Learning Overcomplete Dictionaries with ℓ_0-Sparse Non-negative Matrix Factorisation

Ken O'Hanlon, Mark D. Plumbley
Centre for Digital Music
Queen Mary University of London
London, UK
{keno, Mark.Plumbley}@eecs.qmul.ac.uk

***Abstract*—Non-negative Matrix Factorisation (NMF) is a popular tool in which a 'parts-based' representation of a non-negative matrix is sought. NMF tends to produce sparse decompositions. This sparsity is a desirable property in many applications, and Sparse NMF (S-NMF) methods have been proposed to enhance this feature. Typically these enforce sparsity through use of a penalty term, and a ℓ_1 norm penalty term is often used. However an ℓ_1 penalty term may not be appropriate in a non-negative framework. In this paper the use of a ℓ_0 norm penalty for NMF is proposed, approximated using backwards elimination from an initial NNLS decomposition. Dictionary recovery experiments using overcomplete dictionaries show that this method outperforms both NMF and a state of the art S-NMF method, in particular when the dictionary to be learnt is dense.**

***Index Terms*—sparse, non-negative, dictionary learning, NMF**

I. Introduction

Non-negative Matrix Factorisation (NMF) is a learning algorithm which seeks the approximation

$$\mathbf{M} = \mathbf{DX} \tag{1}$$

where $\mathbf{M} \in \mathbb{R}^{M \times N}$ is the matrix for which the factorisation is sought, $\mathbf{D} \in \mathbb{R}^{M \times K}$ and $\mathbf{X} \in \mathbb{R}^{K \times N}$ are a dictionary matrix containing template atoms and a coefficient matrix with each row containing the activations of the corresponding dictionary atom, respectively, and $\mathbf{M}, \mathbf{D}, \mathbf{X} \geq 0$. NMF was originally proposed by Paatero and Tapper [1], in which it was proposed to minimise a Euclidean distance cost function:

$$\mathcal{C}_E = \|\mathbf{M} - \mathbf{DX}\|_F^2 \tag{2}$$

using Alternating Non-negative Least Squares (ANLS) projections. NMF was popularised by Lee and Seung [2] who proposed using fast multiplicative gradient descent updates instead of the ALS methodology. While NMF algorithms have been proposed for many different cost functions [3], the Euclidean distance NMF is popular for many applications, and several proposals have been proposed for performing fast ANLS by taking various approaches to the Non-Negative Least Squares (NNLS) subproblems [4] [5] [6] [7].

A noted feature of NMF factorisations is that they tend to be sparse, a desirable property that several authors have proposed to augment by introducing a sparse penalty term:

$$\mathcal{C}_S = \frac{1}{2}\|\mathbf{M} - \mathbf{DX}\|_F^2 + \lambda \sum_{n=1}^{N} \|\mathbf{x}_n\|_p \tag{3}$$

where λ is a parameter that controls the sparsity and $\|.\|_p$ is an ℓ_p vector norm. Typically in the NMF literature, a ℓ_1 norm is used as the penalty term on the activation matrix, which can be seen as a non-negative matrix variant of the LASSO [8], or Basis Pursuit Denoising (BPDN) [9]. This was first proposed by Hoyer [10] in the multiplicative update framework. In the ALS framework, Kim and Park [6] proposed to apply a squared ℓ_1-penalty term, $\lambda\|\mathbf{x}\|_1^2$, that can be considered a ℓ_1-penalty that scales to the signal. An ℓ_1 penalty term is effected using an Iterative Soft Thresholding (IST) [11] approach that is known to converge for LASSO / BPDN, in [12], where the authors also propose starting with a large value of λ that is gradually decreased. However, the use of an ℓ_1 penalty may not be optimal in a non-negative framework. It has been shown recently that Thresholded NNLS outperforms non-negative ℓ_1-minimisation, such as LASSO/BPDN due to the innate regularisation of the non-negative constraint [13]. Indeed, in [12] an iterative strategy using hard thresholding was often seen to perform better than the IST approach, the authors noting that the hard thresholding strategy tends towards an ℓ_0 penalty, where $\|\mathbf{x}\|_0 = |\mathbf{x} \neq 0|$.

Other Sparse NMF approaches which seek to approximate a ℓ_0 norm include NMF-ℓ_0 [14] and Non-negative K-SVD [15], characterised by using a different approach to the dictionary update step, such as a modifed K-SVD algorithm [15], or repeated NMF updates [14]. In both cases, pursuit algorithms are used, with an emphasis on matching pursuits, with stopping conditions determined by a predetermined number of atoms, [15] [14], or a relative error measure [15].

The focus of this paper is on the use of an ℓ_0 penalty for sparse NMF. In particular a backwards elimination approach using a modified sparse cost function that we have seen previously to be effective in non-negative sparse decompositions [16] is employed for the sparse aproximation step. While an optimal solution to the problem may not be guaranteed, the backward elimination step is locally optimal, and does not require the use of a predetermined number of atoms, or relative error condition as a stopping condition. In the rest of this paper, the relevant background material is first briefly described before introducing the proposed methodology. Experimental results on synthetic data are offered, which validate the approach taken before concluding with pointers to future work.

 Published in: Proceedings of the 2013 IEEE Global Conference on Signal and Information Processing (GlobalSIP), December 3-5, 2013, Austin, Texas, USA, pp. 977-980. DOI:10.1109/GlobalSIP.2013.6737056

II. Background

A. NMF with ANLS

While NMF was popularised using multiplicative updates, the ANLS approach is generally considered to perform better converging to a lower value of the cost function. The ANLS approach performs alternating NNLS projections:

$$\mathbf{X} \leftarrow \min_{X} \|\mathbf{M} - \mathbf{D}\mathbf{X}\|_F^2 \; s.t. \; \mathbf{X} \geq 0 \tag{4}$$

to update the coefficent matrix and

$$\mathbf{D} \leftarrow \min_{D} \|\mathbf{M}^T - \mathbf{X}^T\mathbf{D}^T\|_F^2 \; s.t. \; \mathbf{D} \geq 0 \tag{5}$$

to update the dictionary. The computational load of using NNLS projections for the subproblems has been noted and variants of the ANLS algorithm have been proposed using projected gradient [5], optimised active set [4], block pivoting [6] and coordinate descent [7] methods in order to counter this computational load. A noted ability of ANLS methods relative to multiplicative update methods is their ability to handle overcomplete dictionaries [7].

B. Backward Elimination

Backwards elimination is a stepwise strategy that starts with an initial set, Γ, of indices of supported atoms, and eliminates an atom with index $\hat{k}$, such that $\Gamma \leftarrow \Gamma \backslash \hat{k}$ at each iteration such that

$$\hat{k} = \arg\min_{k} \Delta^k \mathbf{r} \tag{6}$$

where

$$\Delta^k \mathbf{r} = \|\bar{\mathbf{r}}^k\|_2^2 - \|\mathbf{r}^i\|_2^2 \tag{7}$$

where $\mathbf{r}^i$ is the residual at the current matrix and $\bar{\mathbf{r}}^k$ is the hypothetical residual given the sparse support $\Gamma^k = \Gamma \backslash k$. A fast elimination criteria is proposed as part of the Greedy Sparse Least Squares (GSLS) algorithm [17], derived through using the block matrix inversion formulae:

$$\Delta \mathbf{r} = \hat{\mathbf{x}}^{[2]} \oslash diag([\mathbf{D}_\Gamma^T \mathbf{D}_\Gamma]^{-1}) \tag{8}$$

which calculates $\Delta^k \mathbf{r}$ for all k simultaneously, where $\hat{\mathbf{x}}$ is the least squares solution vector given Γ, the current support, $\mathbf{a}^{[b]}$ denotes elementwise power of $\mathbf{a}$ and $\oslash$ denotes elementwise division.

III. Method

A. Modified Sparse Cost Function

In the sparse representations literature an ℓ_0 penalised least squares solution is considered optimal. In an orthonormal basis [18], hard thresholding using a threshold λ is equivalent to

$$\mathcal{C}_T = \|\mathbf{m} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda^2 \|\mathbf{x}\|_0. \tag{9}$$

We have observed empirically found that in a non-negative framework, that better sparse approximation, using the backwards elimination framework, occurs with a slight modification to this cost function [16] using the residual norm

$$\mathcal{C}_{mod} = \|\mathbf{m} - \mathbf{D}\mathbf{x}\|_2 + \lambda \|\mathbf{x}\|_0 \tag{10}$$

for which the motivation comes from the observation that the backwards elimination criteria (8) scales to the square of $\hat{x}$.

The backwards elimination step (6) can be seen to locally optimise the sparse cost function (9) when $\Delta^{\hat{k}}\mathbf{r} < \lambda^2$. In order to optimise the modified sparse cost function the measure

$$\bar{\Delta}^{\hat{k}}\mathbf{r} = \sqrt{\|\mathbf{r}\|_2^2 + \Delta^{\hat{k}}\mathbf{r}} - \|\mathbf{r}\|_2 \tag{11}$$

can be simply calculated, noting that $\hat{k}$, the index of the selected atom is the same regardless of the cost function applied. The stopping condition for the backwards elimination criteria is then given similarly as $\bar{\Delta}^{\hat{k}}\mathbf{r} < \lambda$.

B. ℓ_0-Sparse NMF

A sparse NMF algorithm is proposed which uses the modified sparse cost (10), an approach referred to as ℓ_0-Sparse NMF, (ℓ_0S-NMF) outlined in Fig. 1. ℓ_0-Sparse NMF, as outlined here follows the ANLS approach. After initialisation of the dictionary $\mathbf{D}$, the ℓ_0S-NMF enters an iterative loop. First NNLS is performed to estimate the coefficient matrix $\mathbf{X}$. Approximation of the minimum of (10) with a non-negative constraint is then performed using backwards elimination at each column of the initial NNLS decomposition.

Input $\mathbf{M} \in \mathbb{R}^{M \times N}$, K, λ
Initialise $\mathbf{D} \in \mathbb{R}^{M \times K}$
repeat
  Calcluate $\mathbf{X}$ using (4)
  for n = 1:N **do**
    $\Gamma_n = \{k | [X]_{k,n} > 0\}$
    repeat
      Select $\hat{k}_n$ using (6) (8)
      $\Gamma_n \leftarrow \Gamma_n \backslash \hat{k}_n$
    until $\bar{\Delta}^{\hat{k}}\mathbf{r}_n < \lambda$
    $\mathbf{x}_n = \arg\min_x \|\mathbf{m}_n - \mathbf{D}_{\Gamma_n}\mathbf{x}\|_2^2 \; s.t. \; \mathbf{x} \geq 0$
  end for
  Update $\mathbf{D}$ using (5)
until stopping condition

Fig. 1: ℓ_0 S-NMF Algorithm

After the elimination process is completed $\mathbf{X}$ is recalculated using NNLS constrained to the new sparse support, Γ. The final step of the iteration consists of re-estimating $\mathbf{D}$, using the transposed NNLS projection (5).

In some cases it may be useful to enforce sparsity in the dictionary, using a cost function such as

$$\mathcal{C}_{mod} = \sum_n \{\|\mathbf{m}_n - \mathbf{D}\mathbf{x}_n\|_2 + \lambda \|\mathbf{x}_n\|_0\} + \eta \|\mathbf{D}\|_0 \tag{12}$$

where $\|\mathbf{D}\|_0$ is the number of non-zero elements in the dictionary. The reason that the ANLS approach is considered is that sparsity of the dictionary can also be enforced using the backwards elimination approach, applied to the transposed NNLS problem (5) with a stopping condition $\bar{\Delta}^{\hat{m}}\mathbf{r} < \eta$.

IV. Experiments

Some synthetic dictionary recovery experiments were designed to test the proposed approach. Random, twice overcomplete non-negative dictionaries $\bar{\mathbf{D}}$ of dimension 200×400 were generated, using a flat equal probability distribution in the range [0,1], and all dictionary columns were normalised to unit ℓ_2 norm. A coefficient matrix $\bar{\mathbf{X}}$ of dimension 200×800 was synthesised using a equal distribution in $[0.021]$. Between 5 and 10 entries of $\bar{\mathbf{X}}$ were randomly selected to be active in each column for all experiments, and all other entries of $\bar{\mathbf{X}}$ were set to zero. Experiments were performed using different sparsity levels in the dictionary, with $\{10, 25, 50\}\%$ of entries set as non-zero.

The matrix $\mathbf{M} = \bar{\mathbf{D}}\bar{\mathbf{X}}$ was synthesised. Subsequent factorisation was performed using different approaches. All approaches use the transposed ANLS approach (5) to perform the dictionary update, while different algorithms are used to estimate the coeffcient matrix $\mathbf{X}$ at each iteration. Each approach was run for 50 iterations of the alternating projection. The proposed ℓ_0S-NMF is used with $\lambda = 0.02$, the minimum value of an activation in the synthesised dictionary. NMF was performed using the ANLS approach. OMP was used as a sparse approximation step, with non-negative constraints applied. OMP stopped iterating when either 15 atoms were selected, or the relative error $\frac{\|\mathbf{r}_n\|_2^2}{\|\mathbf{m}_n\|_2^2} < 0.05$. Thresholded NNLS (T-NNLS) was performed using two different values of the threshold $\lambda = 0.02$ and $\lambda = \sqrt{0.02}$. An ℓ_1-SNMF approach was also performed, with $\lambda = 0.02$, and also with $\lambda = 0.04$ (ℓ_1-SNMF (2λ). For all NNLS calcuations the active set Fast-NNLS [19] method was used, considering each column of $\mathbf{M}$ separately.

The goal of the experiments was to reproduce a similar dictionary using the described NMF techniques. In order to measure the similarity between the original and estimated dictionaries, the simple measure:

$$\mathcal{P} = \frac{\min\{\sum_{k=1}^{K} \max \mathbf{g}_k,\ \sum_{k=1}^{K} \max \mathbf{g}^k\}}{K} \qquad (13)$$

where $\mathbf{G} = \bar{\mathbf{D}}^T\mathbf{D}$, is used. $\mathcal{P}_{max}$ is recorded as the final value of $\mathcal{P}$. A value of $\mathcal{P} = 0.95$ is considered success in these experiments, and an additional measure $\mathcal{I}$, relates the number of iterations taken to achieve $\mathcal{P} = 0.95$

is also tabulated. This threshold of 0.95 is considered success in these experiments.

V. Results

The results for the experiments are shown in Table 1, while Fig. 2 plots the evolution of the average of $\mathcal{P}$ for all experiments of given dictionary sparsity. It is observed that NMF performs poorly in all cases, being unsuccessful for all dictionaries, with the average correlation falling relative to the initialised dictionary. While a stated advantage of the ANLS approach to NMF is that overcomplete dictionaries can be used [7], it would appear not be viable without a sparsity-based approach. The proposed ℓ_0-SNMF approach is seen to be the only algorithm successful in all experiments. However, a small drop-off in performance is observed in the case of the sparsest dictionary, where $\mathcal{P}_{max}$ is reached after around 15 iterations, and not subsequently improved. A sparsity constraint on the dictionary, such as in (12), may improve this performance. The use of OMP was seen to be reasonable, with success seen with both of the sparser dictionaries, and a relatively high value of $\mathcal{P}_{max}$ in the case of the densest dictionary. When a high value of λ was used, the T-NNLS approach was seen to perform well for the denser dictionaries, while failing in the sparsest dictionary. With a lower threshold, $\lambda = 0.02$, the T-NNLS approach was seen to perform similar to NMF, and the results are not recorded. Using an ℓ_1-SNMF approach was seen to be relatively unsuccessful, using the considered parameters. Only when the higher threshold (2\$$\lambda$) was used was success seen in any of the experiments. In this case, however, $\mathcal{P}_{max}$ was seen to be higher than for all other algorithms.

TABLE I
Different NMF algorithms compared for different dictionary sparsity levels with dictionaries synthesised from equiprobable distribution

	50%		25%		10%	
	$\mathcal{P}_{max}$	$\mathcal{I}$	$\mathcal{P}_{max}$	$\mathcal{I}$	$\mathcal{P}_{max}$	$\mathcal{I}$
ℓ_0-S-NMF	0.992	27	0.992	12	0.973	10
NMF	0.408	-	0.507	-	0.660	-
T-NNLS	0.990	17	0.970	14	0.897	-
ℓ_1-SNMF	0.432	-	0.555	-	0.865	-
ℓ_1-SNMF(2λ)	0.476	-	0.893	-	0.995	21
OMP	0.818	-	0.958	15	0.976	11

The effect of the use of a higher threshold is obvious in the results, as seen in the case of T-NNLS, with a high threshold. Some experiments were run with the ℓ_1^2-penalty norm suggested by [6]. The ℓ_1^2-norm can be considered a scaled ℓ_1-norm, and seen to perform well using a value of $\lambda = 0.02$. Considering the synthesis of the coefficient matrices used here, the ℓ_1^2 squared approach is similar to ℓ_1 with a higher value of λ. Other initial experiments performed using high values of λ with the ℓ_1-SNMF approach were seen to bring similar improvements in the dictionary recovery to the ℓ_1^2 approach. These observations would seem to validate the approach taken in [12] where a large value of λ, used at initial iterations, was gradually decreased. However, it is noted that in all cases the proposed ℓ_0 approach performs similarly, without requiring any scaling.

VI. Conclusions

A new variant of Sparse NMF has been presented using a modified sparse cost function with an ℓ_0 penalty with a backwards elimination strategy proposed to perform the approximation. The use of this approach was seen to improve dictionary recovery results in synthetic experiments, and real-world data will be considered in future. The improvement in other approaches by scaling up the sparsity parameter, λ was noted, however, the proposed approach was seen to perform adequately while using the value of λ suggested by the experimental setup.

An extension of this approach to enforce sparsity of the dictionary itself was also proposed, although not implemented

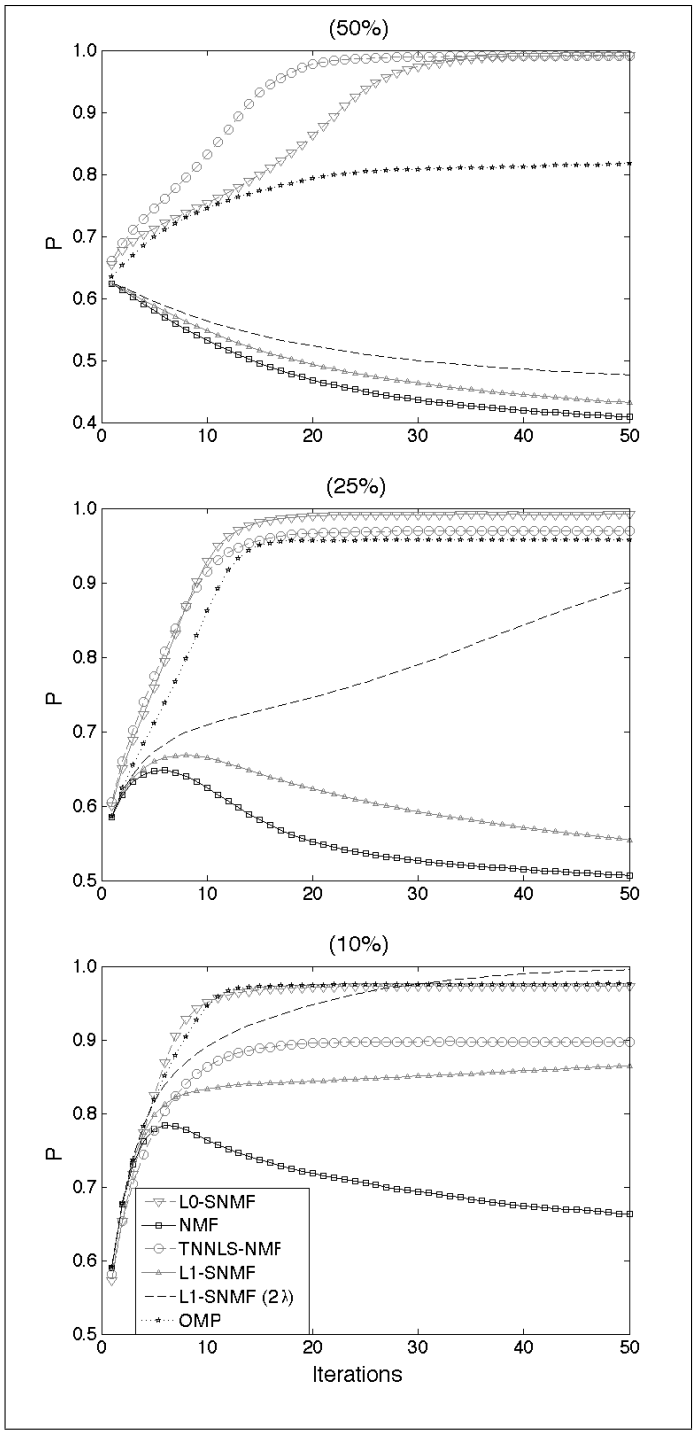

Fig. 2: Comparison of NMF algorithms learning in terms of $\mathcal{P}$ when the dictionary is 50% dense (top), 25% dense (middle) and 10% dense (bottom).

here. Future work will incorporate this approach. It was found that the use of accelerated NNLS algorithms such as [6] was problematic in these experiments, possibly due to the lack of structure between individual matrix columns which might be expected in real-world experiments and also to ill-condition introduced by overcompleteness of the dictionary. Repeated use of the F-NNLS algorithm was seen to be computationally expensive. OMP was seen to perform reasonably well, and use of a bi-directional stepwise optimal pursuit should improve these results, possibly close to that of the backwards elimination approach employed, while reducing the computational load. Such an approach could also be used to solve the transposed problem (5). The backwards elimination approach has been seen to perform well in the case of block sparsity, and an investigation of the use of block sparsity in NMF will be undertaken.

REFERENCES

[1] P. Paatero and U. Tapper, "Positive matrix factorisation: A non-negative factor model with optimal utilization of error," *Environmetrics*, vol. 5, pp. 111–126, 1994.

[2] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems (NIPS) 2000*, 2000, pp. 556–562.

[3] A. Cichocki, S. Cruces, and S. Amari, "Generalized alpha-beta divergences and their application to robust non-negative matrix factorization," *Entropy*, vol. 13, no. 1, pp. 134–170, January 2011.

[4] J. Kim and H. Park, "Non-negative matrix factorization based on alternating non-negativity constrained least squares and active set method," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 2, pp. 713–730, 2008.

[5] D. Kim, S. Sra, and I. S. Dhillon, "Fast projection-based methods for the least squares nonnegative matrix approximation problem," *Statistical Analysis and Data Mining*, vol. 1, no. 1, pp. 38–51, 2008.

[6] J. Kim and H. Park, "Toward faster non-negative matrix factorization: A new algorithm and comparisons," *SIAM Journal on Scientific Computing (SISC)*, vol. 33, no. 6, pp. 3261–3281, 2011.

[7] A. Cichocki and A. H. Phan, "Fast local algorithms for large scale non-negative matrix and tensor factorizations," *IEICE Transaction on Fundamentals*, vol. E92-A(3), pp. 708–721, 2009.

[8] R. Tibrishani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society, Series B*, pp. 267–288, 1994.

[9] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, pp. 33–61, December 1998.

[10] P. Hoyer, "Non-negative sparse coding," in *Proceedings of the 2002 IEEE Workshop on Neural Networks for Signal Processing*, 2002, pp. 557–565.

[11] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.

[12] J. Rapin, J. Bobin, A. Larue, and J. Starck, "Robust non-negative matrix factorization for multispectral data with sparse prior," in *Proceedings of the 7th Conference on Astronomical Data Analysis*, 2012.

[13] M. Slawski and M. Hein, "Sparse recovery by thresholded non-negative least squares," in *In Advances in Neural Information Processing Systems (NIPS 24)*, 2011, pp. 1926–1934.

[14] R. Peharz, M. Stark, and F. Pernkopf, "Sparse nonnegative matrix factorisation using ℓ_0 constraints," in *IEEE Interational Workshop on Machine Learning for Signal Processing*, 2010, pp. 83–88.

[15] M. Aharon, M. Elad, and A. M. Bruckstein, "K-SVD and its non-negative variant for dictionary design," in *Proceedings of the SPIE conference wavelets*, 2005, pp. 327–339.

[16] K. O'Hanlon, N. Keriven, and M. D. Plumbley, "Structured sparsity using backwards elimination for automatic music transcription," in *Proceedings of IEEE Workshop on Machine Learning for Signal Processing*, 2013.

[17] B. Moghaddam, A. Gruber, Y. Weiss, and S. Avidan, "Sparse regression as a sparse eigenvalue problem," in *Information Theory and Applications Workshop, 2008*, February 2008, pp. 219 –225.

[18] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way 3rd edition*. Elsevier, 2009.

[19] R. Bro and S. De Jong, "A fast non-negativity-constrained least squares algorithm," *Journal of Chemometrics*, vol. 11, no. 5, pp. 393–401, September 1997.