

Underdetermined Mixture Matrix Estimation Based on Neural Network and Genetic Algorithm

Shuang Wei^{1,2(✉)}, Jian Peng², Feng Wang², Chungui Tao², and Defu Jiang²

¹ College of Information, Mechanical and Electrical Engineering,
Shanghai Normal University, Shanghai, China
wishuang@gmail.com

² Array and Information Processing Laboratory, College of Computer and Information,
Hohai University, Nanjing, China

Abstract. This paper proposes an improved approach to estimate the underdetermined mixture matrix to improve the performance of underdetermined blind source separation (UBSS) for speech sources. This approach only use two observed signals and consider a tangent value instead of each vector of the mixture matrix for estimation. An improved clustering method based on competitive neural network and genetic algorithm is then designed to estimate these tangent values. In the proposed method, those tangent values are designed as clustering centers. The competitive neural network is used first to obtain the initial clustering centers, and genetic algorithm is applied to search for the global optimum around the initial clustering centers. Experimental results show that the tangent values of the observed vectors have better clustering characteristics, which could reduce the computational complexity for mixture matrix estimation. The improved clustering algorithm based on neural network and genetic algorithm can estimate a better mixture matrix with high precision than the general neural network clustering algorithm, and it can improve the performance of underdetermined blind signal separation.

Keywords: Underdetermined blind source separation · Tangent value · Competitive neural network · Genetic algorithm

1 Introduction

Underdetermined blind source separation (UBSS) technology is widely concerned as an effective speech signals processing tool [1–3]. It can help to recovery the unknown source signals from the observed speech signal that is the mixture of the sound sources, in the case that the number of observed signals is much less than that of sources.

Generally, sparse component analysis (SCA) approach is used to solve the UBSS problem after transforming the unknown parameters into a sparse domain [4]. In the SCA approach, the underdetermined mixture matrix has to be estimated before recovering the unknown sources. Thus, the result of mixture matrix estimation plays an important effect on the performance of SCA approach. The task of this paper is to design

an improved algorithm for mixture matrix estimation with higher precision which benefits to improve the recovery of the speech signals.

To solve the underdetermined mixture matrix estimation problem, K-means clustering approach is a popular idea to solve the vectors of the mixture matrix as the clustering centers [5]. However, traditional K-means clustering methods perform poor for the exception data, and the convergence efficiency is still required to be improved. In order to improve the clustering performance, the optimization method is generally used to solve the K-means clustering problem. Thus, this paper proposes a clustering approach with the combination of neural network and genetic algorithm. In the proposed approach, neural network algorithm is used to generate the initial population for genetic algorithm, and genetic algorithm is designed to obtain the global optimum for clustering. Thus, the proposed approach can improve the global search ability and avoid falling into the local optimal.

In this paper, the UBSS problem model and the preprocessing of the observed signal are given in the Sect. 2. Section 3 introduces the proposed clustering algorithm, which is combined with the neural network algorithm [6] and genetic algorithm [7]. Simulation results of four speech source signals and two observed signals are presented in Sect. 4. Finally, conclusions and future work are summarized in Sect. 5.

2 Problem Statement

2.1 Model

Assume an underdetermined blind source separation model is given as

$$\mathbf{X}(t) = \mathbf{A}\mathbf{S}(t) \quad t = 1, 2, \dots, T \quad (1)$$

where $\mathbf{X}(t) = [x_1(t), x_2(t), \dots, x_m(t)]^T$ denotes the observed signals, \mathbf{A} is the mixture matrix with the size of $m \times n$, and $\mathbf{S}(t) = [s_1(t), s_2(t), \dots, s_n(t)]^T$ denotes the source signals. In the underdetermined system, the number of the source signals is larger than that of the observed signals, i.e., $n > m$. Among them, only $\mathbf{X}(t)$ is known, \mathbf{A} and $\mathbf{S}(t)$ are unknown without any prior information. Our task is to estimate the mixture matrix \mathbf{A} from the observed signals $\mathbf{X}(t)$.

When the sparsity of speech signals are poor in the time domain, the speech source signals should be transformed into a sparse domain as (2).

$$\wp\{\mathbf{X}(t)\} = \mathbf{A}\wp\{\mathbf{S}(t)\} \quad (2)$$

where $\wp\{\bullet\}$ denotes a method to transform the signals in the time-domain to a sparse domain, such as, Discrete Cosine Transform (DCT), Short Time Fourier Transform (STFT), etc. [8]. It depends on the characteristic of source signals $\mathbf{S}(t)$. For example, when voice signals are chosen as the sources, which are mostly unsteady in a long time but have stability in the short time, $\wp\{\bullet\}$ can be the STFT algorithm.

2.2 Preprocessing of Speech Signals

In the signal pre-processing, after signal transformation to the sparse domain, two sides of (1) are simultaneously transferred as (3).

$$\mathbf{X}(\omega) = \mathbf{A}\mathbf{S}(\omega) \quad (3)$$

where $\mathbf{X}(\omega) = [x_1(\omega), x_2(\omega), \dots, x_m(\omega)]^T$ is the spectrum of the observed signals, and $\mathbf{S}(\omega) = [s_1(\omega), s_2(\omega), \dots, s_n(\omega)]^T$ is the spectrum of the source signals. If only one item from the source $s_i(\omega)$ has a non-zero value and the items from other sources are zero at a slot ω , (3) can be abbreviated as:

$$[x_1(\omega) \ x_2(\omega) \ \dots \ x_m(\omega)]^T = a_i \times s_i(\omega) \quad (4)$$

Then, we can get the model (5).

$$s_i(\omega) = \frac{x_1(\omega)}{a_{1i}} = \frac{x_2(\omega)}{a_{2i}} = \dots = \frac{x_m(\omega)}{a_{mi}} \quad (5)$$

Obviously, when the source vector is sparse, the observed signal has the characteristics of linear clustering, and each clustering corresponds to each column of the mixture matrix. Based on this, n source signals will determine n straight lines, so that mixture matrix estimation problem is transformed into estimating the direction of the clustering line in the observation space. However, the observed points in (4) are distributed in the vicinity of the straight line of clustering, therefore a clustering algorithm has to be proposed to mining the underline clustering characteristics on these points.

In order to reduce the interference among data, absolute values of the observed signals are used instead of real or imaginary part. In addition, mirror processing is applied to improve the processing efficiency. Considering that the points near the origin have no obvious linear clustering characteristics, a specific radius can be selected according to the specific data, and data points inside the circle are filtered out. The selected data is then shown in Fig. 1.

The points in Fig. 1 shows the two-dimensional observation signals after short-time Fourier transform. It is found that the points in the same cluster have almost the same linear slope, and this conclusion is also proved by (5). Thus, the tangent value of every point is a good index for distinguishing the clusters. The closer the tangent values of two points are, the high possibility they belong to the same cluster. According to (5), it is known that the tangent value of all points in each cluster is the same as the tangent value of each column in the mixture matrix. Therefore, the problem of solving mixture matrix is converted to a clustering analysis of two-dimensional observation signals. In the clustering analysis, the one-dimensional tangent values are designed as the clustering centers instead of two-dimensional signals. It could simplify the calculation load. The tangent distribution is shown in Fig. 2.

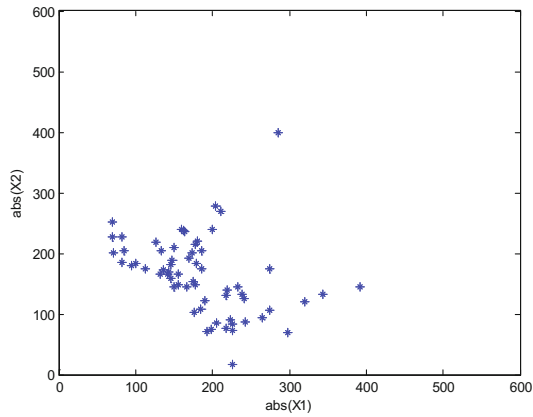


Fig. 1. Scatter distribution after filtrating

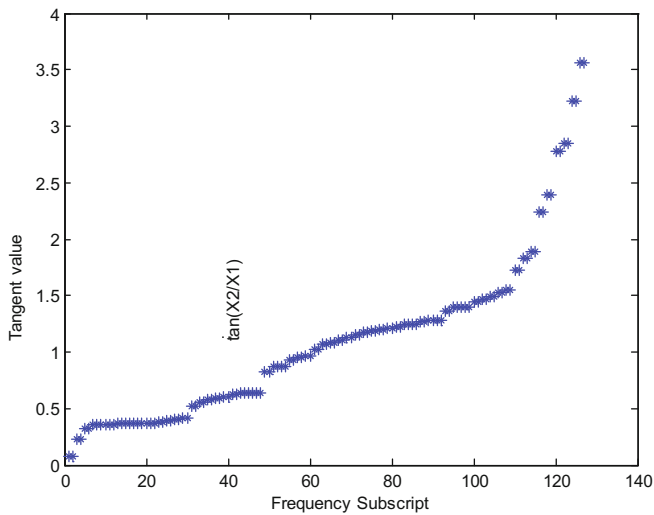


Fig. 2. Distribution of tangent value

From Fig. 2, it is seen that several values change slowly, it means that these values are concentrated in the vicinity of a classification value, which shows the clustering characteristic. Therefore, for two observed signals, it's feasible to use the tangent values of each column of mixture matrix corresponding to the two observed vectors as the input data of the cluster analysis.

At last, in order to reduce the clustering error, it is necessary to perform normalization to project the tangent values to the interval of $[-1, 1]$.

3 Improved Clustering Algorithm Based on Neural Network and Genetic Algorithm

The number of the clustering centers is the same as the column values of mixture matrix. Assume $n = 4$, then the unknown mixture matrix can be calculated by:

$$\hat{A} = \begin{bmatrix} \cos \hat{\alpha}_1 & \cos \hat{\alpha}_2 & \cos \hat{\alpha}_3 & \cos \hat{\alpha}_4 \\ \sin \hat{\alpha}_1 & \sin \hat{\alpha}_2 & \sin \hat{\alpha}_3 & \sin \hat{\alpha}_4 \end{bmatrix} \tag{6}$$

where $\hat{\alpha}_i = \arctan C_i, i = 1, 2, 3, 4$. The input data of the neural network clustering algorithm are all the tangent values.

In order to solve four clustering centers, the output layer of neural network is set with four neurons. The normalized tangent value is one-dimensional input data, so the input layer of the neural network is set with one neuron.

The number of training iterations of neural network is initialized to ensure the applicability of clustering of samples with large data volumes. After finishing the training of neural network of all iterations, the distribution of all the tangent values are learned. Then we can do the clustering analysis with the normalized tangent values after training.

In order to improve the clustering accuracy, genetic algorithm (GA) is then used to optimize the preliminary clustering results obtained by the competitive neural networks above. The clustering problem is converted to an optimization problem [9].

We use the coding approach of the floating point to present the clustering centers. The genetic individuals are coded as below.

$$pop(k).gene = [C_1 \ C_2 \ C_3 \ C_4] \tag{7}$$

Where $pop(k)$ is the k -th individual in the population and the four floating-point numbers denote the values of four cluster centers respectively.

The optimization objective is calculated by the clustering performance, which can be evaluated by the mean of the squares of the distance of clustering points to the respective clustering centers, that is,

$$pop(k).value = \sum_{i=1}^4 \left(\sum_{j=1}^{S_i} (x_{ij} - C_i)^2 \right) / S_i \tag{8}$$

Where x_{ij} is marked as the j -th data in class i, C_i is a cluster center of class i, S_i is the total number of data points in class i . Then the individuals are sorted in ascending order, and the number of each individual is represented as $pop(k).index$. The *fitness* is calculated as:

$$pop(k).fitness = a(1 - a)^{pop(k).index-1} \tag{9}$$

where parameter a is commonly set as $a = 0.6$.

In the selection step, a selection array $cFitness[popSize]$ is built based on the calculated fitness value $pop(k).fitness$ of the individual. The formula is as follows:

$$cFitness[k] = \frac{\sum_{u=1}^k pop(u).fitness}{S} \tag{10}$$

where $S = \sum_{k=1}^{popSize} pop(k).fitness$. Then a random number $randn$ is generated on interval $(0,1)$, and the k -th individual is copied into the population in the next generation if $rand < cFitness(k)$.

In the crossover step, the clustering centers of two individuals are sorted in ascending before crossover. The calculation of the new individual is as follows:

$$pop(k_{new}).gene = a_1 \times pop(k_1).gene + a_2 \times pop(k_2).gene \tag{11}$$

where parameters a_1 and a_2 are 0.6 and 0.4, respectively.

When a mutation is made, a random number is generated on the interval $(0,1)$ for each gene of all individuals in the population. When the random number is less than the mutation probability, -10% to 10% of its own floating value is added randomly.

The flow chart of the above proposed algorithm based on neural network and genetic algorithm is shown in Fig. 3. After that, mixture matrix is calculated by (6) using the tangent values that are the optimum result of the clustering center.

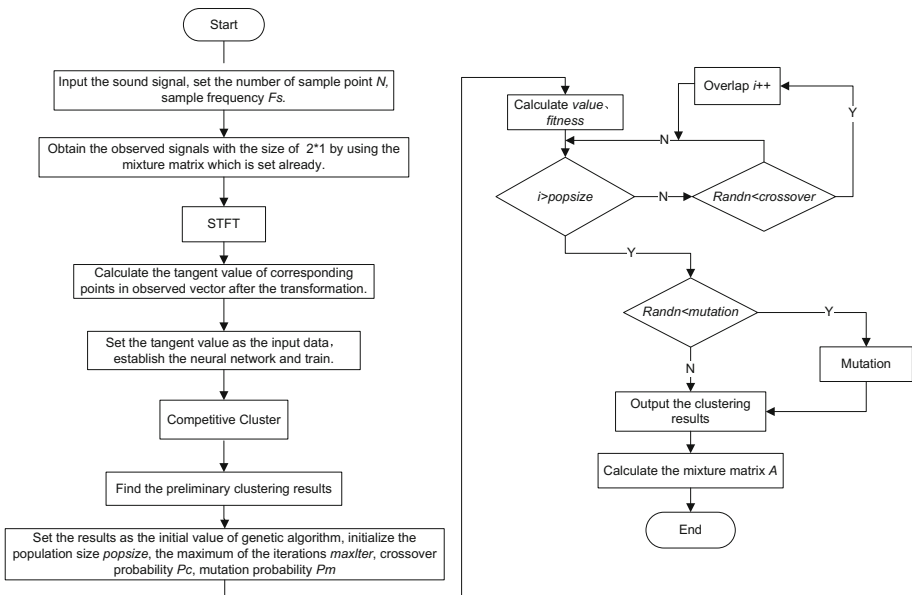


Fig. 3. The flow chart of the proposed algorithm

4 Simulation Results

The experiments of four voice signals are carried out in Matlab 2010b. The speech examples are available on-line in [10]. The signal format is .wav, the coding method is PCM and the sampling point is selected as 65365. The mixture matrix is defined which mix the four source signals into two observation signals as follows:

$$A = [a_1 \ a_2 \ a_3 \ a_4] = \begin{bmatrix} \cos 18^\circ & \cos 36^\circ & \cos 54^\circ & \cos 72^\circ \\ \sin 18^\circ & \sin 36^\circ & \sin 54^\circ & \sin 72^\circ \end{bmatrix} \tag{12}$$

Figure 4 shows the classification result of the tangent values by neural network clustering, where the number of iterations is set to 400, the adjustment step length of weight is initialized to 0.2. It is seen that four classes are clearly separated. In order to illustrate the performance, the clustering center obtained by neural network (NN) is compared with the standard setting shown in Table 1.

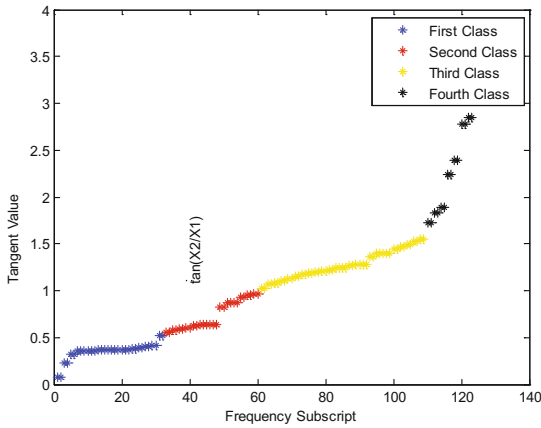


Fig. 4. The classification result of the tangent values

Table 1. Comparison between Neural Network clustering result and the standard setting

| Clustering center | C1 | C2 | C3 | C4 |
|----------------------|--------|--------|--------|--------|
| Standard setting | 0.3249 | 0.7266 | 1.3763 | 3.0780 |
| NN clustering result | 0.3538 | 0.7774 | 1.2170 | 2.1417 |
| Error ratio | 8.91% | 6.99% | 11.57% | 30.42% |

According to Table 1, it is founded that the result of clustering has certain errors compared with the standard setting if only neural network is used for clustering. The result of the estimation \hat{A}_{NN} is obtained as

$$\hat{A}_{NN} = \begin{bmatrix} 0.9427 & 0.7895 & 0.6349 & 0.4231 \\ 0.3335 & 0.6138 & 0.7726 & 0.9061 \end{bmatrix} \quad (13)$$

In the general neural network, the optimal strategy uses gradient descent method, which tends to let the solution fall into the local optimum. So, in the proposed method, genetic algorithm and neural network are combined to reduce the probability of falling into the local optimum. Considering that the clustering solution of neural network is stable and convergent, it can be considered that the error range of the result is valid. Thus, it proved that the clustering results of the neural network can be used as the initial solution, genetic algorithm (GA) is then used to search for the better optimum near the results obtained by neural network algorithm. In addition, GA can simultaneously optimize multiple interval regions near the local optimum, which can improve the convergence efficiency. Therefore, the proposed method can obtain better solutions with less error ratio than those of the single neural network method. The results of the proposed method are shown in Table 2.

Table 2. Comparison between Neural-Genetic clustering result and the standard setting

| Clustering center | C1 | C2 | C3 | C4 |
|-------------------------|--------|--------|--------|--------|
| Standard setting | 0.3249 | 0.7266 | 1.3763 | 3.0780 |
| NN-GA clustering result | 0.3427 | 0.7340 | 1.2743 | 2.4994 |
| Error ratio | 5.46% | 1.02% | 7.41% | 18.81% |

In contrast to Tables 1 and 2, it is found that the error ratio of C1, C2, C3, and C4 are improved by the proposed algorithm. The estimation result by the proposed algorithm based on neural network and genetic algorithm is obtained as

$$\hat{A}_{NN-GA} = \begin{bmatrix} 0.9460 & 0.8061 & 0.6173 & 0.3715 \\ 0.3242 & 0.5917 & 0.7867 & 0.9284 \end{bmatrix} \quad (14)$$

5 Conclusion

This article mainly researches on an improved algorithm of mixture matrix estimation to solve the problem of underdetermined blind speech separation. In the preprocessing section, the two-dimensional observation is converted into one-dimensional tangent value, it is found to have clustering characteristics from the figure of scatter points. The clustering analysis is carried out by the neural network algorithm to obtain the initial clustering center. Then genetic algorithm is used to optimize the result of neural network algorithm, which improves the stability and convergence of clustering. Experimental results of four speech signals prove that the proposed algorithm can achieve the mixture matrix with better accuracy than neural network, and adoption of the tangent value is useful in the process of clustering analysis. In the future, we will improve the proposed algorithm in different cases, such as, it could be applied to separate the power signals to improve the power quality.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (Grant No. 61401145), the Natural Science Foundation of Jiangsu Province (Grant No. BK20140858, BK20151501), and the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

References

1. Bel, A.J., Sejnowski, T.J.: An information-maximization approach to blind separation and blind deconvolution. *Neural Comput.* **7**(6), 1129–1159 (1995)
2. Zou, L., Chen, X., Ji, X., Wang, Z.J.: Underdetermined joint blind source separation of multiple datasets. *IEEE Access* **5**, 7474–7487 (2017)
3. Zhen, L., Peng, D., Yi, Z., Xiang, Y., Chen, P.: Underdetermined blind source separation using sparse coding. *IEEE Trans. Neural Netw. Learn. Syst.* **PP**(99), 1–7 (2015)
4. Li, Y., Cichocki, A., Amari, S.I.: Sparse component analysis for blind source separation with less sensors than sources. In: *Independent Component Analysis* (2010)
5. Huang, X., Ye, Y., Zhang, H.: Extensions of Kmeans-type algorithms: a new clustering framework by integrating intracluster compactness and intercluster separation. *IEEE Trans. Neural Netw. Learn. Syst.* **25**(8), 1433–1446 (2014)
6. Cichocki, A., Kasprzak, W., Amari, S.I.: Neural network approach to blind separation and enhancement of images. In: *1996 8th European Signal Processing Conference (EUSIPCO 1996)*, Trieste, Italy, pp. 1–4 (1996)
7. Pradhan, D., Wang, S., Ali, S., Yue, T., Liaaen, M.: CBGA-ES: a cluster-based genetic algorithm with elitist selection for supporting multi-objective test optimization. In: *2017 IEEE International Conference on Software Testing, Verification and Validation (ICST)*, Tokyo, Japan, pp. 367–378 (2017)
8. Aissa-El-Bey, A., Linh-Trung, N., Abed-Meraim, K., Belouchrani, A., Grenier, Y.: Underdetermined blind separation of nondisjoint sources in the time-frequency domain. *IEEE Trans. Signal Process.* **55**(3), 897–907 (2007)
9. Zhang, M., Li, X., Peng, J.: Blind source separation using joint canonical decomposition of two higher order tensors. In: *2017 51st Annual Conference on Information Sciences and Systems (CISS)*, Baltimore, MD, pp. 1–6 (2017)
10. Bofill, P., Zibulevsky, M.: Sound examples. <http://www.ac.upc.es/homes/pau/>