method by HW+OJD; second, we do NOJD using the QRJ2D algorithm and denote the method by QRJ2D; and third, we do NOJD using the FFDIAG algorithm and denote it by FFDIAG. More specific settings are: We set $n = 40$ and try $N = 2, 3, 4, 5, 10, 40$ and $t = 0, 10^{-6}, 10^{-5}, 10^{-4}$. We choose a random A with condition number of around 10 so that the conditioning of A does not act as a major factor. The Λ_i's are of i.i.d. diagonal elements uniformly distributed in the interval $[0, 1]$. We generate $E_1 = N_1 N_1^T$ (to make sure C_1 is positive definite) and normalize it such that its Frobenius norm $\|E_1\|_F$ is 1. Here, N_1 is a matrix of i.i.d. entries with standard normal distribution. For $2 \leq i \leq N$ we set $E_i = N_i + N_i^T$ and again normalize it such that $\|E_i\|_F = 1$, where N_i is a random matrix with similar distribution as N_1. We find B, the estimate of A^{-1}, via the three mentioned methods, and repeat the experiment $K = 100$ times for each value of N and t. The performance is measured by

$$\text{Index}(P) = \sum_{i=1}^{n}\Big(\sum_{j=1}^{n}\frac{|p_{ij}|}{\max_k |p_{ik}|}-1\Big)+\sum_{j=1}^{n}\Big(\sum_{i=1}^{n}\frac{|p_{ij}|}{\max_k |p_{kj}|}-1\Big) \tag{5}$$

with $P = BA$. Note that Index$(BA) \geq 0$ and equality happens only when B is essentially equal to A^{-1}. We find the median of the K Index values for each set of variables and Figure 1 shows the graphs of Index(BA) in terms of N for different values of noise gain t. We can explain the

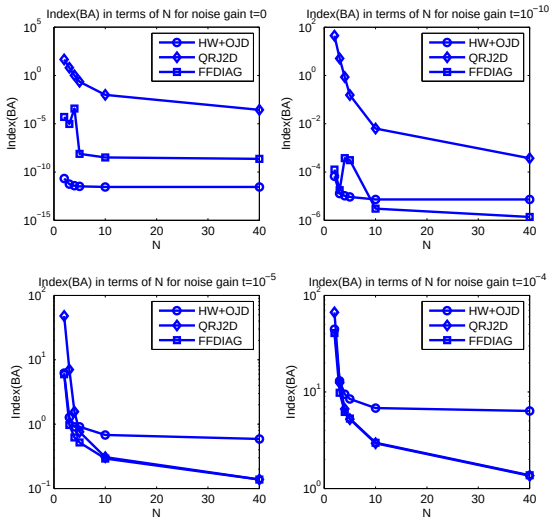

Fig. 1. Comparison of the performance of three methods HW+OJD, QRJ2D and FFDIAG in terms of N as noise level varies in Example 1.

graphs based on the bias-variance trade-off explained before. For $t = 0$ since the bias introduced by HW is zero, the only source of error is finite precision numerical errors; and we see that HW+OJD outperforms NOJD based methods, because the NOJD methods are numerically more sensitive. As t increases, e.g., if $t = 10^{-10}$, the situation changes little bit, for small N still HW+OJD outperforms the other two mrthods, but for $N \geq 10$, FFDIAG outperforms the other two methods. Note that QRJ2D underperforms both of the methods. We do not know its exact source, but it might be related to certain normalization in the QRJ2D algorithm which can cause loss of accuracy. Although, we have not shown it, but when one includes much more matrices (i.e., when conditioning improves) and if noise is not too weak this problem becomes less visible. For stronger noise levels, such as $t = 10^{-5}$ we see that at $N = 2$, HW+OJD and FFDIAG have very similar performances but as N increases again FFDIAG prevails. This time the method using QRJ2D also catches up faster, and soon both the NOJD based methods have close performances. For $t = 10^{-4}$ the bias is so strong that even for $N = 2$, HW+OJD has worse performance than FFDIAG. Of course, for both methods the Index(BA)'s are very high. In the graphs for $t = 10^{-5}$ and $t = 10^{-4}$ we clearly see the harmful effect of the bias that HW causes and cannot be removed by increasing N, whereas the performance of NOJD based methods by increasing N improves significantly.

Example 2: In our simulations when we used the recently proposed UWEDGE algorithm [6] for NOJD, the bias-variance

trade-off when N was above 2 was not as severe as the one in Example 1. Specifically, the performance of UWEDGE for $N = 3$ or $N = 4$ was not very bad. It seems that a re-normalization constraint (proposed in [12]) included in the UWEDGE algorithm and modified in such a way that the algorithm can accommodate non-positive definite matrices, is responsible for this behavior. The mentioned constraint, requires the rows of B, in each iteration of the algorithm, to be normalized such that the absolute values of the diagonal elements of BC_1B^T are one. Note that this constraint is automatically satisfied in the first method, i.e., HW+OJD, where BC_1B^T is identity. But in the UWEDGE algorithm, BC_1B^T will not be diagonal, necessarily. Nevertheless, it seems that for small N the solutions from UWEDGE and HW+OJD are very close to each other. Also the implication of this constraint is that UWEDGE will give a biased answer. It seems that the bias is not significant when N becomes large while for small N the bias helps to reduce the variance and hence the total error compared to when FFDIAG or QRJ2D is used. The graphs in Figure 2 show the results of the experiments similar to our previous ones except that here in addition to HW+OJD, we use NOJD using UWEDGE and its version with no normalization denoted by UWEDGE-NN. As one can see UWEDGE-NN performs similarly to FFDIAG; and for small N, UWEDGE and HW+OJD perform similarly. It is interesting to note that difference between the performance of UWEDGE and UWEDGE-NN for large values of N as t increases is not significant, although for large enough N and higher t, due to the bias, UWEDGE-NN slightly outperforms UWEDGE. Therefore, the bias introduced by the re-normalization in UWEDGE is not significant while it significantly reduces the variance (and hence the error) of the estimation when N is small.

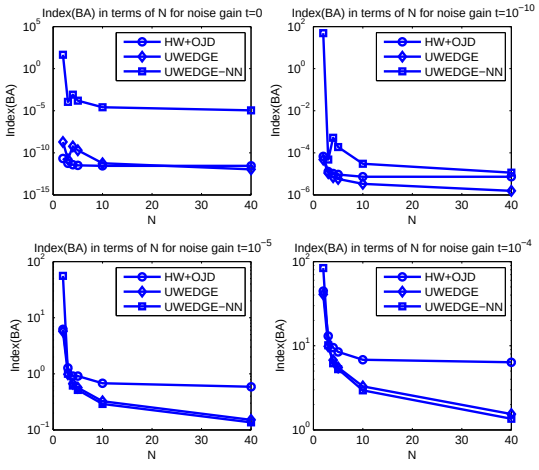

Fig. 2. Comparison of the performance of three methods HW+OJD, UWEDGE and UWEDGE-NN in terms of N as noise level varies in Example 2.

VI. Conclusions and Future Research

We described a bias-variance dilemma which exists in the problem of JD when at least of one the matrices is positive definite. We identified a region depending on the noise level, the number and the dimension of the matrices, where the NOJD method which introduces low bias can result in more estimation error than hard-whitening followed by OJD. In our experiments also we found that a normalization used in the UWEDGE (and QDIAG [12]) algorithms can serve as a good method to bring about a good trade-off between bias and variance. Our arguments and analysis were based on the sensitivity analysis of the JD problem, and further research is needed to investigate the bias-variance dilemma based on detailed statistical analysis. Also whether the mentioned bias-variance trade-off can have practical implications should be investigated further. In our experiments the high variance became significant when we had low to moderate level of noise and very few very large matrices, which in some practical problems might be a realistic scenario.

Acknowledgment

I would like to thank Dr. Jeff O'Connell. Discussions with him motivated me to study the bias-variance dilemma in the hope of comparing NOJD and OJD methods. I also thank the reviewers for their helpful comments.

References

[1] J. F. Cardoso and A. Soloumiac, "Blind beamforming for non-gaussian signals," *IEE Proceedings-F*, vol. 140, no. 6, pp. 362–370, Dec 1993.

[2] A. Belouchrani, K.A. Meraim, J.-F. Cardoso, and E. Moulines, "A blind source separation technique using second-order statistics," *IEEE Transactions on Signal Processing*, vol. 45, no. 2, pp. 434–444, Feb. 1997.

[3] A. Yeredor, "Approximate joint diagonalization using non-orthogonal matrices," in *Proceedings of ICA2000*, Helsinki, June 2000, pp. 33–38.

[4] B. Afsari, "Sensitivity analysis for the problem of matrix joint diagonalization," *SIAM. J. Matrix Anal. & Appl.*, vol. 30, no. 3, pp. 1148–1171, 2008.

[5] B. Afsari, "What can make joint diagonalization difficult?," in *Proceedings of ICASSP07*, Honolulu, HI, April 2007, vol. 3, pp. III–1377–III–1380, IEEE.

[6] P. Tichavský and A. Yeredor, "Fast approximate joint diagonalization incorporating weight matrices," *IEEE Transactions on Signal Processing*, vol. 57, no. 3, pp. 878–891, March 2009.

[7] X. L. Li and X. D. Zhang, "Nonorthogonal joint diagonalization free of degenerate solution," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 1803–1814, May 2007.

[8] A. Souloumiac, "Nonorthogonal joint diagonalization by combining givens and hyperbolic rotations," *IEEE Transactions on Signal Processing*, vol. 57, pp. 2222–2231, June 2009.

[9] J.-F. Cardoso, "On the performance of orthogonal source separation algrithms," 1994, pp. 776–779.

[10] B. Afsari, "Simple LU and QR based non-orthogonal matrix joint diagonalization," in *Proceedings of ICA2006*. March 2006, pp. 1–7, Springer.

[11] A Ziehe, P Laskov, G Nolte, and K R Mueller, "A fast algorithm for joint diagonalization with non-orthogonal transformation and its application to blind source seperation," *Journal of Machine Learning Research*, vol. 5, pp. 777–800, 2004.

[12] K. Vollgraf and R. Obermayer, "Quadratic optimization for simultaneous matrix diagonalization," *IEEE Transactions on Signal Processing*, vol. 54, no. 9, pp. 3270–3278, Sept 2006.