Exploiting Covariance-domain Sparsity for Dimensionality Reduction

Ioannis D. Schizas*, Georgios B. Giannakis*, and Nicholas D. Sidiropoulos†
*Dept. of ECE, University of Minnesota, Minneapolis, MN, USA
† Dept. of ECE, Technical University of Crete, Chania, Greece

Abstract—Novel schemes are developed for linear dimensionality reduction of data vectors whose covariance matrix exhibits sparsity. Two types of sparsity are considered: i) sparsity in the eigenspace of the covariance matrix; or, ii) sparsity in the factors that the covariance matrix is decomposed. Different from existing alternatives, the novel dimensionality-reducing and reconstruction matrices are designed to fully exploit covariance-domain sparsity. They are obtained by solving properly formulated optimization problems using simple coordinate descent iterations. Numerical tests corroborate that the novel algorithms achieve improved reconstruction quality relative to related approaches that do not fully exploit covariance-domain sparsity.

I. INTRODUCTION

Sparsity is an often-encountered property in a broad range of signal processing tasks including sampling, compression, and statistical inference. For example, sparsity has been exploited to solve under-determined linear systems of equations with applications to sub-Nyquist sampling [2]; and also to select variables in linear regression applications using e.g., the least-absolute shrinkage and selection operator (Lasso) [3]. More recently, sparsity has been utilized also for principal component analysis (PCA), where sparse variable selection determines the “instrumental variables” describing the data parsimoniously. To this end, the approach in [4] expanded the standard PCA criterion with a norm-one penalty (as in Lasso) to induce sparsity. Follow-up advances offered computationally efficient sparse PCA algorithms [5]–[8].

Here we explore ways of exploiting sparsity in the covariance domain for efficient dimensionality reduction (DR). Two different types of sparsity are considered: the covariance matrix has either i) a sparse eigenspace, or, ii) it admits a sparse factorization (see Sec. II). Unlike [4]–[7], our sparse eigenspace or factor estimates are derived from judiciously formulated optimization problems that not only account for the reconstruction mean-square error (MSE), but also fully exploit covariance-domain sparsity. Factor estimates are then obtained using simple coordinate descent updates that converge monotonically, at least to a stationary point (Sec. III). Numerical examples illustrate that the proposed algorithms achieve a better trade-off between DR and reconstruction MSE than earlier approaches which do not fully exploit covariance-domain sparsity (Sec. IV).

II. PRELIMINARIES AND PROBLEM STATEMENT

Suppose that the data vector \( x \in \mathbb{R}^{p \times 1} \), whose dimensionality needs to be reduced, is generated by a zero-mean memoryless source with unknown covariance matrix \( C_x \). Consider the eigen-decomposition \( C_x = U_x \Lambda_x U_x^T \), where \( U_x \) contains the eigenvectors and \( \Lambda_x \) the corresponding eigenvalues (\( ^T \) denotes transposition). Assume further that \( C_x \) either has a sparse eigenspace, or, it can be decomposed into sparse factors. There are practical setups that give rise to such covariance matrices. Two motivating scenarios are:

S1) The entries of \( x \) can be partitioned into groups of random variables, say \( G_1, \ldots, G_K \), such that for the \( j \)-th element of \( x \), namely \( x(j) \), there exists a unique \( k \in \{1, \ldots, K\} \) for which \( x(j) \in G_k \). The inter-correlation between random variables that belong to different groups is zero, i.e., if \( x(j) \) and \( x(j') \) belong to \( G_k \) and \( G_{k'} \) respectively and \( k \neq k' \) then \( E[x(j)x(j')] = 0 \). It readily follows that there exists a permutation matrix \( P \), so that the random vector \( x_P := Px = [x_{G_1}, \ldots, x_{G_K}]^T \) has covariance with block diagonal structure, namely \( C_{x_P} = PC_xP^T = \text{bdig}(C_{x_{G_1}}, \ldots, C_{x_{G_K}}) \). This implies that the eigenvector matrix \( U_{x_P} \) of \( C_{x_P} \) is block diagonal and sparse. Since \( U_{x_P} = P^T U_{x_P} \) and \( P \) is a permutation matrix it follows that \( U_{x_P} \) is also sparse.

Vector \( x \) under S1 may correspond to field measurements acquired by a sensor network. These sensor measurements are collected by a cluster head, and are further compressed before being transmitted to a fusion center. Groups \( G_k \) represent different sensor neighborhoods located sufficiently apart, whereas the members of each \( G_k \) are the correlated sensor measurements acquired within the corresponding neighborhood.

S2) The data vector \( x \) obeys the linear model \( x = Hs + n \), where \( s \in \mathbb{R}^{q \times 1} \) is a vector of zero-mean uncorrelated hidden entries with diagonal covariance matrix \( \Lambda_s \). Matrix \( H \) is assumed to be sparse, not necessarily orthogonal, and the noise \( n \) is zero-mean white with covariance \( \sigma_n^2 I_p \), where \( I_p \) denotes the \( p \times p \) identity matrix. The model parameters \( H, \Lambda_s, \sigma_n^2 \), and the parameter vector \( s \) are unknown, while \( q \) is available. For example, \( x \) may be data acquired and compressed by underwater sensors that acquire noisy observations of \( s \) through an unknown sparse underwater channel. The covariance of \( x \) can be written as

\[
C_x = H \Lambda_s H^T + \sigma_n^2 I = \sum_{i=1}^{q} \lambda_{s,i} h_i h_i^T + \sigma_n^2 I
\]
where \( I_i \) denotes the sparse \( i \)-th column of \( H \) and \( \lambda_{s,i} \) the \( i \)-th diagonal entry of \( A_s \).

Linear DR is performed by multiplying \( x \) with a fat matrix \( C \in \mathbb{R}^{r \times p} \), where \( r < p \). The compressed vector \( Cx \) at the encoder output is received by the decoder which multiplies it with a tall \( p \times r \) matrix \( B \) to form the reconstructed vector \( \hat{x} = BCx \). The optimal matrices \( B_o \) and \( C_o \), minimizing the reconstruction MSE

\[
J_{rec}(B, C) := E[\|x - BCx\|^2]
\]

are given as \( B_o = U_{x,r} \), \( C_o = U^T_{r,r} \), where the columns of \( U_{x,r} \) are the \( r \) dominant eigenvectors of \( C_x \); see e.g., [1, Ch. 9]. Note that in S2, the presence of noise, suggests that a more pertinent approach is to determine \( B, C \) such that the estimation MSE for the signal component \( Hs \), i.e., \( J_{est}(B, C) := E[\|Hs - BCX\|^2] \), is minimized. Interestingly, it holds that \( B_o, C_o \) are also the minimizers of \( J_{est}(B, C) \).

Since \( C_x \) is not available, the ‘ensemble criterion’ in (2) is not applicable. The standard alternative is to learn the unknown covariance using a set of training data \( \{x_i\}_{i=1}^n \). Specifically, the MSE cost \( J_{rec}(B, C) \) is replaced with its ‘sample-average’ version, and the corresponding \( B, C \) matrices are determined as [1, Ch. 9]

\[
(B, C) = \arg\min_{B,C} \frac{1}{n} \|X - BCX\|^2_F
\]

where \( \| \cdot \|_F \) denotes the Frobenius norm, and \( X := [x_1 \ldots x_n] \in \mathbb{R}^{p \times n} \) contains the training data. As expected, \( B = C^T = U_{x,r} \), where \( U_{x,r} \) is now formed by the \( r \) principal eigenvectors of the sample covariance matrix \( \hat{C}_x = n^{-1} \sum_{i=1}^n x_i x_i^T \).

The PCA approach summarized in (3) does not exploit the sparsity present in \( C_x \) under scenarios S1 and S2. Existing ‘sparse’ PCA schemes [4]–[7] do not take full advantage of the sparsity present in \( C_x \), especially when it comes to DR. The goal here is to exploit the sparsity in \( C_x \) to derive improved estimates \( B, C \) of the ‘clairvoyant’ matrices \( B_o, C_o \), thereby achieving a better trade-off between the reduced-dimension \( r \) and the MSE cost \( J_{rec}(\cdot) \) in S1, or \( J_{est}(\cdot) \) in S2 than existing alternatives. Towards this end, two novel sparsity-aware algorithms are proposed in the next section.

### III. Covariance-Domain Sparsity: Algorithms

Similar to Lasso-based sparse regression [3], as well as existing sparse PCA approaches [4]–[7], we utilize penalty functions based on the \( \ell_1 \) norm to exploit sparsity. Unlike the aforementioned approaches, however, we fully exploit the sparsity present in \( C_x \).

#### A. Sparsity-Aware Dimensionality Reduction under S1

Under S1, we want to determine compression and reconstruction matrices \( B \) and \( C \) so that the reconstruction error \( n^{-1}\|X - BCX\|^2_F \) is minimized, while both \( B \) and \( C \) are sparse. This is motivated by the fact that under S1 the ‘clairvoyant’ matrices \( B_o \) and \( C_o \) are sparse. Then, pertinent estimates are obtained via

\[
(B_1, C_1) = \arg\min_{B,C} n^{-1}\|X - BCX\|^2_F + \delta\|B - C^T\|^2_F + \sum_{i=1}^r \sum_{j=1}^p \lambda_{i,j} (|C(i,j)| + |B(j,i)|)
\]

where \( \lambda_{i,j} \), for \( i, j = 1, \ldots, p \), are nonnegative constants that control the sparsity of \( B \) and \( C \). The penalty term \( \delta\|B - C^T\|^2_F \) ensures that \( B_1, C^T_1 \) are close. This is in compliance with the fact that the ‘clairvoyant’ compression and reconstruction matrices satisfy \( B_o = C^T_1 \).

The objective function in (4) resembles the sparse PCA one introduced by [5]. However, the cost function in [5] imposes sparsity only on \( C \), while it forces matrix \( B \) to be orthonormal. The latter constraint mitigates scaling issues (otherwise \( C \) could be made arbitrarily small by counter-scaling \( B \), but is otherwise not necessarily well motivated. The formulation in [5] does not fully exploit the sparsity present in the eigenspace of \( C_x \), which further results in sparse ‘clairvoyant’ matrices \( B_o \) and \( C_o \). Thus, it is pertinent to combine the reconstruction error \( n^{-1}\|X - BCX\|^2_F \) with penalty terms that impose sparsity to both \( B \) and \( C \).

The minimization problem given in (4) is nonconvex with respect to (wrt) both \( B \) and \( C \). We bypass this obstacle by constructing an iterative minimization algorithm which converges at least to a stationary point of the cost in (4). Relying on coordinate descent (see e.g., [9, pg. 160]), the cost in (4) is iteratively minimized wrt an element of either \( B \) or \( C \), while keeping the remaining elements fixed. One iteration cycle involves updating all the elements of matrices \( B \) and \( C \).

We have also used block coordinate descent, in which, e.g., \( B \) is updated as a whole given \( C \), and vice-versa; but we have found that element-wise descent is almost as effective and far cheaper computationally.

Let \( \otimes \) denote Kronecker product, and \( \text{vec}(C) \) the \( rp \times 1 \) vector obtained after stacking the \( p \) columns of \( C \). Using \( \text{vec}(BCX) = (X^T \otimes B)\text{vec}(C) \), eq. (4) yields

\[
(B_1, C_1) = \arg\min_{B,C} \frac{1}{n} \|\text{vec}(X) - (X^T \otimes B)\text{vec}(C)\|^2_F + \sum_{i=1}^r \sum_{j=1}^p \lambda_{i,j} (|C(i,j)| + |B(j,i)|) + \delta\|B - C^T\|^2_F
\]

Let \( X_B := X^T \otimes B \) and \( c_v := \text{vec}(C) \), where \( C(i,j) \) equals the \((j - 1)r + i\)-th element of \( c_v \). Next, we show how to form \( c_v(k) \) at iteration \( k \), relying on the most up-to-date values for \( B \) and \( C_v \), namely \( B_{k-1} \), \( \{c_v(k-1)(m)\}_{m=(j-1)r+1}^{(j-1)r+p} \) and \( \{\hat{c}_v(k)(m)\}_{m=(j-1)r+1}^{(j-1)r+p} \). From (5), it follows that \( C_k(i,j) \approx c_v(k-1)(j - 1)r + i) \), for \( i = 1, \ldots, r \) and \( j = 1, \ldots, p \), can be determined as

\[
\hat{C}_k(i,j) = \arg\min_{c} \frac{1}{n} \|\text{vec}(X_B(k-1)(j-1)r+i) - c\|^2_F + \lambda_{i,j} |c|
\]
where
\[ y_{i,j}^k := \text{vec}(X) - \sum_{m=1}^{(j-1)r+i+1} c_{n,k}(m) \hat{X}_{B,m}^{k-1} - \sum_{m=(j-1)r+i+1}^{r} c_{n,k}(m) \hat{X}_{B,m}^{k-1} \]
while \( \hat{X}_{B,m}^{k-1} \) denotes the \( m \)-th column of \( \hat{X}_{B}^{k-1} := X^T \otimes \hat{B}_{k-1} \). Interestingly, the minimization in (6) corresponds to a scalar sparse regression (Lasso) problem, which is known to accept a closed-form solution given by [3]
\[
\hat{C}_k(i,j) = \text{sgn} \left( \frac{n^{-1}(y_{i,j})^T \hat{X}_{B,(j-1)r+i}^{k-1} + \delta \hat{B}_k(i,j)}{n^{-1}\|\hat{X}_{B,(j-1)r+i}^{k-1}\|^2_2 + \delta} \right)
\times \left( \frac{n^{-1}(y_{i,j})^T \hat{X}_{B,(j-1)r+i}^{k-1} + \delta \hat{B}_k(i,j)}{n^{-1}\|\hat{X}_{B,(j-1)r+i}^{k-1}\|^2_2 + \delta} - \frac{\lambda_{i,j}}{2n^{-1}\|\hat{X}_{C,i}\|^2_2 + 2\delta} \right) +
\] (8)
where \((\cdot)_+ := \max(\cdot, 0)\). Similarly, the update of \( \hat{B}_k(j,i) \) is
\[
\hat{B}_k(j,i) = \text{sgn} \left( \frac{n^{-1}(z_{i,j})^T \hat{X}_{C,i}^{k-1} + \delta \hat{C}_k(i,j)}{n^{-1}\|\hat{X}_{C,i}^{k-1}\|^2_2 + \delta} \right)
\times \left( \frac{n^{-1}(z_{i,j})^T \hat{X}_{C,i}^{k-1} + \delta \hat{C}_k(i,j)}{n^{-1}\|\hat{X}_{C,i}^{k-1}\|^2_2 + \delta} - \frac{\lambda_{i,j}}{2n^{-1}\|\hat{X}_{C,i}\|^2_2 + 2\delta} \right) +
\] (9)
where \( z_{i,j}^k := X^T j - \sum_{t=1}^{r-1} B_k(j,l) \hat{X}_{C,l}^t - \sum_{t=i+1}^{r} \hat{B}_k(j,l) \hat{X}_{C,l}^t \) while \( X_j \) denotes the \( j \)-th row of \( X \), and \( \hat{X}_{C,i}^t \) the \( t \)-th column of \( \hat{X}_{C,i}^t := X^T C_i^T \).

The sparse DR algorithm consisting of the scalar updating recursions in (8) and (9), is tabulated as Algorithm 1. Each iteration \( k \) involves minimizing (4) w.r.t. each entry of \( C_0 \), or \( B \), while treating the rest as fixed. Thus, the corresponding cost per iteration is non-increasing and the algorithm always converges to a stationary point of the cost in (4) [9, pg. 273].

Note that the sparsity coefficient \( \lambda_{i,j} \) is common for both \( C(i,j) \) and \( B(j,i) \). Details on how to set \( \lambda \)'s are given in Sec. IV. This together with the explicit dissimilarity penalty in (4) forces the final estimates obtained via Alg. 1 (denoted \( \hat{B}_\infty \) and \( \hat{C}_\infty \)) to be approximately equal. The ideal \( \hat{B}_o \) and \( \hat{C}_o \) are orthonormal and equal. To ensure that the same properties are satisfied by \( \hat{B}_\infty \) and \( \hat{C}_\infty \), we: i) pick one of the matrices \( \hat{B}_\infty \) and \( \hat{C}_\infty \), say the latter one; ii) extract, via singular value decomposition, an orthonormal basis \( U_{s_1} \in \mathbb{R}^{P\times r} \) that spans the range space of \( \hat{C}_\infty \); and iii) form the compression and reconstruction matrices as \( \hat{B}_{s_1} = \hat{C}_o^T = U_{s_1} \).

B. Sparsity-Aware Dimensionality Reduction under S2

In order to exploit the sparsity present in S2, we first form estimates \( \hat{H} \) and \( \sigma_n^2 \) for the sparse factors in \( \hat{H} \) and the variance \( \sigma_n^2 \), respectively; and subsequently construct the corresponding compression and reconstruction matrices \( \hat{C}_{s_2} \in \mathbb{R}^{r\times P} \) and \( \hat{B}_{s_2} \in \mathbb{R}^{o\times r} \), with \( r \leq q \), as \( \hat{B}_{s_2} = \hat{C}_{s_2}^T = U_{h,r} \), where \( U_{h,r} \) contains the \( r \) dominant left singular vectors of \( \hat{H} \). As will be explained soon, the latter can be used as an estimate of \( B_o = C_o^T = U_{x,r} \).

In order to obtain estimates for \( \hat{H} \) and \( \sigma_n^2 \), we propose the following sparsity-aware formulation:
\[
(\hat{H}_1, \hat{H}_2, \sigma_n^2) = \arg \min_{\hat{H}_1, \hat{H}_2, \gamma \geq 0} \| \hat{C}_x - \hat{H}_1 \hat{H}_2^T - \gamma I_p \|^2_F
\]
\[+ \sum_{j=1}^{p} \sum_{i=1}^{q} \lambda_{j,i} (|H_1(i,j)| + |H_2(j,i)|) + \delta \| H_1 - H_2 \|^2_F, \]
(10)
where the \( \ell_1 \)-norm penalties exploit the sparsity in \( \hat{H} \), while \( \delta \| \hat{H}_1 - \hat{H}_2 \|^2_F \), with \( \delta > 0 \), ensures that \( \hat{H}_1, \hat{H}_2 \) will be close. The latter requirement is motivated by the fact that \( \hat{C}_x = \hat{H}_1 \hat{H}_2^T + \sigma_n^2 I_p \) is an approximation for \( C_x = \hat{H}_1 \hat{H}_2^T \), as follows that the \( r \) dominant left singular vectors of \( C_x \), denoted as \( U_{h,r} \), can be used to estimate the \( r \) dominant eigenvectors of \( C_x \).

Again, the nonconvex minimization problem in (10) is tackled via an iterative algorithm which relies on coordinate descent and converges at least to a stationary point of the cost in (10). Applying similar steps as in Section III-A, we can form updates for the entries of \( \hat{H}_1, \hat{H}_2 \) per iteration \( k \). Specifically, \( \hat{H}_1(i,j), j = 1, \ldots, p \) and \( i = 1, \ldots, q \) can be formed using the recursive formula
\[
\hat{H}_1(i,j) = \text{sgn} \left( \frac{(w_{c,j,i})^T \hat{H}_{2,i}^{k-1} + \delta \hat{H}_{1,i}^{k-1}(i,j)}{\| \hat{H}_{2,i}^{k-1} \|^2_2 + \delta} \right)
\times \left( \frac{(w_{c,j,i})^T \hat{H}_{2,i}^{k-1} + \delta \hat{H}_{1,i}^{k-1}(i,j)}{\| \hat{H}_{2,i}^{k-1} \|^2_2 + \delta} - \frac{\lambda_{j,i}}{2\| \hat{H}_{2,i}^{k-1} \|^2_2 + 2\delta} \right) +
\]
(11)
where \( w_{c,j,i} := \hat{C}_x, j - \sum_{l=1}^{i-1} \hat{H}_1(j,l) \hat{H}_{2,i}^{k-1} - \sum_{l=i+1}^{r} \hat{H}_1(j,l) \hat{H}_{2,i}^{k-1} \) while \( \hat{C}_x \) is: \( \hat{C}_x = \hat{C}_x - \gamma_{k-1} e_e \), and \( \hat{C}_x, \hat{H}_{1,i}^{k-1} \) and \( e_e \) denote the \( j \)-th, \( i \)-th and \( i \)-th columns of matrices \( \hat{C}_x, \hat{H}_{1,i}^{k-1} \) and \( I_p \), respectively. The update formula for \( \hat{H}_1(i,j) \) can be obtained by appealing to the symmetry of (10) wrt \( H_1, H_2 \). Finally, the noise variance estimate \( \gamma_k \) can be determined as \( \gamma_k = p^{-1} \text{max}(\text{trace}(C_x - \hat{H}_1(\hat{H}_2)^T), 0) \). The resulting sparse matrix factorization (SMF) algorithm, consisting of recursion (11), its counterpart for \( \hat{H}_2(i,j) \), and \( \gamma_k \), is tabulated as Algorithm 2.

It is worth stressing that under S2 matrix \( C_x \) does not necessarily have a sparse eigenspace since \( \hat{H} \) may not have orthogonal columns. This prevents existing sparse PCA approaches [4],[5],[7] from taking advantage of the sparsity present in \( \hat{H} \) to form better estimates for \( \hat{B}_o \) and \( \hat{C}_o \). The approach in [8] assumes that \( \hat{H} \) is orthogonal; and ii) has rows with all their entries equal to zero. This is not necessarily true in S2. On the other hand, the SMF algorithm is able to exploit the sparsity of \( \hat{H} \) when forming compression and reconstruction matrices \( \hat{C}_{s_2} \) and \( \hat{B}_{s_2} \). This way SMF has the potential of achieving a smaller estimation MSE \( J_{e_2}(\hat{B}_{s_2}, \hat{C}_{s_2}) \).
Algorithm 1 Sparse Dimensionality Reduction (Sparse DR)
1: Initialize $B_0 = C_0^T = U_{x,r}$.
2: for $k = 1, \ldots$ do
3:   For $j = 1, \ldots, p$ and $i = 1, \ldots, r$ determine $C_k(i, j)$ via (8).
4:   for $j = 1, \ldots, p$ and $i = 1, \ldots, r$ determine $B_k(j, i)$ via (9).
5:     if $|\tau_k - \tau_{k-1}| < \epsilon$ for a prescribed $\epsilon$ then break
6: end for
7: $B_{x} = C_{x}^T = U_{x[r]}$ (see text)

Algorithm 2 Sparse Matrix Factorization (SMF)
1: Initialize $H_0^x = H_0^r = U_{x,r}$.
2: for $k = 1, \ldots$ do
3:   For $j = 1, \ldots, p$ and $i = 1, \ldots, q$ determine $H_k(x, i)$ via (11). Then, do the same for $H_k(r, i)$ and $\tilde{\gamma}_i$.
4:   if $|\tau_k - \tau_{k-1}| < \epsilon$ for a prescribed $\epsilon$ then break
5: end for
6: $B_{x} = C_{x}^T = H_{r}$ (see text)

IV. NUMERICAL EXAMPLES AND REMARKS
Here we test the performance of the sparse DR and SMF under scenario S1 and S2, and compare them with existing alternatives. In S1, we measure the reconstruction MSE $J_{rec}(B_m, C_m)$, where $B_m$ and $C_m$ are formed using i) sparse DR; ii) the ‘clairvoyant’ (ensemble) PCA matrices $B_o, C_o$; iii) the ‘sample’-based approach in (3); iv) a genie-aided PCA which relies on (3) but also knows where the zero entries are located within $U_{x,r}$; v) the sparse PCA (SPCA) approach in [5]; vi) the scheme in [7] abbreviated as SPC; and vii) the algorithm of [6], which is abbreviated as DSPCA. Under S2, we compare the estimation/reconstruction MSE $J_{est}(B_m, C_m)$, achieved by: i) SMF; ii) SMF without $\ell_1$ penalties ($\lambda_{i,j} = 0$); iii) ensemble-based PCA; iv) sample-based PCA; v) SPCA; vi) SPC and vii) DSPCA. In S2 we do not apply the genie-aided PCA since $U_{x,r}$ may not be sparse.

In both S1 and S2 we set $p = 14$. Moreover, in S1 $n = 30$ training data are used, while $C_x$ is constructed such that 85% of the elements of $U_x$ are zero. In S2, $q = 5$, $n = 18$, 65% of $H$ is set to zero, while the nonzero ones are normally distributed. Further, $\sigma_n^2$ and $A_i$ are set such that the observation signal-to-noise ratio $SNR := 10 \log_{10}(\text{trace}(HH^T)/(p\sigma_n^2))$ is equal to 8dB. The MSEs $J_{rec}(B_m, C_m)$ and $J_{est}(B_m, C_m)$ are evaluated by averaging 200 Monte Carlo runs using a data set that is different from the training set $X$.

Fig. 1 shows $J_{rec}(\cdot)$ versus $r$ in S1. The sparsity coefficients in sparse DR and SPCA are set such that $\lambda_{i,j} = \frac{\tau_i}{|U_{x(i,j)}|}$, where $\tau_i > 0$. For fairness, the sparsity coefficients in SPC and DSPCA are set correspondingly using matrix $U_x$. In SMF $\lambda_{i,j} = \frac{\tilde{\gamma}_i}{|H_{1,ns(i,j)}|}$, where $\tilde{\gamma}_i \geq 0$, while $H_{1,ns}$ is the estimate of $H$ obtained via SMF after setting $\lambda_{i,j} = 0$. Fig. 1 indicates that Alg. 1 stays close to the genie-aided PCA and outperforms existing alternatives. Similar remarks hold for SMF in Fig. 2.

V. CONCLUSIONS
We developed two novel dimensionality-reducing schemes that fully exploit sparsity in $C_x$. These algorithms have the potential of achieving smaller reconstruction MSE than existing alternatives, at modest complexity due to the use of simple element-wise updating. Various improvements and extensions to factor analysis are currently underway.

REFERENCES