MODIFIED HIERARCHICAL CLUSTERING FOR SPARSE COMPONENT ANALYSIS

N. Mourad and James P. Reilly

Department of Electrical & Computer Eng.
McMaster University, 1280 Main St. W., Hamilton, Ontario, Canada L8S 4K1
email: mouradna@mcmaster.ca, reillyj@mcmaster.ca

ABSTRACT

The under-determined blind source separation (BSS) problem is usually solved using the sparse component analysis (SCA) technique. In SCA, the BSS is usually solved in two steps, where the mixing matrix is estimated in the first step, while the sources are estimated in the second step. In this paper we propose a novel clustering algorithm for estimating the mixing matrix and the number of sources, which is usually unknown. The proposed algorithm is based on incorporating a statistical test with a hierarchical clustering (HC) algorithm. The proposed algorithm is based on sequentially extracting compact clusters that have been constructed by the HC algorithm, where the extraction decision is based on the statistical test. To identify the number of sources, as well as the clusters corresponding to the columns of the mixing matrix, we develop a quantitative measure called the *concentration parameters*. Two numerical examples are presented to present the ability of the proposed algorithm in estimating the mixing matrix and the number of sources. .

Indexing Terms: *blind source separation, sparse component analysis, clustering.* .

I. INTRODUCTION

The blind source separation (BSS) problem is defined as the problem of reconstructing n *unknown* source signals from m linear measurements when the mixing matrix is *unknown*. The relation between the measured signals and the original source signals can be expressed mathematically as

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{V}, \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{m\times T}$ is a matrix of measured signals, $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ is an *unknown* mixing matrix, $\boldsymbol{S} \in \mathbb{R}^{n\times T}$ is a matrix of it unknown source signals, $\boldsymbol{V} \in \mathbb{R}^{m\times T}$ is a matrix of additive random noise, m is the number of observations, n is the number of sources, and T is the number of samples.

In this paper we consider the under-determined BSS problem in which the number of sources is greater than the number of sensors. When the source matrix is sparse, the under-determined BSS problem is usually solved using the recently developed technique known as Sparse Component Analysis (SCA). Since non–sparse sources can often be sparsely represented under a suitable linear transformation, (e.g., the short time Fourier transform and the wavelet related transforms), the SCA problem is quite general and also applicable to non-sparse sources. In this paper we assume, without loss of generality, that the hidden sources are sparse.

Usually, the under–determined BSS problem is solved via SCA by following the two-step approach [1], [2], [3], in which the mixing matrix and the sparse sources are estimated separately. In the first step, the mixing matrix is estimated via *clustering* the columns of the measured matrix $\boldsymbol{X}$, while the sparse source matrix $\boldsymbol{S}$ is estimated in the second step using e.g., [4].

Although the two-step approach produced promising results, precise estimation of the mixing matrix remains a problem in this approach. Most of the clustering algorithms that have been utilized for estimating the mixing matrix have some limitations. For example, most SCA algorithms are based on estimating the mixing matrix via *partitioning clustering* algorithms, e.g. *k*–means [1], fuzzy c–means [3], or modified *k*–means [2]. However, and generally speaking, there are three main problems associated with most partitioning clustering algorithms [**?**]:

- The number of clusters, which equals the number of sources, has to be known in advance; a condition which might not be available in some applications.
- Since each point (column of $\boldsymbol{X}$) must be assigned into a cluster, most partitioning clustering algorithms fail in the presence of noise and/or outlier points.
- All partitioning clustering algorithms are locally convergent and sensitive to the initial choice of the clusters' centroids.

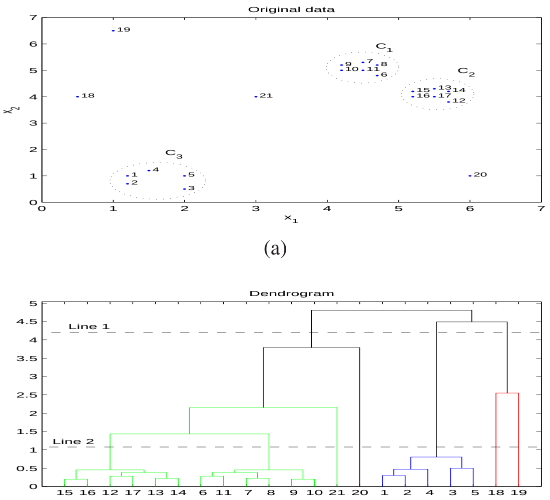
In this paper we propose a novel clustering algorithm that alleviates the impact of the three afore mentioned limitations. Moreover, the proposed algorithm can estimate the number of clusters directly from the data matrix. Accordingly, the proposed algorithm can handle the case where the number of clusters (sources) is unknown.

II. HIERARCHICAL CLUSTERING

HC follows a clustering strategy which quite distinct from that of the partitioning clustering algorithms. HC algorithms start with every object (columns of the data matrix) assigned into a separate cluster. Therefore, in contrast to the partitioning clustering algorithms, hierarchical clustering algorithms does not suffer from the initialization problem. Then, in each successive step, the two closest[1] clusters are merged into a single cluster. The process continues until all objects are assigned into a single cluster.

This work was supported in part by the Egyptian Ministry of High Education and the Natural Sciences and Engineering Research Council of Canada (NSERC).

[1] A definition of the closeness between two clusters is provided later.



(a)

(b)

Fig. 1. (a) A data set consists of 3 clusters and 4 outlier points; (b) Clustering the data set presented in (a) using hierarchical clustering.

It is convenient to depict the result of an HC algorithm using a dendrogram plot. The dendrogram plot resulting from applying a HC algorithm on the simple data set shown in Figure 1(a) is shown in Figure 1(b). In this plot, the root node of the dendrogram represents the whole data set and each leaf node is regarded as a data object. An internal node represents the union of all objects in its subtree, and the height of an internal node represents the distance between its two child nodes [5]. Note that the labels of the root nodes are exactly those of the clustered data points in Figure 1(a).

A useful property associated with the HC algorithms [**?**], which is also depicted in Figure 1(b), is that closer objects are combined in early stages while distant objects (outliers) are combined in late stages. However, the main difficulty associated with HC is identifying the individual clusters. As shown in Figure 1(b), the clusters are not explicit and have to be determined in some way from the dendrogram.

There are several methods for identifying from the dendogram, e.g. [**?**], but they are difficult to use or cannot be automated. In this paper we propose a novel clustering algorithm in which a statistical test is utilized for identifying the individual clusters. The utilized statistical test is summarized in the next section.

III. T^2–STATISTICAL TEST

Consider the following two clusters $R = [\boldsymbol{r}_1 \boldsymbol{r}_2 \dots \boldsymbol{r}_{T_1}]$ and $Q = [\boldsymbol{q}_1 \boldsymbol{q}_2 \dots \boldsymbol{q}_{T_2}]$, where both $\boldsymbol{r}_i$ and $\boldsymbol{q}_j \in \mathbb{R}^{m\times 1}$. It is assumed that: 1) The sample set $\boldsymbol{r}_1 \boldsymbol{r}_2 \dots \boldsymbol{r}_{T_1}$ is a random sample of size T_1 from an m–variate population with mean vector $\boldsymbol{\mu}_1$ and covariance matrix $\boldsymbol{\Sigma}_1$, 2) The sample set $\boldsymbol{q}_1 \boldsymbol{q}_2 \dots \boldsymbol{q}_{T_2}$ is a random sample of size T_2 from an m–variate population with mean vector $\boldsymbol{\mu}_2$ and covariance matrix $\boldsymbol{\Sigma}_2$, 3) Also, $\boldsymbol{r}_1 \boldsymbol{r}_2 \dots \boldsymbol{r}_{T_1}$ are independent of $\boldsymbol{q}_1 \boldsymbol{q}_2 \dots \boldsymbol{q}_{T_2}$. Our goal is to test whether the null hypothesis $H_0 : \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2 = 0$ is true or not. The validity of H_0 can be checked using the T^2–statistic test [6], which is summarized as follows:

1) For each cluster calculate the maximum likelihood estimate of both the mean vector and covariance matrix as follows

$$\bar{\boldsymbol{r}} = \frac{1}{T_1}\sum_{j=1}^{T_1} \boldsymbol{r}_j \qquad \boldsymbol{S}_r = \frac{1}{T_1 - 1}\sum_{j=1}^{T_1} (\boldsymbol{r}_j - \bar{\boldsymbol{r}})(\boldsymbol{r}_j - \bar{\boldsymbol{r}})^T$$

$$\bar{\boldsymbol{q}} = \frac{1}{T_2}\sum_{j=1}^{T_2} \boldsymbol{q}_j \qquad \boldsymbol{S}_q = \frac{1}{T_2 - 1}\sum_{j=1}^{T_2} (\boldsymbol{q}_j - \bar{\boldsymbol{q}})(\boldsymbol{q}_j - \bar{\boldsymbol{q}})^T$$

2) Calculate the pooled covariance matrix

$$\boldsymbol{S}_{po} = \frac{(T_1 - 1)\boldsymbol{S}_r + (T_2 - 1)\boldsymbol{S}_q}{T_1 + T_2 - 2}$$

3) Calculate the quantity

$$T^2 = \frac{T_1 T_2}{T_1 + T_2}(\bar{\boldsymbol{r}} - \bar{\boldsymbol{q}})^T \boldsymbol{S}_{po}^{-1} (\bar{\boldsymbol{r}} - \bar{\boldsymbol{q}})$$

which is distributed as $\frac{(T_1+T_2-2)m}{(T_1+T_2-m-1)} F_{m,T_1+T_2-m-1}$, where $F_{a,b}$ is an F-distribution with a and b degrees of freedom $(d.f.)$

4) Select a value for α, e.g. $\alpha = 0.05$, then reject H_0 if $T^2 > \frac{(T_1+T_2-2)m}{(T_1+T_2-m-1)} F_{m,T_1+T_2-m-1}(\alpha)$, where $F_{m,T_1+T_2-m-1}(\alpha)$ is the upper $(100\alpha) - th$ percentile of an F-distribution with m and $(T_1 + T_2 - m - 1)$ $d.f.$

IV. PROPOSED CLUSTERING ALGORITHM

In this section we propose a novel clustering algorithm in which the statistical test presented in Section III is utilized for identifying the individual clusters constructed using a HC algorithm. The statistical test is applied at each step of the hierarchical process to test whether the two closest clusters are significantly different or not. The two closest clusters are merged into a single cluster if they are not significantly different, and the process continues. Otherwise, the largest one of them is extracted and removed from the data matrix. The process is repeated until all clusters are extracted.

In the following, we use the average–linkage distance to measure the distance between two clusters. The average–distance between any two clusters R and Q is defined as [5]

$$D(R,Q) = \frac{1}{|R||Q|}\sum_{i\in R, j\in Q} d_{ij} \qquad (2)$$

where $|R|$ and $|Q|$ denote the number of objects in clusters R and Q, respectively.

To identify the clusters corresponding to the columns of the mixing matrix, we make use of the assumption that the *disjoint orthogonality* property [2], i.e., $\hat{s}_l[i] \cdot \hat{s}_l[j] = 0, \forall i \neq j$ (or $\hat{s}_l[i] \cdot \hat{s}_l[j] \approx 0$ in the noisy case), is satisfied for a sufficiently large number of columns of the source coefficient matrix $\boldsymbol{S}$. Accordingly, for the columns of the source coefficient matrix $\boldsymbol{S}$ that contain a single nonzero entry, the corresponding columns of $\boldsymbol{X}$ constitute hyperlines in the m dimensional space, where the orientations of the hyperlines correspond to the columns of the mixing matrix $\boldsymbol{A}$ [2]. For a clustering algorithm

Table I. Sequential Cluster Extraction Algorithm

```
Algorithm 1: [$\mathcal{I}_l$, $\boldsymbol{Z}_{l+1}$] = SCA($\boldsymbol{Z}_l$, $\alpha$)

1) Start with $T^l$ singleton clusters, where $T^l$ is the number of columns of $\boldsymbol{Z}_l$.
2) Calculate the dissimilarity matrix $\boldsymbol{D} \in \mathbb{R}^{T^l \times T^l}$, whose $(i,j)$th coefficient is given by $d_{ij} = ||\boldsymbol{z}_i^l - \boldsymbol{z}_j^l||_2, \quad i,j = 1,2,\dots T^l$, where $\boldsymbol{z}_i^l$ is the $i$th column of $\boldsymbol{Z}_l$.
3) Search the minimal distance between clusters
       $D(C_u, C_v) = \min_{1 \leq i < j \leq T_k} D(C_i, C_j),$
   where $D(*,*)$ is a distance function between two clusters, defined in (2), and $T_k$ is the number of clusters at the $k-th$ iteration.
4) For the given value of $\alpha$, apply the $T^2$–statistical test, presented in Section III, to test whether the centroids (means) of the two clusters $C_u$ and $C_v$ are significantly different or not.
5)     if the two means are not significantly different, merge clusters $C_u$ and $C_v$ in a single cluster, and update the dissimilarity matrix. Go to step 3.
       else
           if $|C_v| \leq |C_u|$, set $\mathcal{I}_l = [i_{u1}, \dots, i_{uT_u}]$;
           else set $\mathcal{I}_l = [i_{v1}, \dots, i_{vT_v}]$;
           break
           end
       end
6) $\boldsymbol{Z}_{l+1} = \boldsymbol{Z}_l / \boldsymbol{Z}_l(: \ , \mathcal{I}_l)$.
```

to perform properly, it is necessary to transform each hyperline into a cluster. This may be accomplished by creating a normalized matrix $\boldsymbol{Z}$ formed by projecting the columns of $\boldsymbol{X}$ onto the unit hypersphere. Columns of $\boldsymbol{X}$ with norms smaller than a pre-specified value are excluded from $\boldsymbol{Z}$, since they likely correspond to noise. Each column of $\boldsymbol{Z}$ is multiplied by the sign of its first entry. After constructing the data matrix $\boldsymbol{Z}$, the clusters are sequentially extracted using the proposed algorithm, which is summarized in Table I.

The algorithm has two inputs and two outputs. The first input is the data matrix $\boldsymbol{Z}_l$ which equals the data matrix $\boldsymbol{Z}$ after removing the columns corresponding to the previously extracted $(l-1)$ clusters. Accordingly, $\boldsymbol{Z}_1 = \boldsymbol{Z}$. The second input is the value of α required for applying the statistical test. On the other hand, the first output $\mathcal{I}_l$ contains the indices of the lth extracted cluster, while the second output is the input data matrix after removing the columns corresponding to the extracted cluster. Since the proposed algorithm extracts only one cluster, it can be repeated many times until the number of the remaining columns in the data matrix is less than or equal to 1.

IV-A. Estimating the mixing matrix and the number of sources

After execution of the HC algorithm, we are left with the problem of associating the clusters with the columns of $\boldsymbol{A}$. To this end, we extract a *concentration parameter* (CP) from the data matrix $\boldsymbol{X}$. The proposed CP parameter measures how well the points of each cluster are concentrated around a hyperline in the m dimensional space.

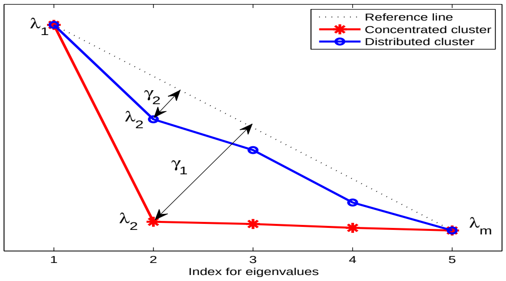


Fig. 2. The distance from λ_2 to the virtual line connecting λ_1 and λ_m can be used as a concentration parameter.

The proposed CP uses the values of the eigenvalues of the correlation matrix of a given cluster to investigate whether this cluster is a concentrated cluster or not. To be specific, let $\boldsymbol{C}^l = \boldsymbol{X}_l(:, \mathcal{I}_l)$ denote the matrix constructed from the columns of the data matrix $\boldsymbol{X}_l$ with indices corresponding to the lth extracted cluster, and let $\boldsymbol{R}^l = \boldsymbol{C}^l {\boldsymbol{C}^l}^T \in \mathbb{R}^{m \times m}$, denote the correlation matrix of this cluster. The eigenvalue decomposition of $\boldsymbol{R}^l$ can be expressed as

$$\boldsymbol{R}^l = \boldsymbol{U} \boldsymbol{\Lambda} \boldsymbol{U}^T,$$

where the columns of $\boldsymbol{U}$ are the eigenvectors and $\boldsymbol{\Lambda}$ is a diagonal matrix containing the sorted eigenvalues.

We define the concentration parameter as the distance from a point representing λ_2 to a *virtual* line connecting λ_1 and λ_m, as shown in Figure 2. The virtual line connecting λ_1 and λ_m is shown by the dotted line in Figure 2, and the distance is depicted by the double arrow. In Figure 2, the eigenvalue distributions corresponding to a concentrated and distributed cluster are depicted by the red and blue curves, respectively. As shown in this figure, $\gamma_1 \gg \gamma_2$, where γ_1 and γ_2 are the respective distances for the concentrated and distributed clusters. Accordingly, the value of the proposed CP can be used to differentiate between concentrated and distributed clusters.

The distance from the point $(2, \lambda_2)$ to the line connecting the two points $(1, \lambda_1)$ and (m, λ_m) is given by

$$\gamma = \frac{(m-1)(\lambda_1 - \lambda_2) - (\lambda_1 - \lambda_m)}{\sqrt{(m-1)^2 + (\lambda_1 - \lambda_m)^2}}. \qquad (3)$$

The value of γ can be used for estimating the number of sources and to identify the clusters corresponding to the columns of $\boldsymbol{A}$. The values $\{\gamma_l, l = 1, \dots, L\}$, arranged into descending order, where γ_l is the value of γ for the lth cluster, and L is the total number of clusters, are plotted vs. the index l. If the disjoint orthogonality property is satisfied for a reasonably large number of columns of the source matrix, the plot will have large values for the clusters corresponding to the columns of the mixing matrix. The value of γ is expected to drop off quickly for those clusters corresponding to outliers or to points which are linear combinations of columns of $\boldsymbol{A}$.

V. SIMULATION RESULTS

In this section, and due to space limitation, we present only two examples. In the two examples, an $(m \times n)$

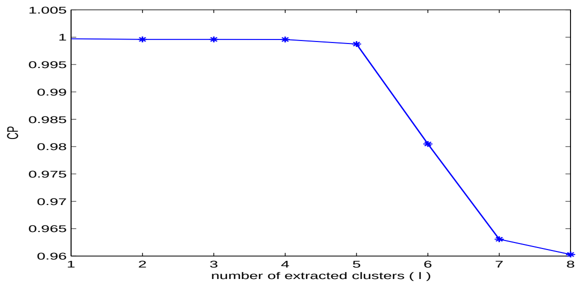


Fig. 3. The sorted CP shows a transition gap at $l = n$.

mixing matrix is randomly generated from a zero mean $i.i.d.$ normal distribution with zero mean and unit variance, while the sparse source matrix $\boldsymbol{S}$ is constructed such that $\delta\%$ of its columns satisfy the disjoint orthogonality principal, i.e., each source signal is uniquely represented by $\beta = (\delta T)/(100n)$ columns. The indices of the nonzero entries of each row of $\boldsymbol{S}$ are randomly selected, and their amplitudes are chosen from a uniform distribution between ± 1. The value of α used for the statistical test is selected equal to 0.02. The signal to interference ration (SIR) [2] between the true mixing matrix and the estimated mixing matrix is used as a measure of performance.

Example 1: In this example we demonstrate the ability of the proposed algorithm to estimate the mixing matrix and the number of sources. The parameters m, n, T, and δ are selected to be equal to 3, 5, 800, and 30, respectively. The proposed algorithm extracted 8 clusters from the constructed data matrix. The plot of the sorted CP parameter is shown in Figure 3. As shown in this figure, the sorted CP has a transition gap at the value of l corresponding to the number of sources. Moreover, the SIR between the true mixing matrix and the centroids of the 5 clusters corresponding to the 5 largest values in CP is 56.6713 dB, while the SIR between the true mixing matrix and the mixing matrix estimated using k-means is 14.8588 dB. Clearly, the mixing matrix, as well as the number of sources, are estimated correctly using the proposed algorithm.

Example 2: In this example we compare the performance of the proposed HC algorithm with the case where the k-means algorithm is used for clustering, for the case of noisy measurements. In this example, we assume that the number of sources is known a priori. This is because we are comparing the proposed algorithm with the k-means algorithm, which requires the number of clusters to be known. Accordingly, the CP parameter in this example is used only for associating the clusters and not for estimating the number of clusters. The values of m, n, and T are selected equal to 5, 7, and 800, respectively. The measurements are constructed as follows. The sparse source matrix is constructed such that $\delta\%$ of its columns satisfy the disjoint orthogonality principal. The experiment is repeated for $\delta = 30$ and $\delta = 50$. For each value of δ, a zero mean white Gaussian noise matrix $\boldsymbol{V} \in \mathbb{R}^{5\times 800}$ is randomly generated. The amplitude of the noise is adjusted to produce one of the following values of the SNR; 20, 25, 35, and 45 dB. For a given value of δ and SNR, a noisy measurement matrix $\boldsymbol{X} = \boldsymbol{AS} + \boldsymbol{V}$ is constructed, and the mixing matrix $\boldsymbol{A}$ is estimated using the k-means algorithm and the proposed algorithm. After estimating the mixing matrix, the SIR parameter is calculated. For each value of δ and the SNR, the experiment is repeated 100 different times and the average SIR is calculated and listed in Table II. It is clear that the proposed algorithm outperform the k-means algorithm, especially for SNR ≥ 35 dB.

Table II. Results of Example 2.

$\delta = 30$				
SNR (dB)	20	25	35	45
Proposed	4.5	9.17	30.1	40
k-means	6.02	7.05	6.54	6.24
$\delta = 50$				
Proposed	6.23	12.6	36.4	48.3
k-means	8.29	9.05	7.68	7.69

VI. CONCLUSION

In this paper we proposed a novel clustering algorithm that can estimates the mixing matrix, as well as the number of sources, for solving the under-determined BSS problem via the SCA technique. The proposed algorithm is based on incorporating a statistical test with a hierarchical clustering algorithm. The proposed algorithm is based on sequentially extracting compact clusters that have been constructed by the HC algorithm, where the extraction decision is based on the statistical test. To identify the number of sources, as well as the clusters corresponding to the columns of the mixing matrix, we developed a quantitative measure called the *concentration parameters*. Two numerical examples were presented to present the ability of the proposed algorithm in estimating the mixing matrix and the number of sources.

VII. REFERENCES

[1] Y. Li, A. Cichocki, and S. Amari, "Sparse Component Analysis for Blind Source Separation With Less Sensors Than Sources," in Proc. 4th Int. Symp. Independent Component Analysis Blind Source Separation (ICA BSS), 2003, pp. 89-94.

[2] Z. He, A. Cichocki, Y. Li, S. Xie, and S. Sanei, "K-hyperline clustering learning for sparse component analysis," Signal Processing, vol. 89, no. 6, pp. 1011-1022, 2009.

[3] A. M. Bronstein, M. M. Bronstein, M. Zibulevsky, and Y. Y. Zeevi, "Sparse ICA for blind separation of transmitted and reflected images," International Journal of Imaging Science and Technology (IJIST), vol. 15/1, pp. 84-91, 2005.

[4] N. Mourad and J. Reilly, "ℓ_p Minimization for Sparse Vector Reconstruction", to appear, *ICASSP2009*, Taipei, Taiwan, Apr. 19-24, 2009.

[5] L. Kaufman and P. J. Rousseeuw, "Finding Groups in Data : an Introduction to Cluster Analysis," New York, Wiley, c 1990

[6] A. C. Rencher, "Multivariate Statistical Inference and Applications," New York, Wiley, c 1998.