

Análise Comparativa de Metodologias de Inteligência Computacional para Identificação do Sistema Misturador em Análise de Componentes Esparsos

Rodrigo C. Silva¹, Everton Z. Nadalin², Romis Attux¹

¹Departamento de Engenharia de Computação e Automação Industrial (DCA)

²Departamento de Microondas e Óptica (DMO)

Faculdade de Engenharia Elétrica e de Computação (FEEC)

Universidade Estadual de Campinas (UNICAMP)

Avenida Albert Einstein, 400 – 13083-852 – Campinas – SP – Brasil

{silvarod, nadalin, attux}@dca.fee.unicamp.br

1. Introdução

O problema de separação cega de fontes (*Blind Source Separation* – BSS) consiste, fundamentalmente, em separar um conjunto de sinais $s(n)$ (denominados *fontes*) sujeitos a alguma modalidade de mistura, a partir dos sinais medidos $x(n)$, com o mínimo de conhecimento sobre ambos [Romano et al. 2010] [Comon e Jutten 2010].

Uma possível fonte de informação *a priori* para a construção de uma metodologia de BSS é o grau de esparsidade das fontes. Um sinal é considerado esparsos quando a maior parte de sua energia está concentrada em uma pequena parte dos seus coeficientes [Comon e Jutten 2010]. Ao contrário do que ocorre em contextos de separação tradicionais, quando os sinais são esparsos, é possível, pelo menos idealmente, recuperar as fontes mesmo em situações em que seu número é maior do que o de misturas (caso chamado de *subparametrizado*) [Comon e Jutten 2010]. Esta suposição de esparsidade é relevante do ponto de vista prático, pois sinais esparsos ocorrem no âmbito de domínios amplos como o de processamento de áudio [Comon e Jutten 2010].

As técnicas que se baseiam no grau de esparsidade dos sinais de interesse são parte do paradigma denominado análise de componentes esparsos (*Sparse Component Analysis* – SCA). Um problema resolvido em termos de SCA normalmente é dividido em três etapas: estimação do número de fontes, estimação do sistema misturador e recuperação das fontes. Os problemas de estimação do número de fontes e do sistema misturador podem ser tratados como problemas de agrupamento de dados ou clusterização, sendo que o número de clusters indica a quantidade de fontes presentes na mistura e seus centroides indicam as direções em que os sinais das fontes estão projetados neste sistema misturador [Bofill e Zibulevsky 2001]. O foco deste trabalho está exatamente na utilização de métodos de clusterização para estimação do sistema misturador.

2. Estimação da Matriz de Mistura através de SCA

Na formulação canônica de BSS, considera-se o sistema misturador como sendo linear, instantâneo e invariante no tempo. Neste caso, o modelo de mistura é representado por

$\mathbf{x}(n) = \mathbf{A}\mathbf{s}(n)$, sendo a matriz \mathbf{A} uma matriz de mistura com dimensão $M \times N$, onde N é o número de fontes e M o número de medições. O método que norteia este trabalho, pioneiramente proposto por [Bofill e Zibulevsky 2001], tem como premissa a existência de uma única fonte ativa a cada intervalo de tempo em um contexto tempo-frequencial. A seguir, é apresentada uma versão adaptada desse para o caso de $M = 2$ e N fontes.

Considerando a transformada tempo-frequencial dos sinais das misturas $x_i(n)$, pode-se gerar $\alpha(t, f)$ como sendo a razão entre as amostras $X_i(t, f)$, dada por:

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

Se, em uma determinada região da representação tempo-frequencial (t_c, f_c) , uma única fonte S_j estiver ativa, então:

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

que é justamente a razão entre os coeficientes de uma das colunas da matriz de mistura. Consequentemente, o valor do arco-tangente desta razão indicará um ângulo θ_j , referente à direção daquela coluna. Uma vez calculados os ângulos para cada intervalo de tempo, uma das formas de se estimar a matriz de mistura é através da clusterização dos valores calculados.

3. Algoritmos de Clusterização *k-means* e *fuzzy c-means*

O algoritmo de clusterização *k-means* pode ser entendido como um método de busca padrão-a-padrão ou em batelada por centros de *clusters* que minimizem a seguinte função objetivo [Pedrycz e Gomide 2007][Nadalin 2011]:

$$J(C, \alpha) = \sum_{j=1}^N \sum_{k=1}^K u_{kj} \|\theta_j - C_k\|^\alpha, \quad (3)$$

onde N é o número de padrões de entrada, K é o número total de *clusters* e C_k representa o centro do k -ésimo *cluster*; $u_{kj} = 1$ se $\theta_j \in C_k$, em caso contrário, $u_{kj} = 0$.

O algoritmo *fuzzy c-means* procura agrupar os dados minimizando a seguinte função objetivo [Pedrycz e Gomide 2007]:

$$J(U, V) = \sum_{i=1}^c \sum_{k=1}^N u_{ik}^m \|x_k - v_i\|^2, \quad (4)$$

onde $U = [u_{ik}]$ é a matriz de pertinências; $V = \{v_i\}$ é o conjunto de protótipos dos *clusters* e m é o coeficiente que define o impacto dos graus de pertinência nos *clusters*.

4. Resultados Preliminares e Conclusões

Para executar os ensaios, foram geradas misturas a partir de um conjunto¹ de sinais de vozes amostrados em 4 KHz. Todos os ensaios foram feitos similarmente aos propostos em [Nadalin et al., 2009], com 2 misturas e variando o número de fontes entre 2 e 15.

¹ Retirados de <http://mlsp2005.conwiz.dk/index.php?id=30.html>

Em cada execução, as fontes foram sorteadas dentro do conjunto existente. As matrizes de mistura foram geradas de forma que suas colunas possuíssem direções igualmente espaçadas. Em seguida, eram geradas as representações tempo-frequenciais das misturas para o cálculo dos ângulos candidatos. Para os testes propostos, foram aplicados os algoritmos *k-means* tradicional (com $\alpha=2$), o *k-means* modificado, como utilizado em [Nadalin et al., 2009] (com $\alpha=1$) e *fuzzy c-means* (com $m=2$) para se encontrar as estimativas para as direções da matriz. Os resultados, avaliados em termos do Erro Angular Médio (EAM) [Nadalin et al., 2009] entre as estimativas e as direções verdadeiras das colunas de **A**, são mostrados na Figura 1.

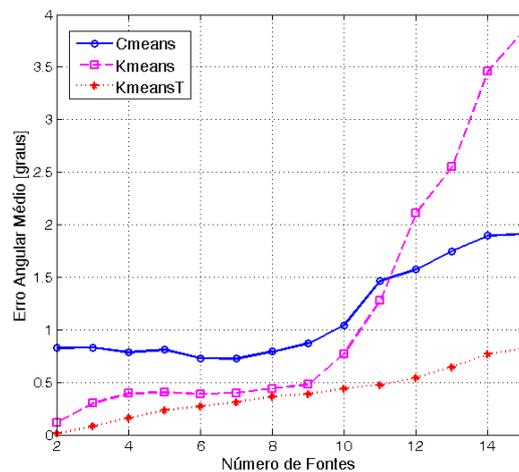


Figura 1 – Erro Angular Médio onde Cmeans representa o *fuzzy c-means*, Kmeans representa o *k-means* com $\alpha=2$, e KmeansT representa o *k-means* com $\alpha=1$.

Os resultados mostram que, apesar de obter piores resultados até 11 fontes, o *fuzzy c-means* se mostrou mais robusto ao aumento do número de fontes, se comparado ao *k-means* tradicional. Vê-se ainda que o uso de $\alpha=1$ parece ser uma opção robusta, o que torna importante analisar essa possibilidade também com o *fuzzy c-means*.

5. Referências

- Comon, P. e Jutten, C. (2010). “Handbook of Blind Source Separation: Independent Component Analysis and Applications”. Burlington: Academic Press.
- Bofill, P. e Zibulevsky, M. (2001) “Underdetermined Blind Source Separation Using Sparse Representation”. Signal Processing Vol. 81, p. 2.353-2.363.
- Nadalin, E., Suyama, R. e Attux, R. (2009) “An ICA-based Method for Blind Source Separation in Sparse Domains”. Lecture Notes in Computer Science. Vol. 5441. p. 597-604.
- Nadalin, E. (2011) “Contribuições ao Problema de Separação Cega de Fontes com Ênfase no Estudo de Sinais Esparsos”. Tese de Doutorado. UNICAMP.
- Pedrycz, W. e Gomide, F. (2007) “Fuzzy Systems Engineering: Toward Human-Centric Computing”. John Wiley & Sons, Inc.
- Romano, J., Attux, R., Cavalcante, C. e Suyama, R. (2010) “Unsupervised Signal Processing - Channel Equalization and Source Separation”. CRC Press.