ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

Analysis of the Weighted Fuzzy C-means in the Problem of Source Location

Everton Z. Nadalin, Rodrigo C. Silva, Romis Attux and João M. T. Romano [1]

School of Electrical and Computer Engineering - DSPCom Laboratory
University of Campinas (UNICAMP), Av. Albert Einstein, 400, 13083-970 – Campinas, SP, Brazil

Abstract. This paper proposes the use of the clustering method called *Weighted Fuzzy C-means* to solve the problem of mixing matrix estimation in underdetermined source separation based on sparse component analysis. The performed comparative analysis shows that the approach has a significant application potential, especially if the distributions of the columns of the mixing matrix has a non-uniform character.

1 Introduction

The problem of *Blind Source Separation* (BSS) consists in recovering a set of signals (called *sources*) from mixed versions of them, using a minimum amount of *a priori* information [1][2]. There are different hypotheses about the sources that can be used to build a BSS methodology in distinct practical contexts. A possibility that has received a great deal of attention in the last years is to assume that sources are *sparse* in some domain. A signal is considered sparse if most of its energy is concentrated on a small parcel of its coefficients [2]. Unlike what occurs in the context of traditional (e.g. ICA-based) source separation, when the signals are sparse, it is possible, at least in theory, to recover the sources even in situations with more sources than mixtures (the called *underdetermined* case) [2]. The use of sparseness is relevant in many applications, like audio and speech signal processing [2], and techniques based on this property are grouped under the aegis of the concept of *Sparse Component Analysis* (SCA). In this work, it is proposed the use of a powerful clustering method, called *Weighted Fuzzy C-means* [3], in the problem of mixing matrix estimation – one important step for BSS in the underdetermined case.

2 Source Separation and Sparse Component Analysis

The problem of BSS can be described as follows: given a set of measurements $\mathbf{x}(n)$, which are mixtures of a set of source signals $\mathbf{s}(n)$, the aim is to recover the sources in an unsupervised fashion. If the mixing process is instantaneous, linear and time-invariant it is possible to represent it with the aid of an $M \times P$ matrix $\mathbf{A}$, where P is the number of sources and M is the number of mixtures. If $P=M$ and $\mathbf{A}$ has full rank, the problem of BSS is equivalent to that of finding a separating matrix $\mathbf{W}$ that inverts

[1] The authors would like to thank FAPESP (process 2012/08212-4) and CNPq.

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7. Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

the mixing process. When the sources are mutually independent, this problem has been classically solved using *Independent Component Analysis* (ICA) [1][2].

If the number of sources is higher than the number of mixtures – i.e. in an underdetermined case – source recovery using only independence is not possible. In these situations, other types of assumption need to be considered. Sparseness has been a most useful choice [2].

2.1 Sparse Component Analysis

Sparseness implies, in essence, that a small proportion of signal samples, in some domain, convey a large proportion of the underlying information. The *Cocktail Party Problem* [2] provides an example: it is quite plausible that not all guests will talk simultaneously at all moments.

Interestingly, [4] showed that, if there is no overlap among the non-null source samples, it is possible to perfectly recover the sources, even when the mixing process is underdetermined. This work also showed that audio sources are good instances of sparse sources, being the time-frequency domain the most suitable one to highlight this property. Techniques that perform BSS based on sparseness belong to the class of *Sparse Component Analysis* (SCA) approaches [2]. Typically, SCA tools divide the BSS problem into three steps: (i) estimating the number of sources, (ii) estimating the mixing system and (iii) separating the sources. This paper will focus only in the second step, the estimation of the mixing matrix **A**.

2.1.1 Estimation of the Mixing System

Some of the first SCA approaches transformed the problem of estimating the elements of **A** into a clustering problem [2]. This strategy is interesting, as it allows the first and the second step to be directly dealt with: the number of clusters is indicative of number of sources and the centroid of each cluster brings information about a mixing direction [2].

A basic premise in [4] is that there is no overlap among the source samples in the time-frequency domain. The *Short Time Fourier Transform* (STFT) is being used in this case. For the sake of clarity of presentation, the estimation approach will be presented for the case of two mixtures. If $X_i(t,f)$ is the time-frequency representation of the i-th mixture of $\mathbf{x}(n)$, it is possible to define $\alpha(t, f)$:

$$\alpha(t,f) = \frac{X_1(t,f)}{X_2(t,f)} = \frac{a_{11}S_1(t,f) + a_{12}S_2(t,f) + \cdots + a_{1N}S_N(t,f)}{a_{21}S_1(t,f) + a_{22}S_2(t,f) + \cdots + a_{2N}S_N(t,f)}. \tag{1}$$

Considering that in one sample or small window of data, represented by (t_c, f_c), only one source $S_j(t_c, f_c)$ is 'active' and all the other sources are 'null', $S_i(t_c, f_c) = 0 \ \forall\ i \neq j$, it is possible to rewrite (1) as:

$$\alpha_j(t_c, f_c) = \frac{a_{1j}S_j(t_c, f_c)}{a_{2j}S_j(t_c, f_c)} = \frac{a_{1j}}{a_{2j}}, \tag{2}$$

i.e., this parameter expresses the ratio between the coefficients of one of the columns of the mixing matrix. Consequently, the arctangent of this ratio will yield one angle θ_j, which is the direction associated with that specific column. Since it is not possible to

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

say, in each instant c, which of the sources is 'active', let us generate a set of θ_c values, defined as:

$$\theta_c = \tan^{-1}\left(\frac{X_1(t_c, f_c)}{X_2(t_c, f_c)}\right). \quad (3)$$

Note that each θ_c is a candidate to be the direction of one of the columns of **A**. If (3) is applied to the whole set of c, there is a set of $\boldsymbol{\theta}$ in which it is possible to extract important information about the mixture, such as the number of sources and the mixing system. A natural option to do so is to cluster the data.

3 Clustering Methods

In this work, three different clustering algorithms will be compared: *k-means*, *fuzzy c-means* and *weighted fuzzy c-means*. Although the first two approaches have been used in works found in the literature [5], to the best of our knowledge, this is not the case for the *weighted fuzzy c-means*. Hence, in addition to the comparative analysis, the use of this method can be considered an original contribution.

3.1 K-means

The clustering method called *k-means* can be understood as a centroid search method that minimizes the following cost function [6]:

$$J(C,\alpha) = \sum_{j=1}^{N}\sum_{k=1}^{K} u_{kj} \left\|\theta_j - C_k\right\|^{\alpha}, \quad (4)$$

where N is the number of samples, K is the total number of clusters and C_k is the center of the k-th cluster; $u_{kj} = 1$ if $\theta_j \in C_k$, in other cases, $u_{kj} = 0$. A pseudo-code of the *k-means* can be found in [6].

3.2 Fuzzy C-means

The *Fuzzy C-means* (FCM) algorithm aims at clustering data through an iterative minimization process of the following cost function [7]:

$$J(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{N} u_{ik}^{m} \left\|x_k - v_i\right\|^{2}, \quad (5)$$

where $U = [u_{ik}]$ is the membership matrix and u_{ik}, $1 \le i \le c$ and $1 \le k \le N$, which denotes the degree of membership of the sample x_k with respect to the i-th cluster; $V = \{v_i\}$ is the set of cluster prototypes and v_i is the prototype of the i-th cluster; m (> 1.0) is a coefficient that defines the membership profile in individual clusters. An FCM pseudo-code is described as follows

```
1 Initialize the membership matrix;
2 t = 0;
3 Do:
  3.1 For i = 1 until c do:
    3.1.1 Calculate the prototypes:
```

$$v_i(t) = \frac{\sum_{k=1}^{N} u_{ik}^m(t) x_k}{\sum_{k=1}^{N} u_{ik}^m(t)}; \tag{6}$$

```
end;
3.2 For i = 1 until c do:
  3.2.1 For k = 1 until N do:
    3.2.1.1 Update the membership matrix:
```

$$u_{ik}(t+1) = \frac{1}{\sum_{j=1}^{c} \left(\frac{\|x_k - v_i(t)\|}{\|x_k - v_j(t)\|} \right)^{2/(m-1)}}; \tag{7}$$

```
  end;
end;
3.3 t = t + 1;
While ‖U(t+1)-U(t)‖ ≤ ε;
```

Algorithm 1: Fuzzy *C-means* pseudo-code

3.3 Weighted Fuzzy C-means

Weighted Fuzzy C-means (WFCM) algorithm performs data clustering by minimizing the following cost function [3]:

$$WJ(U,V) = \sum_{i=1}^{c} \sum_{k=1}^{N} w_k u_{ik}^m \|x_k - v_i\|^2. \tag{8}$$

The function presented in (8) is a modification of (5) that allows a weight w_k to be attributed to each data sample x_k.

The WFCM pseudo-code is analogous to that presented in section 3.2, except for the equation that computes the prototypes - i.e. equation (6) must be replaced by:

$$v_i(t) = \frac{\sum_{k=1}^{N} w_k u_{ik}^m(t) x_k}{\sum_{k=1}^{N} w_k u_{ik}^m(t)}. \tag{9}$$

3.3.1 Strategy to find the weights of WFCM

Two different strategies were conceived to find the weights w_k in the WFCM. The first used the information about the density of neighborhood around each sample, given a distance threshold. This approach showed good results only when the number of sources was small. This fact justified a second and more complex approach, which is to use Gaussian functions for each cluster center. The functions are positioned in a way that cluster centers are aligned with their peaks. The weight of a sample is given by the sum of the values of the functions that refer that sample. All the samples' weights are normalized so that their values do not exceed one.

Another matter is how to choose the dispersion of the Gaussian functions. In the first steps of the algorithm, they are relatively large, but are reduced along the iterations, giving more importance to the samples closer to the center.

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

4 Simulations and Results

In order to analyze the performance of the WFCM in the problem of mixing matrix estimation, the algorithm was compared to the *k-means* and to a variant with α = 1, to the *fuzzy c-means* with *m* = 1 and 2 and to the DEMIXN [8], a classical SCA tool.

Using *P* sources randomly chosen among a set of 200 five-second duration excerpts of Polish voices, all of them sampled at 4 kHz[2], two mixtures (*M* = 2) were generated at each time, with *P* varying from 2 to 12. All possible θ_c's were calculated according to (3), using an average of 100 samples, except for the DEMIXN, which used 10 samples in most cases. The quality of the results is measured using the Angular Mean Error (AME) between the estimates and the true directions of **A**, as described in [8]. For each case, 100 simulations were run.

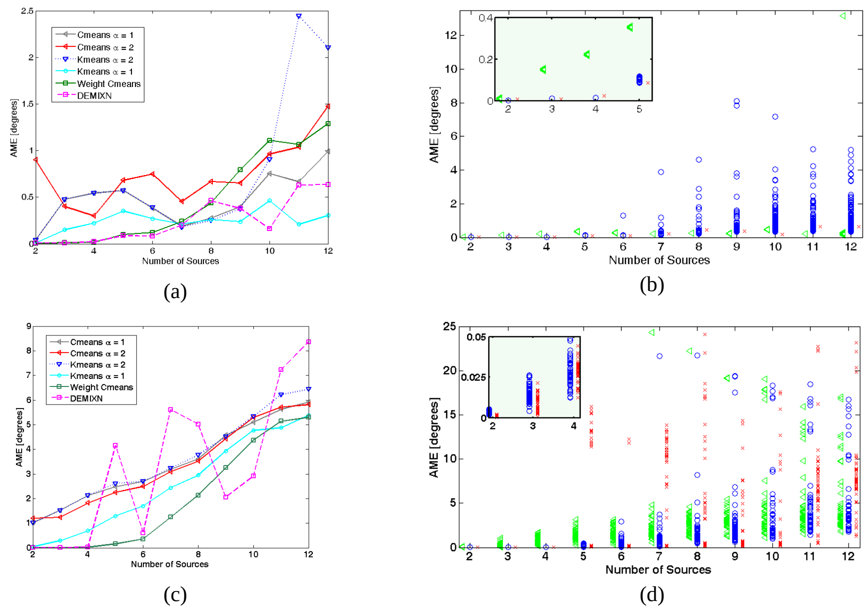

Fig. 1. AME of the estimation of the mixing matrix using *k-means* with α = 1 and 2, FCM with m = 1 and 2, WFCM and to the DEMIXN [8]. On the left, the mean 100 runs for all methods from 2 to 12 sources: (a) with the same mixing matrix, with its directions equally spaced, in all runs; and (c) with the mixing matrix plus a Gaussian disturbance. On the right, the dispersion of all runs for 3 methods: *k-means* with α = 1 (◀), WFCM (○) and DEMIXN (□) with a zoomed window for the first cases: (b) with the same mixing matrix, and (d) with mixing matrix plus disturbance.

In the first case, **A** is the same for all simulations, and was generated in order to have all the dictions of its columns equally spaced. Fig. 1.a shows the mean of the AME. Between 2 and 6 sources, both WFCM and DEMIXN show similar results. From 7 sources on, *k-means* with α = 1 is the best in most cases. It is important to

[2] Available at http://mlsp2005.conwiz.dk/index.php@id=30.html

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7. Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

notice that the number of samples used for DEMIXN had to be changed, for 7 and 11 sources, to 100 and 20, respectively. To allow a better understanding of the results, Fig. 1.b was plotted showing the distribution of all simulations for 3 methods: *k-means* with $\alpha = 1$ (◄), WFCM (○) and DEMIXN (□). DEMIXN converges to the same result in all cases, and *k-means* has a small dispersion in the results. This is not the case for the WFCM, that has excellent results until 5 sources (see zoom detail in the figure), but a larger dispersion from 8 sources.

Considering this results, a second scenario was defined. The mixing matrix is identically generated, but now, in each run, a Gaussian random perturbation is imposed to the directions of the columns, being the rest kept unchanged. Fig. 1.c and Fig. 1.d show the same results now for this new model of **A**. Now, it is noticeable that DEMIXN was not robust, having the worst performance in many situations. On the other hand, WFCM shows results significantly more robust until 9 sources.

5 Conclusions and Perspectives

This work presented a comparative analysis of clustering methods for SCA and proposed the use of the Weighted Fuzzy C-means to estimate the mixing matrix in underdetermined BSS. The results showed that the WFCM has a performance comparable to that of the other methods for a mixing matrix with uniform column distribution and presents better results when the column distribution has a more stochastic character.

As an immediate perspective for future works, the authors intend to derive a method based on the WFCM to estimate the mixing matrix and the number of sources present in the mixture. The comparative analysis will also be extended to include a broader range of SCA techniques.

References

[1] Romano, J., Attux, R., Cavalcante, C. and Suyama, R. Unsupervised Signal Processing - Channel Equalization and Source Separation, CRC Press, 2010.

[2] Comon, P. and Jutten, C. Handbook of Blind Source Separation: Independent Component Analysis and Applications, Burlington Academic Press, 2010.

[3] Hathaway, R. J. and Hu, Y. Density-Weighted Fuzzy c-Means Clustering. IEEE Transactions on Fuzzy Systems, v. 17, n. 1, pages 243-252, 2009.

[4] Abrard, F., Deville Y. and White, P. From Blind Source Separation to Blind Source Cancellation in the Underdetermined Case: A New Approach Based on Timefrequency Analysis. Proceedings of ICA 2001, pages 734-739, USA, 2001.

[5] Haddad, D.B., Petraglia, M.R., Batalheiro, P.B., Filho, J.C.P., Sparse Component Analysis: Blind Source Separation and Applications (in Portuguese). Journal of the Brazilian Neural Network Society, Vol. 10, Iss. 1, pp. 19-35, 2012.

[6] Chinrungrueng, C. and Sequin, C. Optimal Adaptive K-Means Algorithm with Dynamic Adjustment of Learning Rate. IEEE Transactions on Neural Networks, v. 6, n. 1, pages 157-169, 1995.

[7] Pedrycz, W. and Gomide, F. Fuzzy Systems Engineering: Toward Human-Centric Computing. John Wiley & Sons, Inc, 2007.

[8] Arberet, S., Gribonval, R., Bimbot, F.: A Roubust Method to Count and Locate Audio Sources in a Stereophonic Linear Instantaneous Mixture. In: Proc. ICA 2006. USA , 2006.