EXTENDING NONNEGATIVE MATRIX FACTORIZATION—A DISCUSSION IN THE CONTEXT OF MULTIPLE FREQUENCY ESTIMATION OF MUSICAL SIGNALS

Stanisław Raczyński, Nobutaka Ono, Shigeki Sagayama

Graduate School of Information Science and Technology, The University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
phone: +(81)3-5841-6902, fax: +(81)3-5841-6902,
email: {raczynski,sagayama,onono}@hil.t.u-tokyo.ac.jp
web: http://hil.t.u-tokyo.ac.jp/index-e.html

ABSTRACT

Nonnegative Matrix Factorization (NMF) is a very well known technique of multivariate data analysis. However, in its basic form provides very little control over its behaviour. This article explores possible extensions to this method—results of our work in applying NMF to the task of multiple pitch estimation of musical signals. We have proposed an algorithm for NMF generalized to our proposed r-divergence family of divergences, which includes the Euclidean distancem, the I-divergence and many others. Furthermore, we employ a harmonic constraint and extend the algorithm to include any regularizations. Finally, we provide the reader with three regularizations useful for multiple pitch estimation and propose an objective way of evaluating the performance of NMF-based pitch estimators.

1. INTRODUCTION

Music Information Retrieval (MIR) is a dynamically developing and interdisciplinary field of science that aims at retrieving useful information from musical signals, both acoustic and symbolic. One of the most important tasks that belong to that domain is multipitch analysis—a task of estimating simultanous pitches present in a musical signal. Pitched sounds in most cases consist of a fundamental tone and its overtones—tones with frequencies being integer multiples of the fundamental frequency. The presence of these multiple tones corresponding to simultanous multiple pitched sounds makes this task very difficult. Furthermore, in musical signals, it is very common to have rational proportions between fundamental frequencies of different pitched sounds, as such signals sound much more pleasant to the human ear, and, as a consequence, some pitched sounds share overtones between themselves, which further complicates the analysis process.

One of the most common ways of looking at the problem of multiple pitch estimation is as a matrix factorization, where the spectrogram matrix is approximated with a sum of N rank-1 matrices (equivalently, a product of two matrices). Commonly used factorization technique is the Nonnegative Matrix Factorization, or NMF (proposed in [8] with a fast and convinient algorithm given in [7])). It is a method for decomposing a nonnegative (having only nonnegative elements) matrix $\mathbf{X}$ (later referred to as the data matrix) into a product of two, also nonnegative, matrices $\mathbf{A}$ and $\mathbf{S}$:

$$\mathbf{X} \cong \mathbf{AS} = \widetilde{\mathbf{X}}. \tag{1}$$

This method has gained an indisputable importance for researchers in the Music Information Retrieval field—more than half of last year's (2008) entries to MIREX task of multipitch analysis was based on Nonnegative Matrix Factorization [2].

NMF approximates each column of the data matrix with a linear combination of basis vectors $\mathbf{a}_t$ (columns of $\mathbf{A}$):

$$\widetilde{\mathbf{x}}_t = \sum_n s_{n,t}\mathbf{a}_t, \tag{2}$$

where n is the basis vector number and t is the time index. Because the matrices are nonnegative, only additive mixtures of nonnegative basis vectors, interpreted as individual note spectra, are possible, which is consistent with the way individual note sounds are combined to form the musical signal we aim to analyse. Therefore, we call the coefficient matrix $\mathbf{S}$ a *note activity matrix*, since it is assumed to contain amplitudes of notes.

Different variations and extensions of the NMF algorithm have been used for multipitch analysis: a regular NMF [10], penalized NMFs, such as the Nonnegative Sparse Coding (NNSC) [4, 3], or NMF with basis vectors extended to contain spectrotemporal signatures (a number of consequent data frames), such as Nonnegative Matrix Factor 2-D Deconvolution (NMF2D) and Sparse Nonnegative Matrix Factor 2-D Deconvolution (SNMF2D) [9]. NMF2D and SNMF2D use a single signature for every note (of a single instrument), making use of the shift-similarity of logarithmic frequency scale spectra of notes played on a single instrument. All of these methods, however, when used directly, do not guarantee that the results will be usable for multipitch analysis, i.e. that the basis matrix will contain note spectra and the coefficient matrix their activities. This is true even for a very unrealistic case, when note events occur independently and sparsely, note spectra do not change their harmonic structure over time and the number of basis vectors correspond to the number of different notes actually occurring in the analyzed signal.

In this paper, we discuss a range of methods based on Nonnegative Matrix Approximation (NNMA, described in [6] and later developed in [11]), which generelizes NMF to any Bregman divergences. We have proposed to parameterize the approximation with a single parameter by introducing a family of divergences we called r-divergences, and which include most of the common divergences. We have also proposed different regularization function. This, combined with our harmonic constraints, yield much better results in the sense of pitch estimation accuracy, as measured by our proposed F-measure generalized to real-valued data.

2. NONNEGATIVE MATRIX APPROXIMATION

For the purpose of clarity, the following matrix notation is used throughout this article. $|\mathbf{A}| = \sum_{i,j} A_{i,j}$ is shorthand for sum of all matrix elements (1-norm in case of a nonnegative matrix), and $(\mathbf{A} \odot \mathbf{B})_{i,j} = A_{i,j}B_{i,j}$ is the Hadamard (element-wise) product between two matrices. Unless stated otherwise, all other matrix operations in this paper are also element-wise, in particular matrix division $\frac{\mathbf{A}}{\mathbf{B}}$ and power $\mathbf{A}^p$.

Nonnegative Matrix Approximation (NNMA) is a generalization of NMF to Bregman divergences. A Bregman divergence is defined as:

$$D_\varphi(x,y) = \varphi(x) - \varphi(y) - \varphi'(y)(x-y), \tag{3}$$

where $\varphi : \mathscr{R} \to \mathscr{R}$ is a convex *generating function* with a continuous first derivative. Sum of Bregman divergences is also a Bregman divergence, so the definition can be extended to matrices:

$$D_\varphi(\mathbf{X},\mathbf{Y}) = |\varphi(\mathbf{X}) - \varphi(\mathbf{Y}) - \varphi'(\mathbf{Y})(\mathbf{X}-\mathbf{Y})|. \tag{4}$$

Two most commonly used divergences are the Euclidean distance and the I-divergence (see Table 1).

NNMA is an optimization problem with the penalty function being Bregman divergence between the data $\mathbf{X}$ and its approximation $\mathbf{AS}$ and with a constraint of nonnegativity [6]. It can be easily solved using the Karush-Kuhn-Tucker (KKT) conditions [11], which is a method that generalizes the method of Lagrange multipliers to inequality constraints. Let us first minimize the divergence between the data $\mathbf{X}$ and its approximation $\mathbf{AS}$ w.r.t. $\mathbf{S}$:

$$\min_{\mathbf{S}} f(\mathbf{S}) = \mathbf{D}_\varphi(\mathbf{X},\mathbf{AS}) \tag{5}$$

$$\text{s.t. } \mathbf{g}(\mathbf{S}) = -\mathbf{S} \leq \mathbf{0}. \tag{6}$$

From the stationarity condition we get:

$$\nabla_{\mathbf{S}} f(\mathbf{S}) + \nabla_{\mathbf{S}} |\mathbf{M} \odot \mathbf{g}(\mathbf{S})| = \mathbf{0}, \tag{7}$$

$$\nabla_{\mathbf{S}} D_\varphi(\mathbf{X},\mathbf{AS}) - \mathbf{M} = \mathbf{0}, \tag{8}$$

where $\mathbf{M}$ is a matrix of Lagrange multipliers. The feasibility conditions dictate that $\mathbf{S} \geq \mathbf{0}$ and $\mathbf{M} \geq \mathbf{0}$, and the complementary slackness condition requires that:

$$\mathbf{M} \odot g(\mathbf{S}) = -\mathbf{M} \odot \mathbf{S} = \mathbf{0}. \tag{9}$$

Combining equations 8 and 9 we get:

$$\nabla_{\mathbf{S}} D_\varphi(\mathbf{X},\mathbf{AS}) \odot \mathbf{S} = \mathbf{0}, \tag{10}$$

It can be shown that:

$$\nabla_{\mathbf{S}} D_\varphi(\mathbf{X},\mathbf{AS}) = (\mathbf{AS} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}} - (\mathbf{X} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}}. \tag{11}$$

Substituting this into eq. 10, we arrive at:

$$\mathbf{S} \odot ((\mathbf{X} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}}) = \mathbf{S} \odot ((\mathbf{AS} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}}), \tag{12}$$

which suggests a multiplicative update rule:

$$\mathbf{S} \leftarrow \mathbf{S} \odot \frac{(\mathbf{X} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}}}{(\mathbf{AS} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}}}. \tag{13}$$

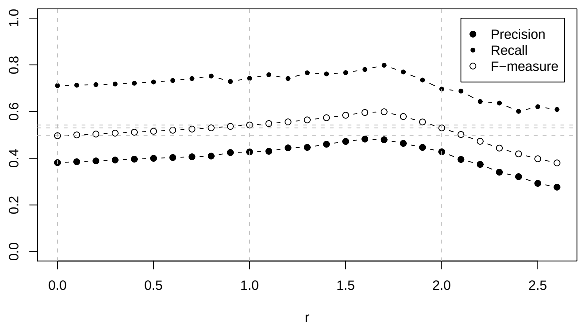


Figure 1: Multiple pitch estimation accuracy for different values of r. Gray dashed lines mark points that correspond to the Euclidean distance, the I-divergence and the Itakura-Saito divergence.

A similar procedure leads to an update rule for the basis matrix $\mathbf{A}$.

$$\nabla_{\mathbf{A}} D_\varphi(\mathbf{X},\mathbf{AS}) \odot \mathbf{A} = \mathbf{0}, \tag{14}$$

$$\nabla_{\mathbf{A}} D_\varphi(\mathbf{X},\mathbf{AS}) = (\mathbf{AS} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}} - (\mathbf{X} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}} \tag{15}$$

$$\mathbf{A} \leftarrow \mathbf{A} \odot \frac{(\mathbf{X} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}}}{(\mathbf{AS} \odot \varphi''(\mathbf{AS}))\mathbf{S}^{\mathbf{T}}}. \tag{16}$$

2.1 *r*-divergence

When Euclidean distance or the I-divergence is used, the NNMA algorithm becomes what is commonly known as the Nonnegative Matrix Factorization. In most practical cases, however, only the I-divergence version is used, as it yields sparser representations. There are, of course, infinitely many Bregman divergences and for the sake of exploring this vast family, we have defined a sub-family of divergences, which we called r-divergences, where r is the shape control parameter.

An r-divergence is a divergence generated by a function, which second derivative is of the following form:

$$\varphi''(x) = x^{-r}. \tag{17}$$

The simplest solution for the generating function is in this case:

$$\varphi(x) = \begin{cases} x\log(x) - x & \text{if} \quad r = 1 \\ -log(x) - 1 & \text{if} \quad r = 2 \\ \frac{x^{2-r}}{(1-r)(2-r)} & \text{otherwise} \end{cases} \tag{18}$$

We can see now that 3 important divergences belong to that family: Euclidean distance for $r = 0$, the I-divergence for $r = 1$ and the Itakura-Saito divergence for $r = 2$ (compare Table 1).

The NNMA algorithm for r-divergences becomes:

$$\mathbf{A} \leftarrow \mathbf{A} \odot \frac{(\mathbf{X} \odot (\mathbf{AS})^{-r})\mathbf{S}^T}{((\mathbf{AS})^{1-r})\mathbf{S}^T}, \tag{19}$$

$$\mathbf{S} \leftarrow \mathbf{S} \odot \frac{\mathbf{S}^T(\mathbf{X} \odot (\mathbf{AS})^{-r})}{\mathbf{S}^T((\mathbf{AS})^{1-r})}, \tag{20}$$

and for $r = 0$ and $r = 1$ these equations become identical to the ones presented in [7].

Divergence	$\varphi(x)$	$D_\varphi(x,y)$	r
r-divergence	$\iint x^{-r}\mathrm{d}^2x$	$\varphi(x)-\varphi(y)-\varphi'(y)(x-y)$	r
Euclidean distance	$\frac{1}{2}x^2$	$\frac{1}{2}(x-y)^2$	0
I-divergence	$x\log x$	$x\log\frac{x}{y}-x+y$	1
Itakura-Saito divergence	$-\log x$	$\frac{x}{y}\log\frac{x}{y}-1$	2

Table 1: Types of divergences used for multipitch analysis with NNMA

3. CONSTRAINING

Using an unconstrained basis matrix poses a series of problems. Basis vectors need to be analyzed and assigned to particular note prior to the analysis of the note activity matrix, which introduces additional errors to the process (compare Figure 4). However, because note events do not occur sparsely and independently and their spectra change greatly over time, using an unconstrained basis usually results in basis vectors that do not have the desired harmonic structure. Furthermore, results for an unconstrained basis are very different each time, and thus very difficult to compare and evaluate. That is why we firmly believe that a harmonic basis matrix with vectors containing harmonic structures strictly corresponding to notes (of, for instance, the diatonic scale) is a must when it comes to multipitch analysis. Analysis of the note activity matrix in this case is straightforward, as each row contains amplitudes of a single note.

Basis harmonicity can be achieved in three ways. We can either: use a fixed harmonic basis vectors, use a basis matrix pretrained on solo instrument data, or adapt the harmonic structure to the data. In the first approach we use an artificial harmonic spectra with peak amplitudes decreasing exponentially. It seems like an oversimplification, but, as we will see later, it gives very good results, especially when additional penalties are used, and the overfitting present in the other two methods is avoided. In the second approach, we use averaged note spectra obtained from the recordings of piano taken from the RWC database, which gives better results than the first method, but the performance drops slightly when different instrument is used.

In the third approach, we use an artificial harmonic basis from the first method and adapt it in such a matter that changes only the frequency peaks amplitudes, leaving the overall harmonic structure intact. This can be easily achieved without modifying the existing algorithm, because zero-valued elements of basis vectors remain at zero throughout the learning process due to its multiplicative nature. We could therefore initialize tthe basis to have zeros everywhere but at the positions of fundamentals of notes from a specific range of the 12-TET (Twelve-tone Equal Temperament) scale and at their harmonics, thus constraining the solution space to harmonic factorizations only.

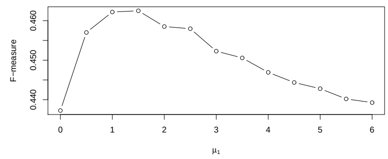
4. REGULARIZATIONS

The NNMA's penalty function can be extended to include additional penalties on both matrices.

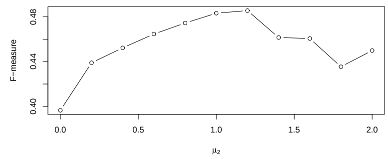
$$\nabla_{\mathbf{A}}(D_\varphi(\mathbf{X},\mathbf{AS})+\alpha(\mathbf{A}))\odot\mathbf{A}=\mathbf{0}, \tag{21}$$

$$\nabla_{\mathbf{S}}(D_\varphi(\mathbf{X},\mathbf{AS})+\beta(\mathbf{S}))\odot\mathbf{S}=\mathbf{0}, \tag{22}$$

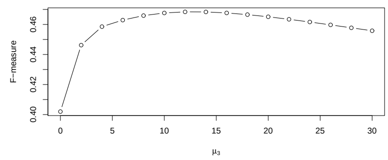
$$\mathbf{A}\leftarrow\mathbf{A}\odot\frac{(\mathbf{X}\odot\varphi''(\mathbf{AS}))\mathbf{S^T}}{(\mathbf{AS}\odot\varphi''(\mathbf{AS}))\mathbf{S^T}+\nabla_{\mathbf{A}}\alpha(\mathbf{A})}, \tag{23}$$



(a) Note activity sparsity, $r=1$



(b) Note activity decorrelation, $r=1$



(c) Temporal smoothness, $r=0$

Figure 2: Accuracy of multipitch analysis when using different additional objectives. A fixed harmonic basis was used.

$$\mathbf{S}\leftarrow\mathbf{S}\odot\frac{(\mathbf{X}\odot\varphi''(\mathbf{AS}))\mathbf{S^T}}{(\mathbf{AS}\odot\varphi''(\mathbf{AS}))\mathbf{S^T}+\nabla_{\mathbf{S}}\beta(\mathbf{S})}, \tag{24}$$

This allows the user to have greater impact on the resulting factorization. However, caution must be exercised when designing these additional penalty functions, as they might cause the solution to become negative or make the algorithm unstable. Though in our experience, using only nonnegative penalties (or rather their derivatives) usually led to a stable algorithm. Among the note activity matrix penalties used most successfully by us, are the sparseness penalty, the cross-correlation penalty and the time smoothness penalty.

To obtain sparse note activities we employ the l_p-norm with $p<2$:

$$\beta_1(\mathbf{S})=\mu_1|\mathbf{S}^p|, \tag{25}$$

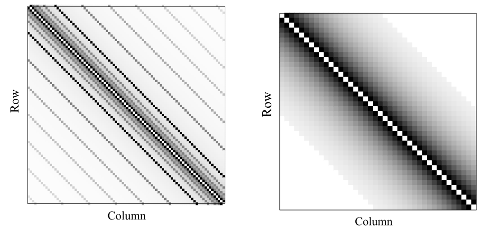
$$\nabla_{\mathbf{S}}\beta_1(\mathbf{S})=\mu_1 p\mathbf{S}^{p-1}, \tag{26}$$

and try to minimize it.

The cross-correlation penalty can be used to decrease the crosstalk between activities of different notes. The penalty function is defined as:

$$\beta_2(\mathbf{S})=\mu_2\sum_{i,j,k}W_{i,j}S_{i,k}S_{j,k}=\mu_2|\mathbf{W}\odot(\mathbf{SS}^T)|, \tag{27}$$

where $\mathbf{W}$ is a weighting matrix. In order to penalize only cross-correlation between different notes, we set $W_{i,i}=0$. Also, the weights will usually only depend on the interval



(a) (b)

Figure 3: Circulant weighting matrices: (a) a matrix that penalizes cross-correlation between activities of close notes and between notes in a common harmonic relation (octave 1:n, major third 5:4 and perfect fifth 3:2), (b) a matrix that encourages temporal smoothness; an exponential smoothness profile was used.

between the notes and the weighting matrix will become circulant. In this case we simply have:

$$\nabla_{\mathbf{S}}\beta_2(\mathbf{S}) = 2\mu_2\mathbf{WS} \tag{28}$$

By using this penalty we can also decrease the number of the most common pitch detection errors: octave errors (by increasing all weights $W_i \equiv 0 \pmod{12}$), major third errors (by increasing all $W_i = 4$) and perfect fifth errors (by increasing all $W_i = 7$). An example of a weighting matrix constructed in this manner is presented in Fig. 3a.

A very similar penalty can be used to encourage temporal smoothness in a way quite similar to the one presented in [12], but using less complicated penalty function.

$$\beta_3(\mathbf{S}) = -\mu_3 \sum_{i,j,k} V_{i,j}S_{k,i}S_{k,j} = -\mu_3|\mathbf{W} \odot (\mathbf{S}^T\mathbf{S})|, \tag{29}$$

where $\mathbf{V}$ is a weighting matrix. As with the note decorrelation penalty, using a circulant matrix with nullified main diagonal leads to a simple derivative:

$$\nabla_{\mathbf{S}}\beta_3(\mathbf{S}) = -2\mu_3\mathbf{SV}. \tag{30}$$

As mentioned before, using negative functions may lead to instability, so we used $\exp(-\mu_3\mathbf{SV})$ in place of 30. An example of weighting matrix $\mathbf{V}$ is depicted in Fig. 3b.

5. RESULTS

Different methods of evaluating multipitch analysis results have been proposed by researchers, but we would like to look at the problem in a somewhat narrower context of NNMA algorithms, and focus our attention only on the resulting note activity matrix. We therefore propose to directly compare this matrix with a ground truth matrix created from available MIDI data.

We have generalized precision (P) and recall (R) for real-valued data:

$$P = \left(\sum_{t,n} tp_{t,n}\right)\left(\sum_{t,n} tp_{t,n} + fp_{t,n}\right)^{-1}, \tag{31}$$

$$R = \left(\sum_{t,n} tp_{t,n}\right)\left(\sum_{t,n} tp_{t,n} + fn_{t,n}\right)^{-1}, \tag{32}$$

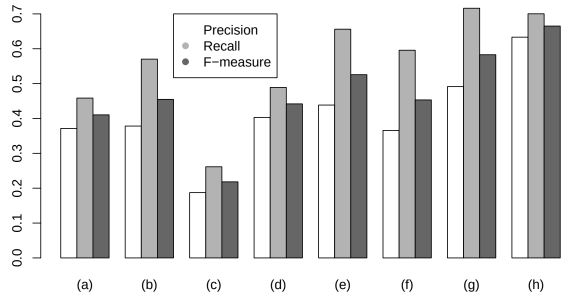


Figure 5: Results of multipitch analysis results obtained for different harmonic bases and NNMA variants. (a) Fixed harmonic basis, (b) basis pretrained on RWC piano recordings, (c) adaptive harmonic basis, (d) adaptive harmonic basis and additional penalties, (e) fixed harmonic basis and penalties, (f) pretrained basis and penalties, (g) average note activities of all methods, (h) averaged activities after simple filtering.

where tp is the *true positive*, fp is the *false positive* and fn is the *false negative*. They are defined as:

$$tp_{n,t} = \begin{cases} \mathbf{S}_{n,t}/\mathbf{G}_{n,t} & \text{if } \mathbf{G}_{n,t} \neq 0 \\ 0 & \text{if } \mathbf{G}_{n,t} = 0 \end{cases}, \tag{33}$$

$$fp_{n,t} = \begin{cases} \mathbf{S}_{n,t} & \text{if } \mathbf{G}_{n,t} = 0 \\ 0 & \text{if } \mathbf{G}_{n,t} \neq 0 \end{cases}, \tag{34}$$

$$fn_{n,t} = \begin{cases} \mathbf{G}_{n,t} - \mathbf{S}_{n,t}/\mathbf{G}_{n,t} & \text{if } \mathbf{G}_{n,t} \neq 0 \\ 0 & \text{if } \mathbf{G}_{n,t} = 0 \end{cases}, \tag{35}$$

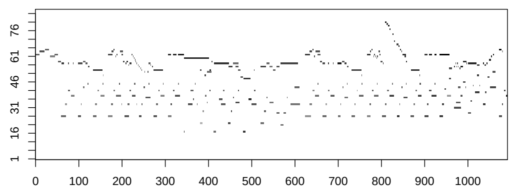
where n, t are the indices of elements from the note activity matrix corresponding to the note number and time, respectively.

All above definitions require that both the note activity matrix $\mathbf{S}$ and the ground truth matrix were normalized to the range $[0, 1]$. Intuitively, when we notice that $\sum_{t,n} tp_{t,n}$ is the amount of correctly identified note activity, $\sum_{t,n} tp_{t,n} + fp_{t,n}$ is the total detected note activity, and $\sum_{t,n} tp_{t,n} + fn_{t,n}$ is the amount of true note activity, it follows that the precision is a measure of how much of the detected note activity matches the ground truth data, and the recall is a measure of how well the note activity is detected. We can now use the standard F-measure definition:

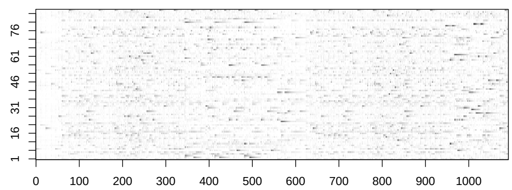
$$F = \frac{2PR}{P+R}. \tag{36}$$

Figure 2 depicts the F-measure of the note activity matrix for different values of μ_1, μ_2 and μ_3, respectively. In each case one coefficient was changed while the other were fixed. All results were obtained from piano recordings synthesized from RWC database's MIDI files. A gain in accuracy when using this penalties is evident and is present for most of experimental septups (different r values, different basis matrices, etc.). We have used piano music to get the results presented in this paper, but our experiments with different instruments did not show any significant loss in accuracy, even when we used a basis matrix pretrained on piano recordings. The music was synthesized from MIDI data recorded on an electric piano, taken mostly from the RWC database. The synthesis was performed using a realistic and accurate Steinway Model-C grand piano sound font[1].

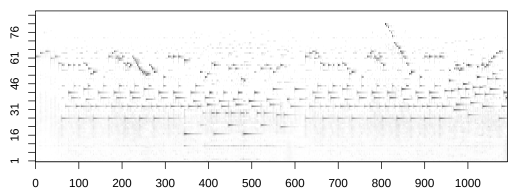
Figure 5 depicts results obtained for different variants of spectrogram factorizations. We had our greatest expectations



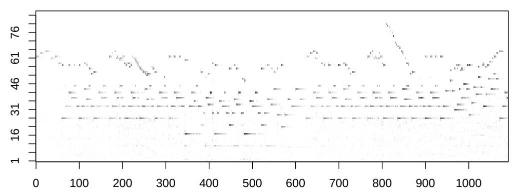
(a) MIDI data used to synthesize the input audio signal and to evalute the results.



(b) Results for an unconstrained and unpenalized NMF.



(c) Results for a harmonically constrained NMF.



(d) Results for a harmonically constrained and penalized NMF.

Figure 4: Original MIDI information and multiple pitch estimation results for different variants of spectrogram factorizations. All data obtained for Chopin's *Nocturne in B flat minor, Op. 9, No. 1*, sequenced on a MIDI piano by Nick Carter [5]. Horizontal axis correspond to time and the vertical axis to piano key number (exluding subfig. 4b, for which there is no correspondance between basis vector number and piano key).

in the adaptive harmonic basic approach. Surprisingly however, the adaptive harmonic basis gave the poorest results if the additional penalties were not used (Fig. 5c), and even with these penalties (Fig. 5d), the penalized fixed harmonic approach (Fig. 5e) gave slightly better results. As we mentioned before, this might be a result of overfitting. Combining all 3 approaches by taking a geometric mean of all 3 note activity matrices (geometric mean was chosen in order to boost the values that were large in all approaches) gave a further 5% boost in the F-measure (Fig. 5g).

6. CONCLUSION

We have presented many new ways in which the Nonnegative Matrix Factorization can be extended to better fit the task of multipitch analysis. By doing this, we have also given the method's user many parameters that can be fine tuned to fit specific problems, which is a big advantage over the generic NMF. We have also proposed an objective method of evaluating the results of using NMF directly, without having to perform the postanalysis note detection part. The obtained results were better than those of any other NMF-based approach we have tested and can be further improved by postprocessing the note activities.

REFERENCES

[1] Steinway model-c grand piano soundfont, December 2008.

[2] Mirex 2008 multiple fundamental frequency estimation and tracking results, February 2009.

[3] S. Abdallah and M. Plumbley. Polyphonic Music Transcription by Non-negative Sparse Coding of Power Spectra. *Proc. 5th International Conference on Music Information Retrieval*, 510:318–325, 2004.

[4] S. Abdallah and M. Plumbley. Unsupervised Analysis of Polyphonic Music by Sparse Coding. *IEEE Trans. Neural Netw.*, 17(1):179–196, 2006.

[5] N. Carter. The ultimate chopin midi collection, February 2009.

[6] I. Dhillon and S. Sra. Generalized Nonnegative Matrix Approximations with Bregman Divergences. *Advances in Neural Information Processing Systems*, 18:283, 2006.

[7] D. Lee and H. Seung. Algorithms for Non-negative Matrix Factorization. *Advances in Neural Information Processing Systems*, pages 556–562, 2001.

[8] P. Paatero and U. Tapper. Positive Matrix Factorization: A Non-negative Factor Model with Optimal Utilization of Error Estimates of Data Values. *Environmetrics*, 5(2):111–126, 1994.

[9] M. Schmidt and M. Mørup. Sparse Non-negative Matrix Factor 2-D Deconvolution for Automatic Transcription of Polyphonic Music. In *Proc. 6th International Symposium on Independent Component Analysis and Blind Signal Separation*, 2006.

[10] P. Smaragdis and J. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, pages 177–180, 2003.

[11] S. Sra and I. Dhillon. Nonnegative matrix approximation: Algorithms and applications. Technical report, Technical report, Dept. of Computer Sciences, The Univ. of Texas at Austin, 2006.

[12] K. Wilson, B. Raj, and P. Smaragdis. Regularized Nonnegative Matrix Factorization with Temporal Dependencies for Speech Denoising.