Sparse Coding Neural Gas for the Separation of Noisy Overcomplete Sources

Kai Labusch, Erhardt Barth, and Thomas Martinetz

University of Lübeck - Institute for Neuro- and Bioinformatics
Ratzeburger Alle 160 23538 Lübeck - Germany

Abstract. We consider the problem of separating noisy overcomplete sources from linear mixtures, i.e., we observe N mixtures of $M > N$ sparse sources. We show that the "Sparse Coding Neural Gas" (SCNG) algorithm [1] can be employed in order to estimate the mixing matrix. Based on the learned mixing matrix the sources are obtained by orthogonal matching pursuit. Using artificially generated data, we evaluate the influence of (i) the coherence of the mixing matrix, (ii) the noise level, and (iii) the sparseness of the sources with respect to the performance that can be achieved on the representation level. Our results show that if the coherence of the mixing matrix and the noise level are sufficiently small and the underlying sources are sufficiently sparse, the sources can be estimated from the observed mixtures.

1 Introduction

Suppose we are given a number of observations $X = (\mathbf{x}_1, \ldots, \mathbf{x}_L)$, $\mathbf{x}_j \in \mathbb{R}^N$ that are a linear mixture of a number of sparse sources $S = (\mathbf{s}_1, \ldots, \mathbf{s}_M)^T = (\mathbf{a}_1, \ldots, \mathbf{a}_L)$, $\mathbf{s}_i \in \mathbb{R}^L$ and $\mathbf{a}_j \in \mathbb{R}^M$:

$$\mathbf{x}_j = C\mathbf{a}_j + \epsilon_j \quad \|\epsilon_j\| \leq \delta \ . \tag{1}$$

Here $C = (\mathbf{c}_1, \ldots, \mathbf{c}_M), \mathbf{c}_j \in \mathbb{R}^N$ denotes the mixing matrix. We require $\|\mathbf{c}_j\| = 1$ without loss of generality. The vector $\mathbf{a}_j = (s_{1,j}, \ldots, s_{M,j})^T$ contains the contribution of the sources $\mathbf{s}_i$ to the mixture $\mathbf{x}_j$. Additionally a certain amount of additive noise ϵ_j is present. Is it possible to estimate the sources $\mathbf{s}_i$ only from the mixtures $\mathbf{x}_j$ without knowing the mixing matrix C? In the past, a number of methods have been proposed that can be used to estimate the $\mathbf{s}_i$ and C knowing only the mixtures $\mathbf{x}_j$ assuming $\|\epsilon_j\| = 0$ and $M = N$ [2]. Some methods assume that the sources are statistically independent [3]. More recently the problem of estimating an overcomplete set of sources has been studied which arises if M is larger than the number of observed mixtures N [4,5,6,7]. Fewer approaches have been proposed for source separation under the presence of noise [8]. An overview of the broad field of blind source separation and independent component analysis (ICA) can be found in [9].

In [1] we have proposed the "Sparse Coding Neural Gas" (SCNG) algorithm in order to learn sparse overcomplete data representations under the presence

of additive noise. Here we show how a slightly modified version of the same algorithm can be employed to tackle the problem of source separation in a noisy overcomplete setting. We do not make assumptions regarding the type of noise but our method requires that the underlying sources $\mathbf{s}_i$ are sufficiently sparse, in particular, it requires that the $\mathbf{a}_j$ have to be sparse and that the noise level δ as well as the number of sources M is known.

1.1   Source Separation and Orthogonal Matching Pursuit

Recently some properties of the orthogonal matching pursuit algorithm (OMP) [10] with respect to the obtained performance on the representation level have been shown [11]. These results provide the theoretical foundation that allows us to apply OMP to the problem of source separation. We here briefly discuss the most important aspects with respect to our work.

Our method does not require that the sources $\mathbf{s}_i$ are independent but it requires that only few sources contribute to each mixture $\mathbf{x}_j$, i.e, that the $\mathbf{a}_j$ are sparse. However, an important observation is that if the underlying components $\mathbf{s}_i$ are sparse and independent, for a given mixture $\mathbf{x}_j$ the vector $\mathbf{a}_j$ will be sparse too.

In order to apply the OMP algorithm to problem (1) let us assume that we know the mixing matrix C. Let us further assume that we know the noise level δ. Let $\mathbf{a}_j$ be the vector containing a small number k of non-zero entries such that

$$\mathbf{x}_j = C\mathbf{a}_j + \epsilon_j \quad \|\epsilon_j\| \leq \delta \tag{2}$$

holds for a given observation $\mathbf{x}_j$. OMP can be used to estimate $\mathbf{a}_j$ by solving

$$\|\mathbf{x}_j - C\mathbf{a}_j^{\mathrm{OMP}}\| \leq \delta \, . \tag{3}$$

It can be shown that

$$\|\mathbf{a}_j^{\mathrm{OMP}} - \mathbf{a}_j\| \leq \Lambda_{\mathrm{OMP}} \, \delta \tag{4}$$

holds if the smallest entry in $\mathbf{a}_j$ is sufficiently large and the number of non-zero entries in $\mathbf{a}_j$ is sufficiently small. Let

$$H(C) = \max_{1 \leq i,j \leq M, i \neq j} |\mathbf{c}_i^T \mathbf{c}_j| \tag{5}$$

be the mutual coherence H of the mixing matrix C. The smaller $H(C)$, N/M and k are, the smaller Λ_{OMP} becomes and the smaller $\min(\mathbf{a}_j)$ is allowed to be [11]. Since (4) only holds if the smallest entry in $\mathbf{a}_j$ is sufficiently large, OMP has the property of local stability with respect to (4) [11]. Furthermore it can be shown that under the same conditions $\mathbf{a}_j^{\mathrm{OMP}}$ contains only non-zeros that also appear in $\mathbf{a}_j$ [11]. An even globally stable approximation of $\mathbf{a}_j$ can be obtained by methods such as basis pursuit [11,12].

1.2 Optimised Orthogonal Matching Pursuit

The SCNG algorithm is based on an approximation technique that is closely related to OMP and called "Optimised Orthogonal Matching Pursuit" (OOMP) [13].

Given an observed mixture $\mathbf{x}_j$ the algorithm iteratively constructs $\mathbf{x}_j$ out of the columns of the mixing matrix C. The indices of the columns of C that have been used already are collected in the set U. Initially $U = \emptyset$. The number of elements in U, i.e, $|U|$, equals the number of iterations that have been performed so far by the OOMP algorithm. The columns of C that are indexed by U are denoted by C^U. We obtain the approximation of the noise term ϵ_j, i.e., the residual ϵ_j^U, by removing the projection of $\mathbf{x}_j$ to the subspace spanned by C^U from $\mathbf{x}_j$. The residual is initialized with $\epsilon_j^U = \mathbf{x}_j$. The algorithm runs until $\|\epsilon_j^U\| \leq \delta$. A temporary mixing matrix R is used whose columns are orthogonal to the subspace spanned by C^U. Initially $R = (\mathbf{r}_1, \ldots, \mathbf{r}_l, \ldots, \mathbf{r}_M) = C$. Note that R can be obtained by removing the projection of the columns of C to the subspace spanned C^U from C and setting the norm of the residuals $\mathbf{r}_l$ to one. In each iteration the algorithm determines the column $\mathbf{r}_l$ of R with $l \notin U$ that has maximum overlap with respect to the current residual ϵ_j^U

$$l_{\text{win}} = \arg \max_{l, l \notin U} (\mathbf{r}_l^T \epsilon_j^U)^2 \ . \tag{6}$$

Then, in the construction step, the orthogonal projection with respect to $\mathbf{r}_{l_{\text{win}}}$ is removed from the columns of R and ϵ_j^U

$$\mathbf{r}_l = \mathbf{r}_l - (\mathbf{r}_{l_{\text{win}}}^T \mathbf{r}_l)\mathbf{r}_{l_{\text{win}}} \tag{7}$$

$$\epsilon_j^U = \epsilon_j^U - (\mathbf{r}_{l_{\text{win}}}^T \epsilon_j^U)\mathbf{r}_{l_{\text{win}}} \ . \tag{8}$$

After the projection has been removed l_{win} is added to U, i.e., $U = U \cup l_{\text{win}}$. The columns $\mathbf{r}_l$ with $l \notin U$ might be selected in the subsequent iterations of the algorithm. The norm of these columns is set to unit length. If the stopping criterion $\|\epsilon_j^U\| \leq \delta$ has been reached, the final entries of $\mathbf{a}_j^{\text{OOMP}}$ can be obtained by recursively collecting the contribution of of each column of C during the construction process taking into account the normalization of the columns of R in each iteration.

Due to the normalization of the columns of R after the orthogonal projection has been performed, the selection criterion (6) ensures that the norm of the residual ϵ_j^U obtained by (8) is minimal. Hence, the OOMP algorithm can provide an approximation of $\mathbf{a}_j$ containing even less non-zeros than the approximation provided by OMP.

2 Learning the Mixing Matrix

We now consider the problem of estimating the mixing matrix $C = (\mathbf{c}_1, \ldots, \mathbf{c}_M)$ from the mixtures $\mathbf{x}_j$ provided that we know the noise level δ and the number of underlying sources M. As a consequence of the sparseness of the underlying

sources $\mathbf{s}_i$, we are looking for a mixing matrix C that minimizes the number of non-zero entries of $\mathbf{a}_j^{\mathrm{OOMP}}$, i.e., the number of iteration steps of the OOMP algorithm, given a noise level of δ

$$\min_C \frac{1}{L} \sum_{j=1}^{L} \|\mathbf{a}_j^{\mathrm{OOMP}}\|_0 \quad \text{subject to} \quad \forall j \; : \; \|\mathbf{x}_j - C\mathbf{a}_j^{\mathrm{OOMP}}\| \leq \delta \; . \tag{9}$$

Here $\|\mathbf{a}_j^{\mathrm{OOMP}}\|_0$ denotes the number of non-zero entries in $\mathbf{a}_j^{\mathrm{OOMP}}$. In order to minimize (9), we perform an update of R and C prior to the construction step (7) and (8) in each iteration of the OOMP algorithm. In order to reduce the total number of iterations, this update step minimizes the norm of the residual that is going to be obtained in the current iteration. The norm of the residual becomes small if

$$(\mathbf{r}_{l_{\mathrm{win}}}^T \epsilon_j^U)^2 \tag{10}$$

is large. Hence, we have to consider the optimization problem

$$\max_{\mathbf{r}_{l_{\mathrm{win}}}} \sum_{j=1}^{L} (\mathbf{r}_{l_{\mathrm{win}}}^T \epsilon_j^U)^2 \quad \text{subject to} \quad \|\mathbf{r}_{l_{\mathrm{win}}}\| = 1. \tag{11}$$

An optimization of (11) can be achieved by using Oja's rule [14], which is

$$\mathbf{r}_{l_{\mathrm{win}}} = \mathbf{r}_{l_{\mathrm{win}}} + \alpha \, y(\epsilon_j^U - y \, \mathbf{r}_{l_{\mathrm{win}}}) \tag{12}$$

with $y = \mathbf{r}_{l_{\mathrm{win}}}^T \epsilon_j^U$ and learning rate α. Instead of updating only the winning column of R, i.e, $\mathbf{r}_{l_{\mathrm{win}}}$, we employ the soft competitive learning approach of the "Neural Gas" (NG) algorithm [15] in order to update each column of R that might be selected in the next iteration. We determine the sequence

$$- \left(\mathbf{r}_{l_0}^T \epsilon_j^U\right)^2 \leq \ldots \leq - \left(\mathbf{r}_{l_k}^T \epsilon_j^U\right)^2 \leq \ldots \leq - \left(\mathbf{r}_{l_{M-|U|}}^T \epsilon_j^U\right)^2 \quad , \; l_k \notin U \; . \tag{13}$$

Combining Oja's rule with the soft-competitive update of the NG algorithm, we obtain

$$\Delta \mathbf{r}_{l_k} = \Delta \mathbf{c}_{l_k} = \alpha_t e^{-k/\lambda_t} y \left(\epsilon_j^U - y \, \mathbf{r}_{l_k}\right) \tag{14}$$

Here α_t and λ_t are the learning rate resp. neighbourhood size at time t:

$$\lambda_t = \lambda_0 \left(\lambda_{\mathrm{final}}/\lambda_0\right)^{t/t_{\max}} \tag{15}$$

$$\alpha_t = \alpha_0 \left(\alpha_{\mathrm{final}}/\alpha_0\right)^{t/t_{\max}} \; . \tag{16}$$

Equation (14) corresponds to the update of the NG algorithm with the distance measure

$$D(a,b) = \|a - b\| \tag{17}$$

in the input space replaced by

$$D(a,b) = \frac{\langle a,b \rangle}{\|a\|\|b\|} \tag{18}$$

for determining the rank in the sequence (13). For $t \to t_{\max}$ one obtains equation (12) as update rule. Note that (14) accumulates the updates of all iterations in the learned mixing matrix C. Due to the orthogonal projection (7) and (8) performed in each iteration, these updates are pairwise orthogonal. Furthermore, note that the columns of the original matrix emerge in random order in the learned mixing matrix and due to Oja's rule the learned columns might be multiplied by -1. The entire SCNG method is shown in Algorithm 1.

Algorithm 1. The sparse coding neural gas algorithm for source separation

> initialize $C = (\mathbf{c}_1, \ldots, \mathbf{c}_M)$ using uniform random values
> **for** $t = 0$ to $t_{\max}$ **do**
>   select random sample $\mathbf{x}$ out of X
>   set $\mathbf{c}_1, \ldots, \mathbf{c}_M$ to unit length
>   calculate current size of neighbourhood: $\lambda_t = \lambda_0 \left(\lambda_{\text{final}}/\lambda_0\right)^{t/t_{\max}}$
>   calculate current learning rate: $\alpha_t = \alpha_0 \left(\alpha_{\text{final}}/\alpha_0\right)^{t/t_{\max}}$
>   set $U = \emptyset$, $\epsilon^U = \mathbf{x}$ and $R = (\mathbf{r}_1, \ldots, \mathbf{r}_M) = C = (\mathbf{c}_1, \ldots, \mathbf{c}_M)$
>   **while** $\|\epsilon^U\| > \delta$ **do**
>   determine $l_0, \ldots, l_k, \ldots, l_{M-|U|}$ with $l_k \notin U$:
>
> $$-(\mathbf{r}_{l_0}^T \epsilon^U)^2 \leq \ldots \leq -(\mathbf{r}_{l_k}^T \epsilon^U)^2 \leq \ldots \leq -(\mathbf{r}_{l_{M-|U|}}^T \epsilon^U)^2$$
>
>   **for** $k = 1$ to $M - |U|$ **do**
>   with $y = \mathbf{r}_{l_k}^T \epsilon^U$ update $\mathbf{c}_{l_k} = \mathbf{c}_{l_k} + \Delta_{l_k}$ and $\mathbf{r}_{l_k} = \mathbf{r}_{l_k} + \Delta_{l_k}$ with
>
> $$\Delta_{l_k} = \alpha_t e^{-k/\lambda_t} y \left(\epsilon^U - y \, \mathbf{r}_{l_k}\right)$$
>
>   set $\mathbf{r}_{l_k}$ to unit length
>   **end for**
>   determine $l_{\text{win}} = \arg \max_{l \notin U} (\mathbf{r}_l^T \epsilon^U)^2$
>   remove projection to $\mathbf{r}_{l_{\text{win}}}$ from ϵ^U and R:
>
> $$\epsilon^U = \epsilon^U - (\mathbf{r}_{l_{\text{win}}}^T \epsilon^U) \mathbf{r}_{l_{\text{win}}}$$
> $$\mathbf{r}_l = \mathbf{r}_l - (\mathbf{r}_{l_{\text{win}}}^T \mathbf{r}_l) \mathbf{r}_{l_{\text{win}}} \quad, l = 1, \ldots, M$$
>
>   set $U = U \cup l_{\text{win}}$
>   **end while**
> **end for**

---

3   Experiments

In order to evaluate the performance of the SCNG algorithm with respect to the reconstruction of the underlying sources, we performed a number of experiments on artificial data. We generated sparse underlying sources $S = (\mathbf{s}_1, \ldots, \mathbf{s}_M)^T = (\mathbf{a}_1, \ldots, \mathbf{a}_L)$, $\mathbf{s}_i \in \mathbb{R}^L$, $\mathbf{a}_j \in \mathbb{R}^M$. This was done by setting up to k entries of the $\mathbf{a}_j$ to uniformly distributed random values in $[-1, 1]$. For each $\mathbf{a}_j$ the number of non-zero entries was obtained from a uniform distribution in $[0, k]$. The noise
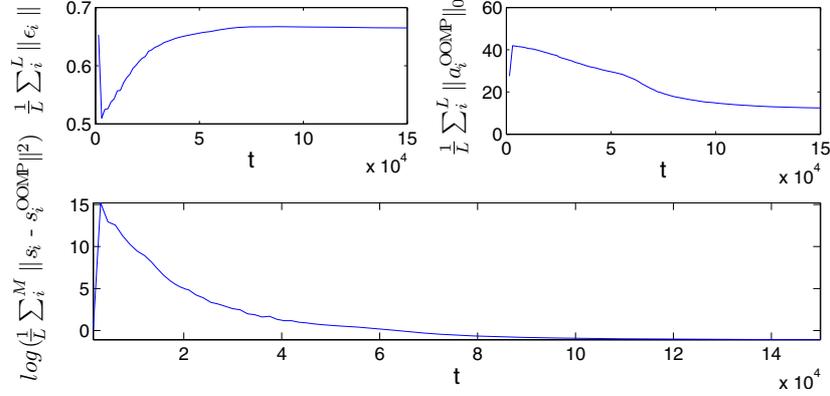
Fig. 1. The figure shows the convergence of the SCNG algorithm. Top left: The mean norm of the residual ϵ_j^U of the last iteration of the OOMP algorithm. Middle: Mean number of iterations performed by the OOMP algorithm until $\epsilon_j^U \leq \delta$. Right: Logarithmic plot of the mean squared distance between the estimated sources s_i^{OOMP} and the true sources s_i. We used $M = 100$, $N = 50$, $NR = 0.01$, $H(C) = 0.4$, $k = 15$ and $\delta = 0.7$.

was generated by adding $E = (\mathbf{n}_1, \ldots, \mathbf{n}_M)^T = (\mathbf{e}_1, \ldots, \mathbf{e}_L), \mathbf{n}_i \in \mathbb{R}^L, \mathbf{e}_j \in \mathbb{R}^M$ containing small uniformly distributed random values in $[-1, 1]$ such that

$$\mathbf{x}_i = C(\mathbf{a}_j + \mathbf{e}_j) = C\mathbf{a}_j + \epsilon_j . \tag{19}$$

The noise parameter δ was obtained as

$$\delta = \frac{1}{L} \sum_{j=1}^{L} \|\epsilon_j\| . \tag{20}$$

We scaled the S and thereby the $\mathbf{a}_j$ so that $var(CS) = 1$. The amplitude of the values in E was chosen such that

$$\frac{var(CE)}{var(CS)} = NR . \tag{21}$$

In order to obtain a random mixture matrix $C \in \mathbb{R}^{N \times M}$ with coherence z, we repeatedly chose a matrix from a uniform distribution in $[-1, 1]$ until $H(C) = z$. Then, the norm of the columns of the mixture matrix was set to unit length.

In our experiments, we study the error on the representatition level. This means that for each observation $\mathbf{x}_j$, we evaluate the difference between the original contributions of the underlying sources, i.e., $\mathbf{a}_j$ to $\mathbf{x}_j$, and the contributions $\mathbf{a}_j^{\text{OOMP}}$ that were estimated by the OOMP algorithm on the basis of the mixing matrix C that was learned by the SCNG algorithm

$$\frac{1}{L} \sum_{j=1}^{L} \|\mathbf{a}_j - \mathbf{a}_j^{\text{OOMP}}\|_2^2 = \frac{1}{L} \sum_{i=1}^{M} \|\mathbf{s}_i - \mathbf{s}_i^{\text{OOMP}}\|_2^2. \tag{22}$$

Here $S^{\mathrm{OOMP}} = (\mathbf{s}_1^{\mathrm{OOMP}}, \ldots, \mathbf{s}_M^{\mathrm{OOMP}})^T = (\mathbf{a}_1^{\mathrm{OOMP}}, \ldots, \mathbf{a}_L^{\mathrm{OOMP}})$ are the underlying sources obtained from the OOMP algorithm. In order to evaluate (22) we have to assign the entries in $\mathbf{a}_j^{\mathrm{OOMP}}$ to the entries in $\mathbf{a}_j$ which is equivalent to assigning the original sources $\mathbf{s}_i$ to the estimated sources $\mathbf{s}_i^{\mathrm{OOMP}}$. This problem arises due to the random order in which the columns of the original mixing matrix appear in the learned mixing matrix. For the assignment we perform the following procedure:

1. Set $I_{\mathrm{orig}} : \{1, \ldots, M\}$ and $I_{\mathrm{learned}} : \{1, \ldots, M\}$.
2. Find and assign $\mathbf{s}_i$ and $\mathbf{s}_j^{\mathrm{OOMP}}$ with $i \in I_{\mathrm{orig}}, j \in I_{\mathrm{learned}}$ such that

$$\frac{|\mathbf{s}_j^{\mathrm{OOMP}} \mathbf{s}_i^T|}{\|\mathbf{s}_i\| \|\mathbf{s}_j^{\mathrm{OOMP}}\|} \text{ is maximal.}$$

3. Remove i from I_{orig} and j from I_{learned}.
4. If $\mathbf{s}_j^{\mathrm{OOMP}} \mathbf{s}_i^T < 0$ set $\mathbf{s}_j^{\mathrm{OOMP}} = -\mathbf{s}_j^{\mathrm{OOMP}}$.
5. Proceed with (2) until $I_{\mathrm{orig}} = I_{\mathrm{learned}} = \emptyset$.

For all experiments we used $L = 10000$ and $\alpha_0 = 0.1$, $\alpha_{\mathrm{final}} = 0.0001$ for the learing rate as well as $\lambda_0 = M/2$ and $\lambda_{\mathrm{final}} = 0.01$ for the neighbourhood size. We repeated all experiments 10 times and report the mean result over the 10 runs. The number of learning iterations of the SCNG algorithm was set to $t_{\mathrm{max}} = 15 * 10000$.

In our first experiment, we evaluated the convergence of the SCNG algorithm over time in case of $N = 50$ observations of $M = 100$ underlying sources with up to $k = 15$ non-zero entries. The result is shown in Figure 1. The norm of the residual of the final iteration ϵ_j^U of the OOMP algorithm converges to δ. The number of iterations of the OOMP algorithm converges to the mean number of non-zero entries. At the same time also the error on the representation level is minimized. The 3 underlying sources that were estimated best as well as 3 of the mixtures from which they were obtained are shown in Figure 2.

In the next experiment, we set $N = 20$, $M = 40$ and $NR = 0.1$ and varied the coherence $H(C)$ of the mixing matrix as well as the sparseness k of the underlying components. The result is shown in Figure 3. The sparser the sources are and the smaller the coherence of the mixing matrix is, the better the obtained performance is. Then, we fixed $H(C) = 0.6$ and $k = 5$ and varied the overcompleteness by setting $M = 20, \ldots, 80$. From Figure 3 it can be seen that only slightly varying performance is obtained though the overcompleteness strongly increases. Furthermore we varied N from 10 to 50 and set $M = 2N$ as well as $k = \lceil N/10 \rceil$. Figure 3 shows that almost the same performance is obtained, i.e., the obtained performance does not depend on the number of sources and observations if the fraction N/M as well as the sparseness of the sources is constant.

The result of the next experiment is shown in Figure 4. We set $N = 20$, $M = 40$ and $H(C) = 0.6$ and varied the noise level as well as the sparseness of the sources. As expected, the more noise is present and the less sparse the sources are, the lower the obtained performance is. In a last experiment, we took the
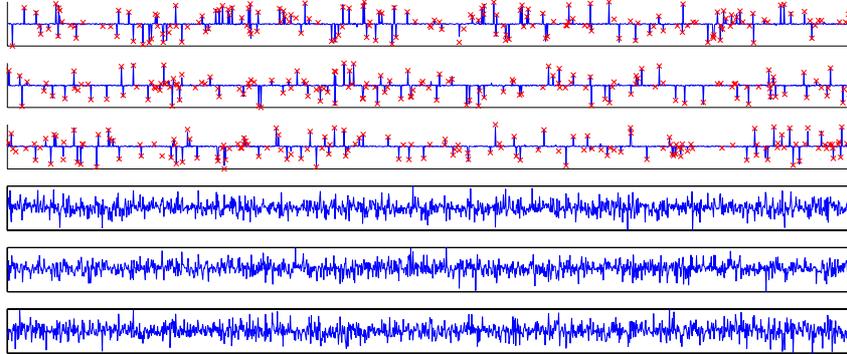
Fig. 2. The upper part of the figure shows the 3 out of $M = 100$ underlying sources that were reconstructed best from the $N = 50$ mixtures observed. 3 examples of the observations are shown in the lower part. In the upper part the solid line depicts $\mathbf{s}_i + \mathbf{n}_i$ whereas the crosses depict $\mathbf{s}_i^{\mathrm{OOMP}}$ that were obtained by applying the OOMP to the mixtures using the mixing matrix that was learned by the SCNG algorithm. The coherence of the mixture matrix was set to $H(C) = 0.4$. The data was generated using $k = 15$, $NR = 0.01$. We used $\delta = 0.7$.
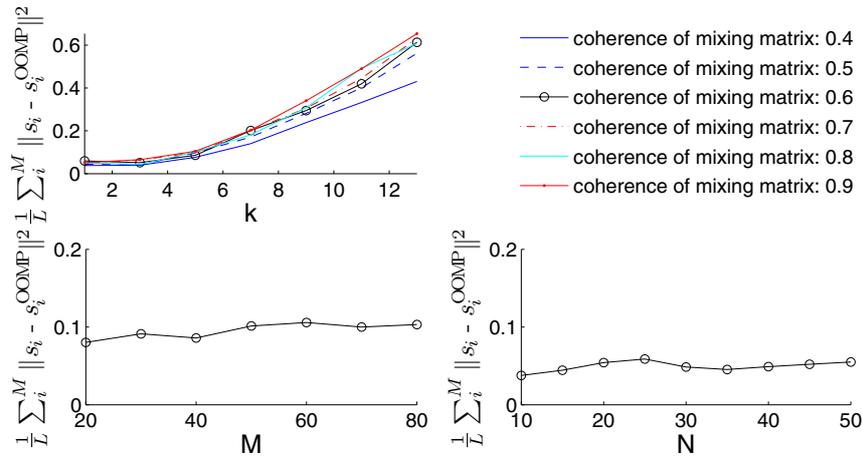
Fig. 3. Top left: The influence of decreasing sparseness and increasing coherence of the mixing matrix with respect to the reconstruction error is shown. We used $N = 20$, $M = 40$ and $NR = 0.1$. Bottom left: The obtained reconstruction error using $M = 20, \ldots, 80$ and $k = 5$. Bottom right: The obtained reconstruction error for $N = 10, \ldots, 50$ with $M = 2N$ and $k = \lceil N/10 \rceil$.

same parameters, set $k = 5$ and studied the obtained reconstruction performance depending on the coherence of the mixing matrix and the noise level. The result is also shown in Figure 4. It can be seen that in our experimental setting the
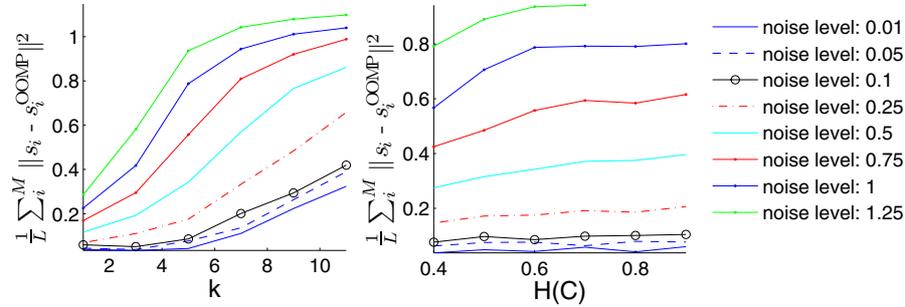
Fig. 4. Left: We used $N = 20, M = 40$. The coherence of the mixing matrix was set to 0.6. The reconstruction performance depending on the noise level as well as on the sparseness is shown. Right: The sparseness parameter k was set to 5. The obtained reconstruction error depending on the noise level and the coherence of the mixing matrix is shown.

noise level has an strong impact on the performance. The influence of the noise cannot be compensated by the coherence of the mixing matrix.

4   Conclusion

We introduced the SCNG algorithm in order to tackle the difficult problem of estimating the underlying sources of a linear mixture in a noisy overcomplete setting. Our model does not make assumptions regarding the distribution of the sources or the distribution of the noise. However, the method requires that the sources are sparse and that the noise level as well as the number of the underlying sources are known or can be estimated.

Based on the mixing matrix that was learned by the SCNG algorithm, we evaluated the performance on the representation level by employing the OOMP algorithm in order to obtain the sources from the observations. We analyzed the performance with respect to the reconstruction of the original sources that can be achieved. We studied the influence of the coherence of the mixing matrix, the noise level and the sparseness of the underlying sources. If the sources are sufficiently sparse and the coherence of the mixing matrix and the noise level are sufficiently small, the SCNG algorithm is able to learn the mixing matrix and the sources can be reconstructed. We also evaluated the influence of the overcompleteness with respect to the obtained performance. Our results show that sufficiently sparse sources can be reconstructed even in highly over-complete settings. In order to improve the performance on the representation level computationally more demanding methods such as basis pursuit might be used.

References

1. Labusch, K., Barth, E., Martinetz, T.: Learning Data Representations with Sparse Coding Neural Gas. In: Verleysen, M. (ed.) Proceedings of the 16th European Symposium on Artificial Neural Networks, pp. 233–238. D-Side Publishers (2008)
2. Bell, A.J., Sejnowski, T.J.: An information-maximization approach to blind separation and blind deconvolution. Neural Computation 7(6), 1129–1159 (1995)
3. Hyvärinen, A.: Fast and robust fixed-point algorithms for independent component analysis. IEEE Transactions on Neural Networks 10(3), 626–634 (1999)
4. Hyvarinen, A., Cristescu, R., Oja, E.: A fast algorithm for estimating overcomplete ica bases for image windows. In: Proceedings of the International Joint Conference on Neural Networks, IJCNN 1999, vol. 2, pp. 894–899 (1999)
5. Lee, T.W., Lewicki, M., Girolami, M., Sejnowski, T.: Blind source separation of more sources than mixtures using overcomplete representations. IEEE Signal Processing Letters 6(4), 87–90 (1999)
6. Lewicki, M.S., Sejnowski, T.J.: Learning Overcomplete Representations. Neural Computation 12(2), 337–365 (2000)
7. Theis, F., Lang, E., Puntonet, C.: A geometric algorithm for overcomplete linear ICA. Neurocomputing 56, 381–398 (2004)
8. Hyvärinen, A.: Gaussian moments for noisy independent component analysis. IEEE Signal Processing Letters 6(6), 145–147 (1999)
9. Hyvarinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. Wiley-Interscience, Chichester (2001)
10. Pati, Y., Rezaiifar, R., Krishnaprasad, P.: Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In: Proceedings of the 27 th Annual Asilomar Conference on Signals, Systems (November 1993)
11. Donoho, D.L., Elad, M., Temlyakov, V.N.: Stable recovery of sparse overcomplete representations in the presence of noise. IEEE Transactions on Information Theory 52(1), 6–18 (2006)
12. Chen, S.S., Donoho, D.L., Saunders, M.A.: Atomic decomposition by basis pursuit. SIAM Journal on Scientific Computing 20(1), 33–61 (1998)
13. Rebollo-Neira, L., Lowe, D.: Optimized orthogonal matching pursuit approach. IEEE Signal Processing Letters 9(4), 137–140 (2002)
14. Oja, E.: A simplified neuron model as a principal component analyzer. J. Math. Biol. 15, 267–273 (1982)
15. Martinetz, T., Berkovich, S., Schulten, K.: "Neural-gas" Network for Vector Quantization and its Application to Time-Series Prediction. IEEE-Transactions on Neural Networks 4(4), 558–569 (1993)