Available online at www.sciencedirect.com

ScienceDirect

Computational Statistics & Data Analysis 52 (2007) 155 – 173

COMPUTATIONAL STATISTICS & DATA ANALYSIS

www.elsevier.com/locate/csda

Algorithms and applications for approximate nonnegative matrix factorization

Michael W. Berry[a,*], Murray Browne[a], Amy N. Langville[b,1], V. Paul Pauca[c,2], Robert J. Plemmons[c,2]

[a] *Department of Computer Science, University of Tennessee, Knoxville, TN 37996-3450, USA*
[b] *Department of Mathematics, College of Charleston, Charleston, SC 29424-0001, USA*
[c] *Departments of Computer Science and Mathematics, Wake Forest University, Winston-Salem, NC 27109, USA*

Available online 29 November 2006

Abstract

The development and use of low-rank approximate nonnegative matrix factorization (NMF) algorithms for feature extraction and identification in the fields of text mining and spectral data analysis are presented. The evolution and convergence properties of hybrid methods based on both sparsity and smoothness constraints for the resulting nonnegative matrix factors are discussed. The interpretability of NMF outputs in specific contexts are provided along with opportunities for future work in the modification of NMF algorithms for large-scale and time-varying data sets.

Keywords: Nonnegative matrix factorization; Text mining; Spectral data analysis; Email surveillance; Conjugate gradient; Constrained least squares

1. Introduction

Recent technological developments in sensor technology and computer hardware have resulted in increasing quantities of data, rapidly overwhelming many of the classical data analysis tools available. Processing these large amounts of data has created new concerns with respect to data representation, disambiguation, and dimensionality reduction. Because information gathering devices have only finite bandwidth, the collected data are not often exact. For example, signals received by antenna arrays often are contaminated by noise and other degradations. Before useful deductive science can be applied, it is often important to first reconstruct or represent the data so that the inexactness is reduced while certain feasibility conditions are satisfied.

Secondly, in many situations the data observed from complex phenomena represent the integrated result of several interrelated variables acting together. When these variables are less precisely defined, the actual information contained

[*] Corresponding author. Tel.: +1 865 974 3838; fax: +1 865 974 4404.

E-mail addresses: berry@cs.utk.edu (M.W. Berry), mbrowne@cs.utk.edu (M. Browne), langvillea@cofc.edu (A.N. Langville), paucavp@wfu.edu (V.P. Pauca), plemmons@wfu.edu (R.J. Plemmons).

[1] Research supported in part by the National Science Foundation under Grant CAREER-CCF-0546622.

[2] Research supported in part by the Air Force Office of Scientific Research under Grant FA49620-03-1-0215, the Army Research Office under Grants DAAD19-00-1-0540 and W911NF-05-1-0402, and by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Grant DAAD19-01-2-0008.

doi:10.1016/j.csda.2006.11.006

in the original data might be overlapping and ambiguous. A reduced system model could provide a fidelity near the level of the original system. One common ground in the various approaches for noise removal, model reduction, feasibility reconstruction, and so on, is to replace the original data by a lower dimensional representation obtained via subspace approximation. The use of low-rank approximations, therefore, comes to the forefront in a wide range of important applications. Factor analysis and principal component analysis are two of the many classical methods used to accomplish the goal of reducing the number of variables and detecting structures among the variables.

Often the data to be analyzed is nonnegative, and the low-rank data are further required to be comprised of nonnegative values in order to avoid contradicting physical realities. Classical tools cannot guarantee to maintain the nonnegativity. The approach of finding reduced rank nonnegative factors to approximate a given nonnegative data matrix thus becomes a natural choice. This is the so-called nonnegative matrix factorization (NMF) problem which can be stated in generic form as follows:

NMF problem. *Given a nonnegative matrix* $\mathbf{A} \in R^{m\times n}$ *and a positive integer* $k < \min\{m, n\}$, *find nonnegative matrices* $\mathbf{W} \in R^{m\times k}$ *and* $\mathbf{H} \in R^{k\times n}$ *to minimize the functional*

$$f(\mathbf{W}, \mathbf{H}) = \tfrac{1}{2}\|\mathbf{A} - \mathbf{W}\mathbf{H}\|_{\mathrm{F}}^2. \tag{1}$$

The product **WH** is called a NMF of **A**, although **A** is not necessarily equal to the product **WH**. Clearly the product **WH** is an approximate factorization of rank at most k, but we will omit the word “approximate” in the remainder of this paper. An appropriate decision on the value of k is critical in practice, but the choice of k is very often problem dependent. In most cases, however, k is usually chosen such that $k \ll \min(m, n)$ in which case **WH** can be thought of as a compressed form of the data in **A**.

Another key characteristic of NMF is the ability of numerical methods that minimize (1) to extract underlying features as basis vectors in **W**, which can then be subsequently used for identification and classification. By not allowing negative entries in **W** and **H**, NMF enables a non-subtractive combination of parts to form a whole (Lee and Seung, 1999). Features may be parts of faces in image data, topics or clusters in textual data, or specific absorption characteristics in hyperspectral data. In this paper, we discuss the enhancement of NMF algorithms for the primary goal of feature extraction and identification in text and spectral data mining.

Important challenges affecting the numerical minimization of (1) include the existence of local minima due to the non-convexity of $f(\mathbf{W}, \mathbf{H})$ in both **W** and **H**, and perhaps more importantly the lack of a unique solution which can be easily seen by considering $\mathbf{W}\mathbf{D}\mathbf{D}^{-1}\mathbf{H}$ for any nonnegative invertible matrix **D** whose inverse, $\mathbf{D}^{-1}$, is also nonnegative. These and other convergence related issues are dealt with in Section 3. Still, NMF is quite appealing for data mining applications since, in practice, even local minima can provide desirable properties such as data compression and feature extraction as previously explained.

The remainder of this paper is organized as follows. In Section 2 we give a brief description of numerical approaches for the solution of the NMF problem. Fundamental NMF algorithms and their convergence properties are discussed in Section 3. The use of constraints or penalty terms to augment solutions is discussed in Section 4 and applications of NMF algorithms in the fields of text mining and spectral data analysis are highlighted in Section 5. The need for further research in NMF algorithms concludes the paper in Section 6.

2. Numerical approaches for NMF

The 1999 article in *Nature* by Daniel Lee and Sebastian Seung (Lee and Seung, 1999) started a flurry of research into the new NMF. Hundreds of papers have cited Lee and Seung, but prior to its publication several lesser known papers by Pentti Paatero (Paatero and Tapper, 1994; Paatero, 1997, 1999) actually deserve more credit for the factorization’s historical development. Though Lee and Seung cite Paatero’s 1997 paper on his so-called positive matrix factorization in their *Nature* article, Paatero’s work is rarely cited by subsequent authors. This is partially due to Paatero’s unfortunate phrasing of positive matrix factorization, which is misleading as Paatero’s algorithms create a NMF. Moreover, Paatero actually published his initial factorization algorithms years earlier in Paatero and Tapper (1994).

Since the introduction of the NMF problem by Lee and Seung, a great deal of published and unpublished work has been devoted to the analysis, extension, and application of NMF algorithms in science, engineering and medicine. The NMF problem has been cast into alternate formulations by various authors. Lee and Seung (2001) provided an information theoretic formulation based on the Kullback–Leibler divergence of **A** from **WH** that, in turn, lead to various related approaches. For example, Cichocki et al. (2006) have proposed cost functions based on Csiszár's φ-divergence. Wang et al. (2004) propose a formulation that enforces constraints based on Fisher linear discriminant analysis for improved determination of spatially localized features. Guillamet et al. (2001) have suggested the use of a diagonal weight matrix **Q** in a new factorization model, $\mathbf{AQ} \approx \mathbf{WHQ}$, in an attempt to compensate for feature redundancy in the columns of **W**. This problem can also be alleviated using column stochastic constraints on **H** (Pauca et al., 2006b). Other approaches that propose alternative cost function formulations include but are not limited to Hamza and Brady (2006), Dhillon and Sra (2005). A theoretical analysis of NMF of symmetric matrices can be found in Catral et al. (2004).

Various alternative minimization strategies for the solution of (1) have also been proposed in an effort to speed up convergence of the standard NMF iterative algorithm of Lee and Seung. Lin (2005b) has recently proposed the use of a projected gradient bound-constrained optimization method that is computationally competitive and appears to have better convergence properties than the standard (multiplicative update rule) approach. Use of certain auxiliary constraints in (1) may, however, break down the bound-constrained optimization assumption, limiting the applicability of projected gradient methods. Gonzalez and Zhang (2005) proposed accelerating the standard approach based on an interior-point gradient method. Zdunek and Cichocki (2006) proposed a quasi-Newton optimization approach for updating **W** and **H** where negative values are replaced with small $\varepsilon > 0$ to enforce nonnegativity, at the expense of a significant increase in computation time per iteration. Further studies related to convergence of the standard NMF algorithm can be found in Chu et al. (2004), Lin (2005a), Salakhutdinov et al. (2003) among others.

In the standard NMF algorithm **W** and **H** are initialized with random nonnegative values, before the iteration starts. Various efforts have focused on alternate approaches for initializing or seeding the algorithm in order to speed up or otherwise influence convergence to a desired solution. Wild et al. (2003) and Wild (2002), for example, employed a spherical k-means clustering approach to initialize **W**. Boutsidis and Gallopoulos (2005) use an SVD-based initialization and show anecdotical examples of speed up in the reduction of the cost function. Effective initialization remains, however, an open problem that deserves further attention.

Recently, various authors have proposed extending the NMF problem formulation to include additional auxiliary constraints on **W** and/or **H**. For example smoothness constraints have been used to regularize the computation of spectral features in remote sensing data (Piper et al., 2004; Pauca et al., 2006a). Chen and Cichocki (2005) employed temporal smoothness and spatial correlation constraints to improve the analysis of EEG data for early detection of Alzheimer's disease. Hoyer (2002, 2004) employed sparsity constraints on either **W** or **H** to improve local rather than global representation of data. The extension of NMF to include such auxiliary constraints is problem dependent and often reflects the need to compensate for the presence of noise or other data degradations in **A**.

3. Fundamental algorithms

In this section, we provide a basic classification scheme that encompasses many of the NMF algorithms previously mentioned. Although such algorithms can straddle more than one class, in general they can be divided into three general classes: multiplicative update algorithms, gradient descent algorithms, and alternating least squares (ALS) algorithms. We note that Cichocki and Zdunek (2006) have recently created an entire library (NMFLAB) of MATLAB® routines for each class of the NMF algorithms.

3.1. Multiplicative update algorithms

The prototypical multiplicative algorithm originated with Lee and Seung (2001). Their multiplicative update algorithm with the mean squared error objective function (using MATLAB array operator notation) is provided below.

```
        MULTIPLICATIVE UPDATE ALGORITHM FOR NMF
$\mathbf{W} = \mathrm{rand}(m, k);$     % initialize $\mathbf{W}$ as random dense matrix
$\mathbf{H} = \mathrm{rand}(k, n);$     % initialize $\mathbf{H}$ as random dense matrix
for i = 1 : maxiter
  (MU)   $\mathbf{H} = \mathbf{H} .* (\mathbf{W}^{\mathrm{T}}\mathbf{A})./(\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H} + 10^{-9});$
  (MU)   $\mathbf{W} = \mathbf{W}. * (\mathbf{A}\mathbf{H}^{\mathrm{T}})./(\mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}} + 10^{-9});$
end
```

The 10^{-9} in each update rule is added to avoid division by zero. Lee and Seung used the gradient and properties of continual descent (more precisely, continual non-increase) to claim that the above algorithm converges to a local minimum, which was later shown to be incorrect (Chu et al., 2004; Finesso and Spreij, 2004; Gonzalez and Zhang, 2005; Lin, 2005b). In fact, the proof by Lee and Seung merely shows a continual descent property, which does not preclude descent to a saddle point. To understand why, one must consider two basic observations involving the Karush–Kuhn–Tucker optimality conditions.

First, if the initial matrices $\mathbf{W}$ and $\mathbf{H}$ are strictly positive, then these matrices remain positive throughout the iterations. This statement is easily verified by referring to the multiplicative form of the update rules. Second, if the sequence of iterates $(\mathbf{W}, \mathbf{H})$ converge to $(\mathbf{W}^*, \mathbf{H}^*)$ and $\mathbf{W}^* > \mathbf{0}$ and $\mathbf{H}^* > \mathbf{0}$, then $(\partial f/\partial \mathbf{W})(\mathbf{W}^*, \mathbf{H}^*) = \mathbf{0}$ and $(\partial f/\partial \mathbf{H})(\mathbf{W}^*, \mathbf{H}^*) = \mathbf{0}$. This second point can be verified for $\mathbf{H}$ by using the additive form of the update rule

$$\mathbf{H} = \mathbf{H} + [\mathbf{H}./(\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H})]. * [\mathbf{W}^{\mathrm{T}}(\mathbf{A} - \mathbf{W}\mathbf{H})]. \tag{2}$$

Consider the (i, j)-element of $\mathbf{H}$. Suppose a limit point for $\mathbf{H}$ has been reached such that $\mathbf{H}_{ij} > 0$. Then from Eq. (2), we know

$$\frac{\mathbf{H}_{ij}}{[\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H}]_{ij}}([\mathbf{W}^{\mathrm{T}}\mathbf{A}]_{ij} - [\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H}]_{ij}) = 0.$$

Since $\mathbf{H}_{ij} > 0$, this implies $[\mathbf{W}^{\mathrm{T}}\mathbf{A}]_{ij} = [\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H}]_{ij}$, which implies $[\partial f/\partial \mathbf{H}]_{ij} = 0$.

While these two points combine to satisfy the Karush–Kuhn–Tucker optimality conditions below (Bertsekas, 1999), this holds only for limit points $(\mathbf{W}^*, \mathbf{H}^*)$ that do not have any elements equal to 0.

$$\begin{aligned}
&\mathbf{W} \geqslant \mathbf{0},\\
&\mathbf{H} \geqslant \mathbf{0},\\
&(\mathbf{W}\mathbf{H} - \mathbf{A})\mathbf{H}^{\mathrm{T}} \geqslant \mathbf{0},\\
&\mathbf{W}^{\mathrm{T}}(\mathbf{W}\mathbf{H} - \mathbf{A}) \geqslant \mathbf{0},\\
&(\mathbf{W}\mathbf{H} - \mathbf{A})\mathbf{H}^{\mathrm{T}}. * \mathbf{W} = \mathbf{0} \quad \text{and}\\
&\mathbf{W}^{\mathrm{T}}(\mathbf{W}\mathbf{H} - \mathbf{A}). * \mathbf{H} = \mathbf{0}.
\end{aligned}$$

Despite the fact that, for example, $\mathbf{H}_{ij} > 0$ for all iterations, this element could be converging to a limit value of 0 with $[\partial f/\partial \mathbf{H}]_{ij} \geqslant 0$ at $(\mathbf{W}^*, \mathbf{H}^*)$ even when $\mathbf{H}^*_{ij} = 0$. Thus, it is possible that $\mathbf{H}^*_{ij} = 0$, in which case one must prove the corresponding complementary slackness condition that $(\partial f/\partial \mathbf{H})(\mathbf{W}^*, \mathbf{H}^*) \geqslant \mathbf{0}$, and it is not apparent how to use the multiplicative update rules to do this. Thus, in summary, we can only make the following statement about the convergence of the Lee and Seung multiplicative update algorithms: *When the algorithm has converged to a limit point in the interior of the feasible region, this point is a stationary point. This stationary point may or may not be a local minimum. When the limit point lies on the boundary of the feasible region, its stationarity cannot be determined.*

Due to their status as the first well-known NMF algorithms, the Lee and Seung multiplicative update algorithms have become a baseline against which the newer algorithms are compared. It has been repeatedly shown that the Lee and Seung algorithms, when they converge (which is often in practice), are notoriously slow to converge. They require many more iterations than alternatives such as the gradient descent and ALS algorithms discussed below, and the work per iteration is high since each iteration requires $O(mnk)$ work. Nevertheless, clever implementations can improve the situation. For example, in the update rule for **W**, which requires the product $\mathbf{WHH}^{\mathrm{T}}$, the small $k \times k$ product $\mathbf{HH}^{\mathrm{T}}$ should be created first.

In order to overcome some of these shortcomings, researchers have proposed modifications to the original Lee and Seung algorithms. For example, Gonzalez and Zhang (2005) created a modification that accelerates the Lee and Seung algorithm, but unfortunately, still has the same convergence problems. Recently, Lin created a modification that resolves one of the convergence issues. Namely, Lin's modified algorithm is guaranteed to converge to a stationary point (Lin, 2005a). However, this algorithm requires slightly more work per iteration than the already slow Lee and Seung algorithm. In addition, Dhillon and Sra derive multiplicative update rules that incorporate weights for the importance of elements in the approximation (Dhillon and Sra, 2005).

3.2. Gradient descent algorithms

NMF algorithms of the second class are based on gradient descent methods. We have already mentioned the fact that the above multiplicative update algorithm can be considered a gradient descent method (Chu et al., 2004; Lee and Seung, 2001). Algorithms of this class repeatedly apply update rules of the form shown below.

```
BASIC GRADIENT DESCENT ALGORITHM FOR NMF
$\mathbf{W} = \mathrm{rand}(\mathrm{m}, \mathrm{k});$   % initialize $\mathbf{W}$
$\mathbf{H} = \mathrm{rand}(\mathrm{k}, \mathrm{n});$   % initialize $\mathbf{H}$
for $i = 1$ : maxiter
    $\mathbf{H} = \mathbf{H} - \varepsilon_H \frac{\partial f}{\partial \mathbf{H}}$
    $\mathbf{W} = \mathbf{W} - \varepsilon_W \frac{\partial f}{\partial \mathbf{W}}$
end
```

The step size parameters ε_H and ε_W vary depending on the algorithm, and the partial derivatives are the same as those shown in Section 3.1. These algorithms always take a step in the direction of the negative gradient, the direction of steepest descent. The trick comes in choosing the values for the stepsizes ε_H and ε_W. Some algorithms initially set these stepsize values to 1, then multiply them by one-half at each subsequent iteration (Hoyer, 2004). This is simple, but not ideal because there is no restriction that keeps elements of the updated matrices **W** and **H** from becoming negative. A common practice employed by many gradient descent algorithms is a simple projection step (Shahnaz et al., 2006; Hoyer, 2004; Chu et al., 2004; Pauca et al., 2006a). That is, after each update rule, the updated matrices are projected to the nonnegative orthant by setting all negative elements to the nearest nonnegative value, 0.

Without a careful choice for ε_H and ε_W, little can be said about the convergence of gradient descent methods. Further, adding the nonnegativity projection makes analysis even more difficult. Gradient descent methods that use a simple geometric rule for the stepsize, such as powering a fraction or scaling by a fraction at each iteration, often produce a poor factorization. In this case, the method is very sensitive to the initialization of **W** and **H**. With a random initialization, these methods converge to a factorization that is not very far from the initial matrices. Gradient descent methods, such as the Lee and Seung algorithms, that use a smarter choice for the stepsize produce a better factorization, but as mentioned above, are very slow to converge (if at all). As discussed in Chu et al. (2004), the Shepherd method is a proposed gradient descent technique that can accelerate convergence using wise choices for the stepsize. Unfortunately, the convergence theory to support this approach is somewhat lacking.

3.3. ALS algorithms

The last class of NMF algorithms is the ALS class. In these algorithms, a least squares step is followed by another least squares step in an alternating fashion, thus giving rise to the ALS name. ALS algorithms were first used by Paatero (Paatero and Tapper, 1994). ALS algorithms exploit the fact that, while the optimization problem of Eq. (1) is not convex in both **W** and **H**, it is convex in either **W** or **H**. Thus, given one matrix, the other matrix can be found with a simple least squares computation. An elementary ALS algorithm follows.

```
                    BASIC ALS ALGORITHM FOR NMF
$\mathbf{W} = \mathrm{rand}(m, k)$;   % initialize $\mathbf{W}$ as random dense matrix or use another
            initialization from Langville et al. (2006)
for i = 1: maxiter
  (LS)      Solve for $\mathbf{H}$ in matrix equation $\mathbf{W}^T\mathbf{W}\mathbf{H} = \mathbf{W}^T\mathbf{A}$.
  (NONNEG)  Set all negative elements in $\mathbf{H}$ to 0.
  (LS)      Solve for $\mathbf{W}$ in matrix equation $\mathbf{H}\mathbf{H}^T\mathbf{W}^T = \mathbf{H}\mathbf{A}^T$.
  (NONNEG)  Set all negative elements in $\mathbf{W}$ to 0.
end
```

In the above pseudocode, we have included the simplest method for insuring nonnegativity, the projection step, which sets all negative elements resulting from the least squares computation to 0. This simple technique also has a few added benefits. Of course, it aids sparsity. Moreover, it allows the iterates some additional flexibility not available in other algorithms, especially those of the multiplicative update class. One drawback of the multiplicative algorithms is that once an element in **W** or **H** becomes 0, it must remain 0. This locking of 0 elements is restrictive, meaning that once the algorithm starts heading down a path towards a fixed point, even if it is a poor fixed point, it must continue in that vein. The ALS algorithms are more flexible, allowing the iterative process to escape from a poor path.

Depending on the implementation, ALS algorithms can be very fast. The implementation shown above requires significantly less work than other NMF algorithms and slightly less work than an SVD implementation. Improvements to the basic ALS algorithm appear in Paatero (1999), Langville et al. (2006). Most improvements incorporate sparsity and nonnegativity constraints such as those described in Section 4.

We conclude this section with a discussion of the convergence of ALS algorithms. Algorithms following an alternating process, approximating **W**, then **H**, and so on, are actually variants of a simple optimization technique that has been used for decades, and is known under various names such as alternating variables, coordinate search, or the method of local variation (Nocedal and Wright, 1999). While statements about global convergence in the most general cases have not been proven for the method of alternating variables, a bit has been said about certain special cases (Berman, 1969; Cea, 1971; Polak, 1971; Powell, 1964; Torczon, 1997; Zangwill, 1967). For instance, Polak (1971) proved that every limit point of a sequence of alternating variable iterates is a stationary point. Others (Powell, 1964, 1973; Zangwill, 1967) prove convergence for special classes of objective functions, such as convex quadratic functions. Furthermore, it is known that an ALS algorithm that properly enforces nonnegativity, for example, through the nonnegative least squares (NNLS) algorithm of Lawson and Hanson (1995), will converge to a local minimum (Bertsekas, 1999; Grippo and Sciandrone, 2000; Lin, 2005b). Unfortunately, solving nonnegatively constrained least squares problems rather than unconstrained least squares problems at each iteration, while guaranteeing convergence to a local minimum, greatly increases the cost per iteration. So much so that even the fastest NNLS algorithm of Bro and de Jong (1997) increases the work by a few orders of magnitude. In practice, researchers settle for the speed offered by the simple projection to the nonnegative orthant, sacrificing convergence theory. Nevertheless, this tradeoff seems warranted. Some experiments show that saddle point solutions can give reasonable results in the context of the problem, a finding confirmed by experiments with ALS-type algorithms in other contexts (de Leeuw et al., 1976; Gill et al., 1981; Smilde et al., 2004; Wold, 1966, 1975).

3.4. General convergence comments

In general, for an NMF algorithm of any class, one should input the fixed point solution into optimality conditions (Chu et al., 2004; Gonzalez and Zhang, 2005) to determine if it is indeed a minimum. If the solution passes the optimality conditions, then it is at least a local minimum. In fact, the NMF problem does not have a unique global minimum. Consider that a minimum solution given by the matrices **W** and **H** can also be given by an infinite number of equally good solution pairs such as **WD** and $\mathbf{D}^{-1}\mathbf{H}$ for any nonnegative **D** and $\mathbf{D}^{-1}$. Since scaling and permutation cause uniqueness problems, some algorithms enforce row or column normalizations at each iteration to alleviate these. If, in a particular application, it is imperative that an excellent local minimum be found, we suggest running an NMF algorithm with several different initializations using a Monte Carlo type approach.

Of course, it would be advantageous to know the rate of convergence of these algorithms. Proving rates of convergence for these algorithms is an open research problem. It may be possible under certain conditions to make claims about the rates of convergence of select algorithms, or least relative rates of convergence between various algorithms.

A related goal is to obtain bounds on the quality of the fixed point solutions (stationary points and local minimums). Ideally, because the rank-k SVD, denoted by $\mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k$, provides a convenient baseline, we would like to show something of the form

$$1-\varepsilon \leqslant \frac{\|\mathbf{A}-\mathbf{W}_k\mathbf{H}_k\|}{\|\mathbf{A}-\mathbf{U}_k\Sigma_k\mathbf{V}_k\|} \leqslant 1,$$

where ε is some small positive constant that depends on the parameters of the particular NMF algorithm. Such statements were made for a similar decomposition, the **CUR** decomposition of Drineas et al. (2006).

The natural convergence criterion, $\|\mathbf{A}-\mathbf{WH}\|_{\mathrm{F}}$, incurs an expense, which can be decreased with careful implementation. The following alternative expression

$$\|\mathbf{A}-\mathbf{WH}\|_{\mathrm{F}}^2 = \mathrm{trace}(\mathbf{A}^{\mathrm{T}}\mathbf{A}) - 2\,\mathrm{trace}(\mathbf{H}^{\mathrm{T}}(\mathbf{W}^{\mathrm{T}}\mathbf{A})) + \mathrm{trace}(\mathbf{H}^{\mathrm{T}}(\mathbf{W}^{\mathrm{T}}\mathbf{WH}))$$

contains an efficient order of matrix multiplication and also allows the expense associated with trace$(\mathbf{A}^{\mathrm{T}}\mathbf{A})$ to be computed only once and used thereafter. Further, $\mathbf{W}^{\mathrm{T}}\mathbf{A}$ and $\mathbf{W}^{\mathrm{T}}\mathbf{WH}$ are required by most NMF algorithms and thus need not be recomputed. As a result, the cost of this residual computation can be reduced to $\mathrm{O}(nk)$. Nearly all NMF algorithm implementations use a maximum number of iterations as secondary stopping criteria (including the NMF algorithms presented in this paper). However, a fixed number of iterations is not a mathematically appealing way to control the number of iterations executed because the most appropriate value for *maxiter* is problem-dependent. The first paper to mention this convergence criterion problem is Lin (2005b), which includes alternatives, experiments, and comparisons. Another alternative is also suggested in Langville et al. (2006).

4. Application-dependent auxiliary constraints

As previously explained, the NMF problem formulation given in Section 1 is sometimes extended to include auxiliary constraints on **W** and/or **H**. This is often done to compensate for uncertainties in the data, to enforce desired characteristics in the computed solution, or to impose prior knowledge about the application at hand. Penalty terms are typically used to enforce auxiliary constraints, extending the cost function of Eq. (1) as follows:

$$f(\mathbf{W},\mathbf{H}) = \|\mathbf{A}-\mathbf{WH}\|_{\mathrm{F}}^2 + \alpha J_1(\mathbf{W}) + \beta J_2(\mathbf{H}). \tag{3}$$

Here $J_1(\mathbf{W})$ and $J_2(\mathbf{H})$ are the penalty terms introduced to enforce certain application-dependent constraints, and α and β are small regularization parameters that balance the trade-off between the approximation error and the constraints.

Smoothness constraints are often enforced to regularize the computed solutions in the presence of noise in the data. For example the term,

$$J_1(\mathbf{W}) = \|\mathbf{W}\|_{\mathrm{F}}^2 \tag{4}$$

penalizes **W** solutions of large Frobenius norm. Notice that this term is implicitly penalizing the columns of **W** since $\|\mathbf{W}\|_{\mathrm{F}}^2 = \sum_i \|\mathbf{w}_i\|_2^2$. In practice, the columns of **W** are often normalized to add up to one in order to maintain **W** away from zero. This form of regularization is known as *Tikhonov regularization* in the inverse problems community. More generally, one can rewrite (4) as $J_1(\mathbf{W}) = \|\mathbf{LW}\|_{\mathrm{F}}^2$, where **L** is a regularization operator. Other choices than the identity for **L** include Laplacian operators. Smoothness constraints can be applied likewise to **H**, depending on the application needs. For example, Chen and Cichocki (2005) enforce temporal smoothness in the columns of **H** by defining

$$J_2(\mathbf{H}) = \frac{1}{n}\sum_i \|(\mathbf{I}-\mathbf{T})\mathbf{h}_i^{\mathrm{T}}\|_2^2 = \frac{1}{n}\|(\mathbf{I}-\mathbf{T})\mathbf{H}^{\mathrm{T}}\|_{\mathrm{F}}^2, \tag{5}$$

where n is the total number of columns in the data matrix **A** and **T** is an appropriately defined convolution operator. The effectiveness of constraints of the form (4) is demonstrated in Section 5, where it is shown that features of higher quality can be obtained than with NMF alone.

Sparsity constraints on either **W** or **H** can be similarly imposed. The notion of sparsity refers sometimes to a representational scheme where only a few features are effectively used to represent data vectors (Hoyer, 2002, 2004). It also appears to refer at times to the extraction of local rather than global features, the typical example being local facial features extracted from the CBCL and ORL face image databases (Hoyer, 2004). Measures for sparsity include, for example, the ℓ^p norms for $0 < p \leqslant 1$ (Karvanen and Cichocki, 2003) and Hoyer's measure,

$$\text{sparseness}(\mathbf{x}) = \frac{\sqrt{n} - \|\mathbf{x}\|_1/\|\mathbf{x}\|_2}{\sqrt{n} - 1}.$$

The latter can be imposed as a penalty term of the form

$$J_2(\mathbf{H}) = (\omega\|\text{vec}(\mathbf{H})\|_2 - \|\text{vec}(\mathbf{H})\|_1)^2, \tag{6}$$

where $\omega = \sqrt{kn} - (\sqrt{kn} - 1)\gamma$ and $\text{vec}(\cdot)$ is the vec operator that transforms a matrix into a vector by stacking its columns. The desired sparseness in **H** is specified by setting γ to a value between 0 and 1.

In certain applications such as hyperspectral imaging, a solution pair (**W**, **H**) must comply with constraints that make it physically realizable (Keshava, 2003). One such physical constraint requires mixing coefficients h_{ij} to sum to one, i.e., $\sum_i h_{ij} = 1$ for all j. Enforcing such a physical constraint can significantly improve the determination of inherent features (Pauca et al., 2006b), when the data are in fact linear combinations of these features. Imposing additivity to one in the columns of **H** can be written as a penalty term in the form

$$J_2(\mathbf{H}) = \|\mathbf{H}^{\mathrm{T}}\mathbf{e}_1 - \mathbf{e}_2\|_2^2, \tag{7}$$

where $\mathbf{e}_1$ and $\mathbf{e}_2$ are vectors with all entries equal to 1. This is the same as requiring that **H** be column stochastic (Berman and Plemmons, 1994) or alternatively that the minimization of (3) seek solutions **W** whose columns form a convex set containing the data vectors in **A**. Notice, however, that full additivity is often not achieved since $\mathbf{H}^{\mathrm{T}}\mathbf{e}_1 \approx \mathbf{e}_2$ depending on the value of the regularization parameter β.

Of course, the multiplicative update rules for **W** and **H** in the alternating gradient descend mechanism of Lee and Seung change when the extended cost function (3) is minimized. In general assuming that $J_1(\mathbf{W})$ and $J_2(\mathbf{H})$ have partial derivatives with respect to w_{ij} and h_{ij}, respectively, the update rules can be formulated as

$$\mathbf{W}_{ij}^{(t)} = \mathbf{W}_{ij}^{(t-1)} \cdot \frac{(\mathbf{A}\mathbf{H}^{\mathrm{T}})_{ij}}{(\mathbf{W}^{(t-1)}\mathbf{H}\mathbf{H}^{\mathrm{T}})_{ij} + \alpha(\partial J_1(\mathbf{W})/\partial w_{ij})}, \tag{8}$$

$$\mathbf{H}_{ij}^{(t)} = \mathbf{H}_{ij}^{(t-1)} \cdot \frac{(\mathbf{W}^{\mathrm{T}}\mathbf{A})_{ij}}{(\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H}^{(t-1)})_{ij} + \beta(\partial J_2(\mathbf{H})/\partial h_{ij})}. \tag{9}$$

The extended cost function is non-increasing with these update rules for sufficiently small values of α and β (Chen and Cichocki, 2005; Pauca et al., 2006b). Algorithms employing these update rules belong to the multiplicative update class described in Section 3.1 and have similar convergence issues.

In Section 5, we apply NMF with the extended cost function (3) and with smoothness constraints as in (4) to applications in the fields of text mining and spectral data analysis. The algorithm, denoted CNMF (Pauca et al., 2006a) is specified below for completeness.

```
                         CNMF
$\mathbf{W}$ = rand(m, k);   % initialize $\mathbf{W}$ as random dense matrix or use another
                     initialization from Langville et al. (2006)
$\mathbf{H}$ = rand(k, n);   % initialize $\mathbf{H}$ as random dense matrix or use another
initialization from Langville et al. (2006)
for i = 1 : maxiter
    (MU)  $\mathbf{H} = \mathbf{H}$ .* $(\mathbf{W}^{\mathrm{T}}\mathbf{A})./(\mathbf{W}^{\mathrm{T}}\mathbf{W}\mathbf{H} + \beta\mathbf{H} + 10^{-9})$;
    (MU)  $\mathbf{W} = \mathbf{W}. * (\mathbf{A}\mathbf{H}^{\mathrm{T}})./(\mathbf{W}\mathbf{H}\mathbf{H}^{\mathrm{T}} + \alpha\mathbf{W} + 10^{-9})$;
end
```

5. Sample applications

The remaining sections of the paper illustrate two prominent applications of NMF algorithms: text mining and spectral data analysis. In each case, several references are provided along with new results achieved with the CNMF algorithm specified above.

5.1. *Text mining for email surveillance*

The Federal Energy Regulatory Commission's (FERC) investigation of the Enron Corporation has produced a large volume of information (electronic mail messages, phone tapes, internal documents) to build a legal case against the corporation. This information initially contained over 1.5 million electronic mail (email) messages that were posted on FERC's web site (Grieve, 2003). After cleansing the data to improve document integrity and quality as well as to remove sensitive and irrelevant private information, an improved version of the *Enron Email Set* was created and publicly disseminated.[3] This revamped corpus contains 517, 431 email messages from 150 Enron employee accounts that span a period from December 1979 through February 2004 with the majority of messages spanning the three years: 1999, 2000, and 2001. Included in this corpus are email messages sent by top Enron executives including the Chief Executive Officer Ken Lay, president and Chief Operating Officer Jeff Skilling, and head of trading Greg Whalley.

Several of the topics represented by the Enron Email Set relate more to the operational logistics of what at the time was America's seventh largest company. Some threads of discussion concern the Dabhol Power Company (DPC), which Enron helped to develop in the Indian state of Maharashtra. This company was fraught with numerous logistical and political problems from the start. The deregulation of the California energy market that led to the rolling blackouts in the summer of 2000 was another hot topic reflected in the email messages. Certainly Enron and other energy-focused companies took advantage of that situation. Through its excessive greed, overspeculation, and deceptive accounting practices, Enron collapsed in the fall of 2001. After an urgent attempt to merge with the Dynegy energy company failed, Enron filed for Chapter 11 bankruptcy on December 2, 2001 (McLean and Elkind, 2003).

As initially discussed in Berry and Browne (2005a), the Enron Email Set is a truly heterogeneous collection of textual documents spanning topics from important business deals to personal memos and extracurricular activities such as fantasy football betting. As with most large-scale text mining applications, the ultimate goal is to be able to classify the communications in a meaningful way. The use of NMF algorithms for data clustering is well-documented (Xu et al., 2003; Ding et al., 2005). In the context of surveillance, an automated classification approach should be both efficient and reproducible.

5.1.1. *Electronic mail subcollections*

In Berry and Browne (2005a), the CNMF algorithm was used to compute the NMF of term-by-message matrices derived from the Enron corpus. These matrices were derived from the creation and parsing of two subcollections derived from specific mail folders from each account. In this work, we apply the CNMF algorithm with smoothness constraints only ($\beta = 0$) on the $\mathbf{W}$ matrix factor to noun-by-message matrices generated by a larger (289,695 messages) subset of the Enron corpus. Using the frequency-ranked list of English nouns provided in the British National Corpus (BNC) (British National Corpus (BNC), 2004), 7424 nouns were previously extracted from the 289,695-message Enron subset (Keila and Skillicorn, 2005). A noun-by-message matrix $\mathbf{A} = [\mathbf{A}_{ij}]$ was then constructed so that $\mathbf{A}_{ij}$ defines a frequency at which noun i occurs in message j. Like the matrices constructed in Berry and Browne (2005a), Shahnaz et al. (2006), statistical weighting techniques were applied to the elements of matrix $\mathbf{A}$ in order to create more meaningful noun-to-message associations for concept discrimination (Berry and Browne, 2005b).

Unlike previous studies (Berry and Browne, 2005a), no restriction was made for the global frequency of occurrence associated with each noun in the resulting dictionary. In order to define meaningful noun-to-message associations for concept discrimination, however, term weighting was used to generate the resulting 289, 695 × 7424 message-by-noun matrix (transposed form is normally generated as the complete dictionary cannot be fully resolved till all messages have been parsed).

[3] See http://www-2.cs.cmu.edu/∼enron.

5.1.2. *Term weighting*

As explained in Berry and Browne (2005b), a collection of n messages indexed by m terms (or keywords) can be represented as a $m \times n$ term-by-message matrix $\mathbf{A} = [\mathbf{A}_{ij}]$. Each element or component $\mathbf{A}_{ij}$ of the matrix $\mathbf{A}$ defines a *weighted* frequency at which term i occurs in message j. In the extraction of nouns from the 289,695 Enron email messages, we simply define $\mathbf{A}_{ij} = l_{ij}g_i$, where l_{ij} is the local weight for noun i occurring in message j and g_i is the global weight for noun i in the email subset. Suppose f_{ij} defines be the number of times (frequency) that noun i appears in message j, and $p_{ij} = f_{ij}/\sum_j f_{ij}$. The *log-entropy* term weighting scheme (Berry and Browne, 2005a) used in this study is defined by $l_{ij} = \log(1 + f_{ij})$ and $g_i = 1 + ((\sum_j p_{ij} \log(p_{ij}))/\log n)$, where all logarithms are base 2. Term weighting is typically used in text mining and retrieval to create better term-to-document associations for concept discrimination.

5.1.3. *Observations*

To demonstrate the use of the CNMF algorithm with smoothing on the W matrix, we approximate the $7424 \times 289,695$ Enron noun-by-message matrix $\boldsymbol{X}$ via

$$\mathbf{A} \simeq \mathbf{WH} = \sum_{i=1}^{50} \mathbf{W}_i\mathbf{H}^i, \tag{10}$$

where $\mathbf{W}$ and $\mathbf{H}$ are 7424×50 and $50 \times 289,695$, respectively, nonnegative matrices. $\mathbf{W}_i$ denotes the ith column of $\mathbf{W}$, $\mathbf{H}^i$ denotes the ith row of the matrix $\mathbf{H}$, and $k = 50$ factors or parts are produced. The nonnegativity of the $\mathbf{W}$ and $\mathbf{H}$ matrix factors facilitates the parts-based representation of the matrix $\mathbf{A}$ whereby the basis (column) vectors of $\mathbf{W}$ or $\mathbf{W}_i$ combine to approximate the original columns (messages) of the sparse matrix $\mathbf{A}$. The outer product representation of $\mathbf{WH}$ in Eq. (10) demonstrates how the rows of $\mathbf{H}$ or $\mathbf{H}^i$ essentially specify the weights (scalar multiples) of each of the basis vectors needed for each of the 50 parts of the representation. As described in Lee and Seung (1999), we can interpret the semantic feature represented by a given basis vector $\mathbf{W}_i$ by simply sorting (in descending order) its 7424 elements and generating a list of the corresponding dominant nouns for that feature. In turn, a given row of $\mathbf{H}$ having n elements (i.e., $\mathbf{H}^i$) can be used to reveal messages sharing common basis vectors $\mathbf{W}_i$, i.e., similar semantic features or meaning. The columns of $\mathbf{H}$, of course, are the projections of the columns (messages) of $\mathbf{A}$ onto the basis spanned by the columns of $\mathbf{W}$. The best choice for the number of parts k (or column rank of $\mathbf{W}$) is certainly problem-dependent or corpus-dependent in this context. However, as discussed in Shahnaz et al. (2006) for standard topic detection benchmark collections (with human-curated document clusters), the accuracy of the CNMF algorithm for document clustering degrades as the rank k increases or if the sizes of the clusters become greatly imbalanced.

The association of features (i.e., feature vectors) to the Enron mail messages is accomplished by the nonzeros of each $\mathbf{H}^i$ which would be present in the ith *part* of the approximation to $\mathbf{A}$ in Eq. (10). Each part (or span of $\mathbf{W}_i$) can be used to classify the messages so the sparsity of $\mathbf{H}$ greatly affects the diversity of topics with which any particular semantic feature can be associated. Using the rows of the $\mathbf{H}$ matrix and a threshold value for all nonzero elements, one can produce clusters of messages that are described (or are spanned by) similar feature vectors ($\mathbf{W}_i$). Effects associated with the smoothing of the matrix $\mathbf{H}$ (e.g., reduction in the size of document clusters) are discussed in Shahnaz et al. (2006).

In Shahnaz et al. (2006), a gradual reduction in elapsed CPU time was observed for a similar NMF computation (for a heterogeneous newsfeed collection) based on smoothing the $\mathbf{H}$ matrix factor. As mentioned in Section 4 and discussed in Karvanen and Cichocki (2003), ℓ^p norms for $0 < p \leqslant 1$ can be used to measure changes in the sparsity of either $\mathbf{H}$ or $\mathbf{W}$. A consistent reduction in $\|\mathbf{H}\|_p/\|\mathbf{H}\|_1$ was demonstrated in Shahnaz et al. (2006) for both $p = 0.5$ and 0.1 to verify the increase in sparsity as the value of the smoothing parameter was increased. Fig. 1 illustrates the variation in $\|\mathbf{W}\|_p/\|\mathbf{W}\|_1$ for $p = 0.5$ along with the elapsed CPU timings for computing the NMF of the noun-by-message matrix. We note that in this study, smoothing on the $50 \times 289,695$ matrix $\mathbf{H}$ yielded no improvement in the clustering of identifiable features and even stalled convergence of the CNMF algorithm. Smoothing on the smaller 7424×50 matrix $\mathbf{W}$ did improve the cost-per-iteration of CNMF but not in a consistent manner as α was increased. The gain in sparsity associated with settings of $\alpha = 0.1, 0.75$ did slightly reduce the required computational time for 100 iterations of CNMF on a 3.2 GHz Intel Xeon 3.2 GHz having a 1024 KB cache and 4.1 GB RAM (see Fig. 1). Further research is clearly needed to better calibrate smoothing for an efficient yet interpretable NMF of large sparse (unstructured) matrices from text mining.

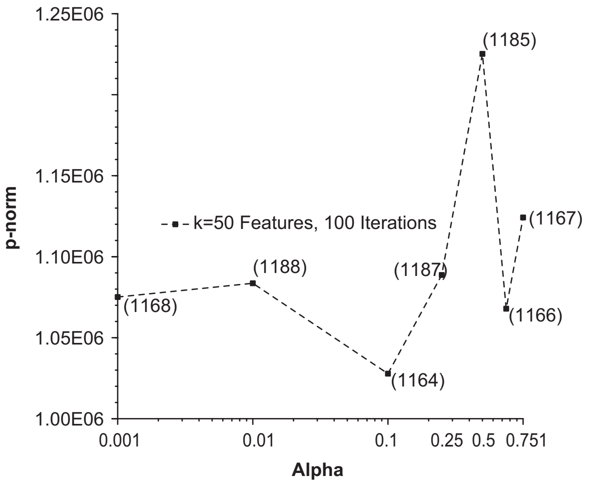

Fig. 1. Reduction in $\|\mathbf{W}\|_p/\|\mathbf{W}\|_1$ (*p-norm*) for $p=0.5$ as the smoothing parameter α is increased for the nonnegative matrix factorization $\mathbf{A}=\mathbf{WH}$ of the Enron noun-by-message matrix. Elapsed CPU times (seconds) for CNMF to produce $k=50$ features in 100 iterations are provided in parentheses.

Table 1
Six Enron clusters (topics) identified by the rows of H or H^i produced by the CNMF algorithm with smoothing on the matrix $\mathbf{W}$ ($\alpha=0.1,\ \beta=0$)

Feature index (i)	Cluster size	Topic description	Dominant nouns
2	3970	Professional football	Bettis, McNair, McNabb, stats, Faulk, rushing
10	6190	California blackout	Fichera, Escondido, biomass
13	9234	Enron downfall	Vinson, Elkins, destruction, scandal, auditing, Lieberman
18	3583	Business ventures	Teijin, Janus, Coale, Tata, BSES
21	4011	India	Prabhu, Cline, Suresh, rupees, Vinay, renegotiation
23	8526	World energy/ Scotland	Scottish, power, ENEL, Mitsui, vessel

Exactly $k=50$ feature vectors ($\mathbf{W}_i$) were generated. Several of the dominant (having values of largest magnitude) nouns for each feature vector are listed for each selected feature (i). Cluster size reflects the number of row elements in H^i of magnitude greater than $row_{\max}/10$.

5.1.4. Topic extraction

Table 1 illustrates some of the extracted topics (i.e., message clusters) as evidenced by large components in the same row of the matrix $\mathbf{H}$ (or $\mathbf{H}^i$) generated by CNMF for the sparse noun-by-message Enron matrix. The terms corresponding to the 10-largest elements of the particular feature (or part) i are also listed to explain and derive the context of the topic. By feature, we are referring to the ith column of the matrix factor $\mathbf{W}$ or $\mathbf{W}_i$ in Eq. (10), of course.

In a perfect email surveillance world, each cluster of nouns would point to the documents by a specific topic. Although our experiments did not produce such results for every cluster, they did give some indication of the *nature* of a message subset. With 50 clusters or features produced by CNMF from the sparse noun-by-message matrix $\mathbf{A}$, we analyzed several of the dominant (in magnitude) nouns per feature for clues about the content of a cluster. The initial set of nouns for tracking the topics illustrated in Table 1 was obtained by simply ranking all nouns according to their global

Table 2
Top 20 BNC nouns by ascending global entropy weight (g_i) and the corresponding topics identified among the 50 features ($\mathbf{W}_i$) generated by CNMF using different values of the smoothing parameter α

Noun	f_i	g_i	Cluster/topic frequency per α							Topic
			0.001	0.01	0.1	0.25	0.50	0.75	1.0	
Waxman	680	0.424	2		2	2	2	2		Downfall
Lieberman	915	0.426	2	2	2	2			2	Downfall
Scandal	679	0.428	2				2		2	Downfall
Nominee	544	0.436		4	3	2		2	2	
Barone	470	0.437	2	2	2				2	Downfall
Meade	456	0.437							2	Downfall
Fichera	558	0.438	2			2				CA blackout
Prabhu	824	0.445	2	2	2	2		2	2	India
Tata	778	0.448							2	India
Rupee	323	0.452	3	4	4	4	3	4	2	India
Soybean	499	0.455	2	2	2	2	2	2	2	
Rushing	891	0.486	2	2	2					Football
Dlrs [dollars]	596	0.487							2	
Janus	580	0.488	2	3				2	3	Business ventures
BSES	451	0.498	2	2					2	Business ventures
Caracas	698	0.498						2		
Escondido	326	0.504	2			2				CA blackout
Promoters	180	0.509	2							Energy/scots
Aramco	188	0.550	2							India
Doorman	231	0.598		2						

entropy weight g_i (see Section 5.1.2). As illustrated in Table 2, the occurrence of these higher entropy nouns varies with changes in the smoothing parameter α which can be used to produce sparser feature vectors $\mathbf{W}_i$. Determining *indicator* nouns (or terms in general) that can be used to track specific topics or discussions in electronic messages is a critical task in surveillance-based applications.

The clusters depicted in Table 1 reflect some of the more *meaningful* topics extracted from the Enron email subset. Professional football discussions were clearly identified by the names of players (e.g., Jerome *Bettis*, Steve *McNair*, and Marshall *Faulk*) and the prominent cluster of messages on the downfall of Enron are tracked by names of the Houston law firm representing the company (e.g., *Vinson* and *Elkins*) or Senate investigator (Joesph *Lieberman*). Business ventures with firms such as *Teijin* and *Tata Power*, and negotiations between Enron officials (e.g., Wade *Cline*) and foreign officials (India's Federal Power Minister *Suresh Prabhu* and Chairman of the Maharashtra State Electricity Board *Vinay* Bansai) were also identified. Certainly expanding the number of high entropy terms for tracking topics would support better (human-curated) topic interpretations. This was evidenced by the need to track more nouns for the California blackout discussion trail. Keep in mind that cluster or feature size is measured by the number of row elements in the matrix **H** with magnitude greater than a specified tolerance (which is $row_{\max}/10$ for this study). Adjustments to this tolerance value could make the number of messages more manageable for inspectors.

5.2. *Spectral unmixing for non-resolved space object characterization*

The key problem in non-resolved space object characterization is to use spectral reflectance data to gain knowledge regarding the physical properties (e.g., function, size, type, status change) of space objects that cannot be spatially resolved with telescope technology. Such objects may include geosynchronous satellites, rocket bodies, platforms, space debris, or nano-satellites. Fig. 2 shows an artist's rendition of a JSAT type satellite in a 36,000 km high synchronous orbit around the Earth. Even with adaptive optics capabilities, this object is generally not resolvable using ground-based telescope technology. For safety and other considerations in space, non-resolved space object characterization is an important component of Space Situational Awareness.

Spectral reflectance data of a space object can be gathered using ground-based spectrometers and contains essential information regarding the make up or types of materials comprising the object. Different materials such as aluminum,

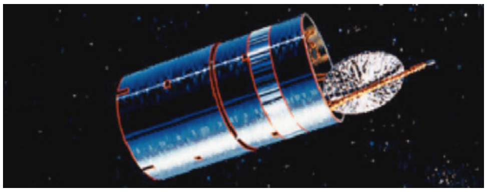

Fig. 2. Artist rendition of a JSAT satellite. Image obtained from the Boeing Satellite Development Center.

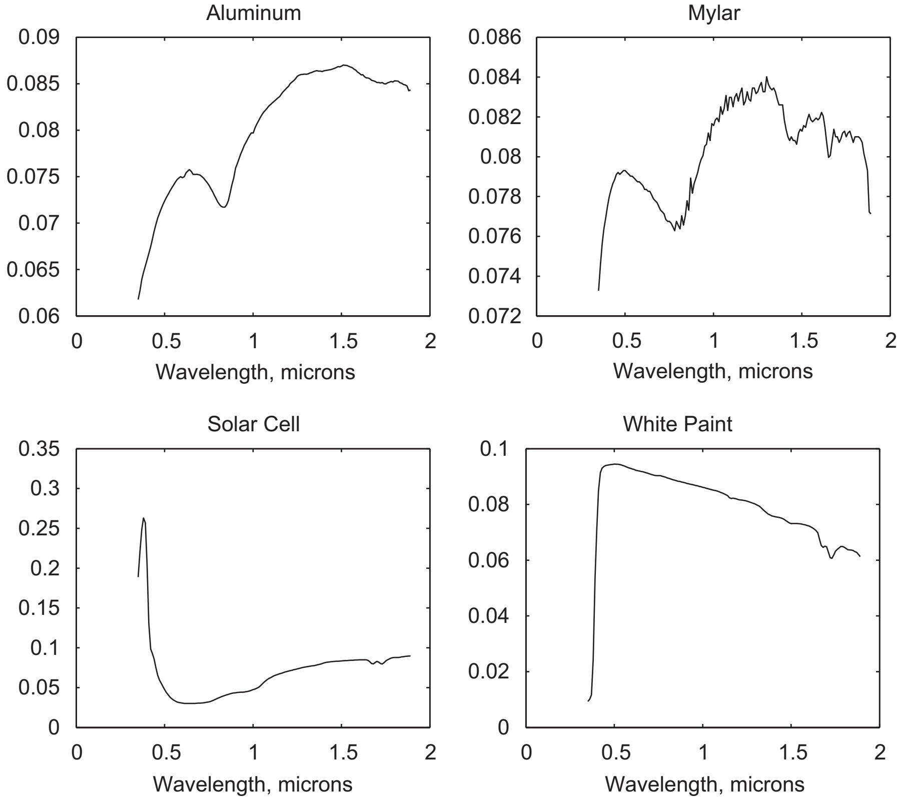

Fig. 3. Laboratory spectral signatures for aluminum, mylar, solar cell, and white paint. For details see Luu et al. (2003).

mylar, paint, etc. possess characteristic wavelength-dependent absorption features, or spectral *signatures*, that mix together in the spectral reflectance measurement of an object. Fig. 3 shows spectral signatures of four materials typically used in satellites, namely, aluminum, mylar, white paint, and solar cell.

The objective is then, given a set of spectral measurements or traces of an object, to determine (i) the type of constituent materials and (ii) the proportional amount in which these materials appear. The first problem involves the detection of material spectral signatures or *endmembers* from the spectral data. The second problem involves the computation of corresponding proportional amounts or *fractional abundances*. This is known as the *spectral unmixing* problem in the hyperspectral imaging community (Chang, 2000; Keshava, 2003; Plaza et al., 2004).

A reasonable assumption for spectral unmixing is that a spectral measurement of an object results from a *linear* combination of the spectral signatures of its constituent materials. Hence in the linear mixing model, a spectral measurement of an object $\mathbf{y} \geqslant 0$ along m spectral bands is given by $\mathbf{y} = \mathbf{U}\mathbf{v} + \mathbf{n}$, where $\mathbf{U} \geqslant 0$ is an $m \times k$ matrix whose columns are the spectral reflectance signatures of k constituent materials (endmembers), $\mathbf{v} \geqslant 0$ is a vector of fractional abundances, and $\mathbf{n}$ is a noise term. For n spectral measurements we write in block form

$$\mathbf{Y} = \mathbf{UV} + \mathbf{N}, \tag{11}$$

where now $\mathbf{Y}$ is the $m \times n$ data matrix whose columns are spectral measurements, $\mathbf{V}$ is a $k \times n$ matrix of fractional abundances, and $\mathbf{N}$ is noise.

In this section, we apply NMF to the spectral unmixing problem. Specifically, we minimize the extended cost function (3) with $\mathbf{A} = \mathbf{Y}$ and seek column vectors in basis matrix $\mathbf{W}$ that approximate endmembers in matrix $\mathbf{U}$. We then gather the best computed endmembers into a matrix $\mathbf{B} \approx \mathbf{U}$ and solve an inverse problem to compute $\mathbf{H} \approx \mathbf{V}$. Other authors have investigated the use of NMF for spectral unmixing (Plaza et al., 2004). We report on the use of smoothness constraints and the CNMF algorithm presented in Section 4 for improved extraction of material spectral signatures.

5.3. Simulation setup

While spectral reflectance data of space objects is being collected at various sites, such as the Air Force Maui Optical and Supercomputing Site in Maui, HI, publicly disseminated reflectance data of space objects is, to our knowledge, not yet available. Hence, like other researchers (Luu et al., 2003), we employed laboratory-obtained spectral signatures for the creation of simulated data using the linear mixing model of Eq. (11).

More specifically, we let $\mathbf{U}=[\mathbf{u}_1\mathbf{u}_2\mathbf{u}_3\mathbf{u}_4]$ where $\mathbf{u}_i$ are the spectra associated with the four materials in Fig. 3 measured in the 0.3–1.8 μm range, with $m=155$ spectral bands. We form the matrix of fractional abundances as $\mathbf{V}=[\mathbf{V}_1\mathbf{V}_2\mathbf{V}_3\mathbf{V}_4]$, where $\mathbf{V}_i$ is a 4×100 matrix whose rows vary sinusoidally with random amplitude, frequency and phase shift as to model space object rotation with respect to a fixed detector, but are chosen so that the ith material spectral signature is dominant in the mixture $\mathbf{Y}_i = \mathbf{UV}_i + \mathbf{N}_i$. The noise is chosen to be 1% Gaussian, that is $\|\mathbf{N}_i\|_\mathrm{F} = 0.01 * \|\mathbf{Y}_i\|_\mathrm{F}/\|\mathbf{R}_i\|_\mathrm{F}$ where the entries in $\mathbf{R}_i$ are randomly chosen from a Gaussian distribution $N(0, 1)$. Representative simulated spectral traces from each sub-data set $\mathbf{Y}_i$ are shown in Fig. 4 and are consistent with simulated and real data employed in related work (Luu et al., 2003).

5.4. Numerical results

We first focus on the problem of endmember determination from the simulated spectral measurements of Fig. 4. To do this, we minimize cost function (3) using algorithm CNMF given in Section 4, with $\mathbf{A} = [\mathbf{Y}_1\mathbf{Y}_2\mathbf{Y}_3\mathbf{Y}_4]$ and with smoothness constraint (4) only in $\mathbf{W}$. The regularization parameter α was set to $\alpha = 0$ (no auxiliary constraint) and $\alpha = 1$ (penalize solutions $\mathbf{W}$ of large Frobenius norm). For each value of α we run CNMF with 20 different starting points and number of basis $k = 6$, resulting in a total of 20 candidate (local) solutions $\mathbf{W}$ or a set of 120 candidate endmembers $\{\mathbf{w}_i\}$, $(i = 1 : 120)$.

We employ an information theoretic measure to quantitatively evaluate the performance in endmember extraction of NMF with and without smoothness constraints. The measure of choice is $D_\mathrm{s}(\mathbf{u}_j, \mathbf{w}_i)$, the symmetric Kullback–Leibler divergence of $\mathbf{u}_j$ with respect to $\mathbf{w}_i$ given by

$$D_\mathrm{s}(\mathbf{u}_j, \mathbf{w}_i) = D(\mathbf{u}_j\|\mathbf{w}_i) + D(\mathbf{u}_j\|\mathbf{w}_i), \tag{12}$$

where

$$D(\mathbf{x}\|\mathbf{z}) = \sum_{\ell} \frac{\mathbf{x}(\ell)}{\|\mathbf{x}\|_1} \log\left(\frac{\mathbf{x}(\ell)\|\mathbf{z}\|_1}{\mathbf{z}(\ell)\|\mathbf{x}\|_1}\right).$$

For any nonnegative $\mathbf{x}$ and $\mathbf{z}$, the symmetric Kullback–Leibler divergence satisfies $D_\mathrm{s}(\mathbf{x}, \mathbf{z}) \geqslant 0$, and the smaller the value of $D_\mathrm{s}(\mathbf{x}, \mathbf{z})$ the better the match. Thus, for each *true* endmember $\mathbf{u}_j$ with $j = 1 : 4$, we find the best match $\mathbf{w}_* \in \{\mathbf{w}_i\}$ as,

$$\mathbf{u}_j \approx \mathbf{w}_* = \arg\min_i \{D_\mathrm{s}(\mathbf{u}_j, \mathbf{w}_i)\}.$$

The scores of the best matches found in $\{\mathbf{w}_i\}$ for each of the four materials are given in Table 3. Values in parentheses correspond to the scores of best matches found with NMF and no auxiliary constraints ($\alpha = 0$). Notice that

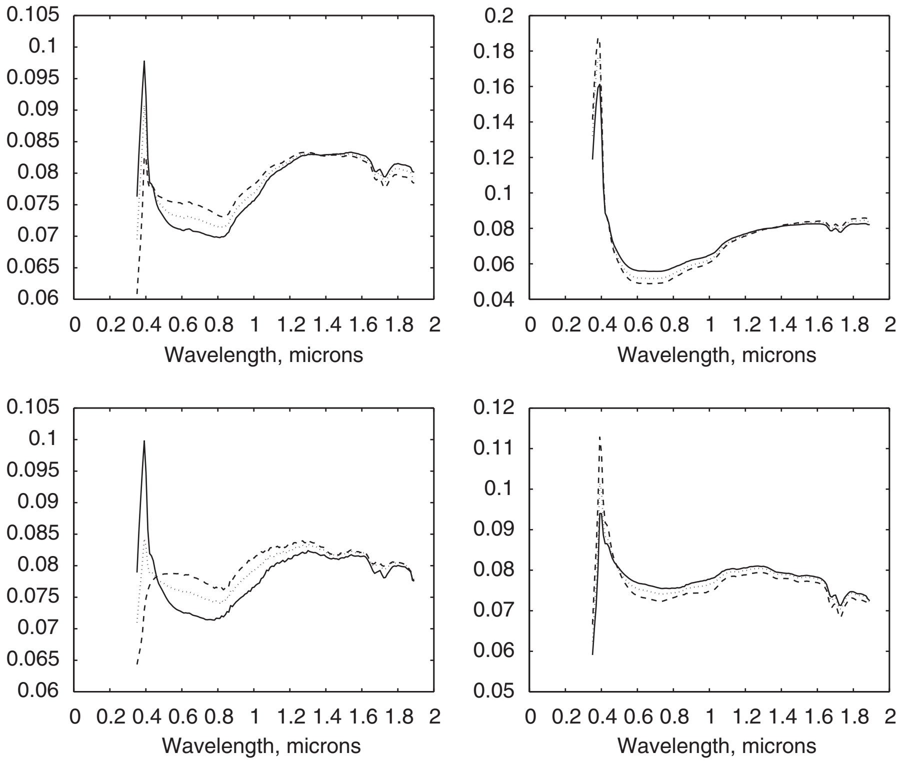

Fig. 4. Representative simulated spectral traces from $\mathbf{Y}_i$ $(i = 1 : 4)$, where i is the dominant material in the mixture. (top-left) $\mathbf{Y}_1$: aluminum. (top-right) $\mathbf{Y}_2$: mylar. (bottom-left) $\mathbf{Y}_3$: solar cell. (bottom-right) $\mathbf{Y}_4$: white paint.

Table 3
Kullback–Leibler divergence scores of the best matching endmembers given by NMF with and without smoothness constraints

Input	Aluminum	Mylar	Solar cell	White paint
$[\mathbf{Y}_1\mathbf{Y}_2\mathbf{Y}_3\mathbf{Y}_4]$	0.0280 (0.1137)	0.0659 (0.0853)	0.0130 (0.0161)	0.0223 (0.0346)

Values in parentheses correspond to standard NMF (no constraints).

in all cases the scores corresponding to CNMF with $\alpha = 1$ are lower than those obtained with no constraints at all, showing that enforcing smoothness constraints can significantly improve the performance of NMF for feature extraction.

Table 4 shows the scores of the best matches obtained with CNMF when the input data matrix to CNMF is $\mathbf{A} = \mathbf{Y}_i$, rather than $A = [\mathbf{Y}_1\mathbf{Y}_2\mathbf{Y}_3\mathbf{Y}_4]$. Again scores corresponding to CNMF with $\alpha = 1$ are generally smaller than those obtained with standard NMF. It is interesting to note that for $\mathbf{Y}_3$ the smoothness constraint enabled significantly better performance for aluminum and mylar, at the expense of a slight loss of performance in the determination of solar cell and white paint.

The visual quality of endmember spectra computed using CNMF with $\alpha = 1$ can be appreciated in Fig. 5. Notice that using the Kullback–Leibler divergence measure both aluminum and mylar are well represented by the same computed endmember spectra.

This result is not surprising as the spectral signatures of aluminum and mylar are quite similar (see Fig. 3). This strong similarity can also be easily observed in the confusion matrix shown in Table 5.

Table 4
Kullback–Leibler divergence scores of the best matching endmembers given by NMF with and without smoothness constraints

Input (dominant)	Aluminum	Mylar	Solar cell	White paint
$\mathbf{Y}_1$ (aluminum)	0.0233	0.0124	0.4659	0.1321
	(0.0740)	(0.0609)	(0.6844)	(0.1661)
$\mathbf{Y}_2$ (mylar)	0.0165	0.0063	0.4009	0.1203
	(0.0615)	(0.0606)	(0.5033)	(0.1827)
$\mathbf{Y}_3$ (solar cell)	0.0645	0.0292	0.0302	0.2863
	(0.1681)	(0.1358)	(0.0266)	(0.1916)
$\mathbf{Y}_4$ (white paint)	0.0460	0.0125	0.8560	0.1735
	(0.0882)	(0.0571)	(0.8481)	(0.2845)

Values in parentheses correspond to standard NMF (no constraints).

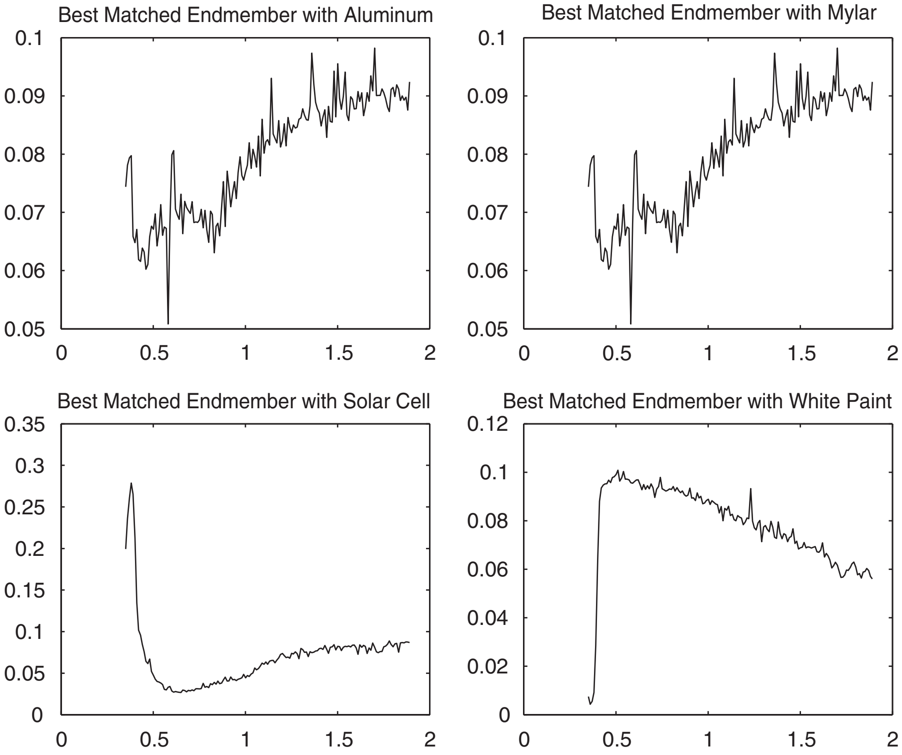

Fig. 5. Plot of the best endmembers extracted using CNMF with $\alpha = 1$, whose Kullback–Leibler divergence scores appear in Table 3.

Table 5
Confusion matrix of the four material spectra in Fig. 3

	Aluminum	Mylar	Solar cell	White paint
Aluminum	0	0.0209	1.2897	0.3317
Mylar	—	0	1.2719	0.2979
Solar cell	—	—	0	2.5781
White paint	—	—	—	0

The scores correspond to $D_s(\mathbf{u}_i, \mathbf{u}_j)$ for $i = 1 : 4$ and $j = i : 4$.

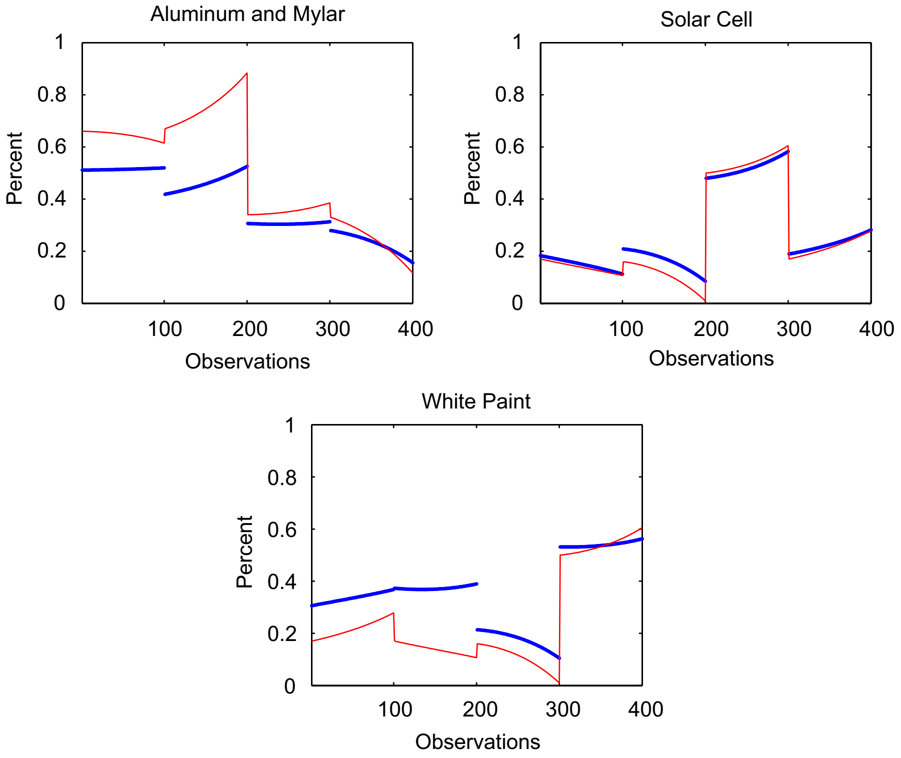

Fig. 6. Fractional abundances obtained using PMRNSD with $\mathbf{B} \approx \mathbf{U}$ given in Fig. 5. True fractional abundances are represented in red. The result of the approximation using PMRNSD is in blue.

Next, we focus on the inverse process of computing fractional abundances given the data matrix $\mathbf{A} = [\mathbf{Y}_1\mathbf{Y}_2\mathbf{Y}_3\mathbf{Y}_4]$ and computed material spectral signatures $\mathbf{B} \approx \mathbf{U}$. Thus we minimize

$$\min_{\mathbf{H}} \|\mathbf{A} - \mathbf{B}\mathbf{H}\|_F^2 \quad \text{subject to } \mathbf{H} \geqslant 0, \tag{13}$$

where $\mathbf{B} \approx \mathbf{U}$ is formed using computed endmember spectra shown in Fig. 5 and selected using the Kullback–Leibler divergence measure of Eq. (12). We use algorithm PMRNSD (Nagy and Strakos, 2000) for the numerical minimization of (13).

Fig. 6 shows the fractional abundances calculated with PMRNSD in blue, compared with the true fractional abundances of matrix $\mathbf{V}$ in red. Here aluminum and mylar are shown in the same plot since the same computed endmember was selected to represent these materials. The calculated fractional abundances for solar cell are quite good, while those for the other materials are less accurate. Note that in all cases the relative change of fractional abundance per observation is well represented by the calculated fractional abundances.

6. Further improvements

In this paper, we have attempted to outline some of the major concepts related to nonnegative matrix factorization. In addition to developing applications for space object identification and classification and topic detection and tracking in text mining, several open problems with NMF remain. Here are a few of them:

- Initializing the factors **W** and **H**: Methods for choosing, or seeding, the initial matrices **W** and **H** for various algorithms (see, e.g., Wild, 2002; Wild et al., 2003; Boutsidis and Gallopoulos, 2005) is a topic in need of further research.
- Uniqueness: Sufficient conditions for uniqueness of solutions to the NMF problem can be considered in terms of simplicial cones (Berman and Plemmons, 1994), and have been studied in Donoho and Stodden (2003). Algorithms

for computing the factors **W** and **H** generally produce local minimizers of $f(\mathbf{W}, \mathbf{H})$, even when constraints are imposed. It would thus be interesting to apply global optimization algorithms to the NMF problem.
- Updating the factors: Devising efficient and effective updating methods when columns are added to the data matrix **A** in Eq. (1) also appears to be a difficult problem and one in need of further research.

Our plans are thus to continue the study of nonnegative matrix factorizations and develop further applications to spectral data analysis. Work on applications to air emission quality (Chu et al., 2004) and on text mining (Pauca et al., 2004; Shahnaz et al., 2006) will also be continued.

Acknowledgments

The authors would like to thank the anonymous referees for their valuable comments and suggestions on the original manuscript.

References

Berman, A., Plemmons, R., 1994. Non-Negative Matrices in the Mathematical Sciences. SIAM Press Classics Series, Philadelphia, PA.

Berman, G., 1969. Lattice approximations to the minima of functions of several variables. J. Assoc. Comput. Mach. 16, 286–294.

Berry, M., Browne, M., 2005a. Email surveillance using nonnegative matrix factorization. Comput. Math. Organization Theory 11, 249–264.

Berry, M., Browne, M., 2005b. Understanding Search Engines: Mathematical Modeling and Text Retrieval. second ed.. SIAM, Philadelphia, PA.

Bertsekas, D., 1999. Nonlinear Programming. Athena Scientific, Belmont, MA.

Boutsidis, C., Gallopoulos, E., 2005. On SVD-based initialization for nonnegative matrix factorization. Technical Report HPCLAB-SCG-6/08-05, University of Patras, Patras, Greece.

British National Corpus (BNC), 2004. ⟨http://www.natcorp.ox.ac.uk⟩

Bro, R., de Jong, S., 1997. A fast non-negativity constrained linear least squares algorithm. J. Chemometrics 11, 393–401.

Catral, M., Han, L., Neumann, M., Plemmons, R., 2004. On reduced rank non-negative factorization for symmetric non-negative matrices. Linear Algebra Appl. 393, 107–126.

Cea, J., 1971. Optimisation: Theorie et algorithmes. Dunod, Paris.

Chang, C.-I., 2000. An information theoretic-based approach to spectral variability, similarity, and discriminability for hyperspectral image analysis. IEEE Trans. Inform. Theory 46, 1927–1932.

Chen, Z., Cichocki, A., 2005. Nonnegative matrix factorization with temporal smoothness and/or spatial decorrelation constraints, preprint.

Chu, M., Diele, F., Plemmons, R., Ragni, S., 2004. Optimality, computation, and interpretations of nonnegative matrix factorizations, available at ⟨http://www.wfu.edu/∼plemmons⟩.

Cichocki, A., Zdunek, R., 2006. NMFLAB for Signal Processing, available at ⟨http://www.bsp.brain.riken.jp/ICALAB/nmflab.html⟩.

Cichocki, A., Zdunek, R., Amari, S., 2006. Csiszar's divergences for non-negative matrix factorization: family of new algorithms. In: Proceedings of the Sixth International Conference on Independent Component Analysis and Blind Signal Separation, Charleston, SC, March 5–8.

de Leeuw, J., Young, F., Takane, Y., 1976. Additive structure in qualitative data: an alternating least squares method with optimal scaling features. Psychometrika 41, 471–503.

Dhillon, I., Sra, S., 2005. Generalized nonnegative matrix approximations with bregman divergences. In: Proceeding of the Neural Information Processing Systems (NIPS) Conference, Vancouver, BC.

Ding, C., He, X., Simon, H., 2005. On the equivalence of nonnegative matrix factorization and spectral clustering. In: Proceedings of the Fifth SIAM International Conference on Data Mining, Newport Beach, CA.

Donoho, D., Stodden, V., 2003. When does non-negative matrix factorization give a correct decomposition into parts? In: Seventeenth Annual Conference on Neural Information Processing Systems.

Drineas, P., Kannan, R., Mahoney, M., 2006. Fast Monte Carlo algorithms for matrices iii: computing a compressed approximate matrix decomposition. SIAM J. Comput. 36 (1), 132–157.

Finesso, L., Spreij, P., 2004. Approximate nonnegative matrix factorization via alternating minimization. In: Proceedings of the 16th International Symposium on Mathematical Theory of Networks and Systems, Leuven, Belgium, July 5–9.

Gill, P., Murray, W., Wright, M., 1981. Practical Optimization. Academic Press, London.

Gonzalez, E., Zhang, Y., 2005. Accelerating the Lee–Seung algorithm for nonnegative matrix factorization. Technical Report TR-05-02, Rice University.

Grieve, T., 2003. The Decline and Fall of the Enron Empire. Slate ⟨http://www.salon.com/news/feature/2003/10/14/enron/index_np.html⟩.

Grippo, L., Sciandrone, M., 2000. On the convergence of the block nonlinear Gauss–Seidel method under convex constraints. Oper. Res. Lett. 26 (3), 127–136.

Guillamet, D., Bressan, M., Vitria, J., 2001. A weighted non-negative matrix factorization for local representations. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, Kavai, HI, pp. 942–947.

Hamza, A.B., Brady, D., 2006. Reconstruction of reflectance spectra using robust non-negative matrix factorization. IEEE Transactions on Signal Processing 54 (9), 3637–3642.

Hoyer, P., 2002. Non-negative sparse coding. In: Proceedings of the IEEE Workshop on Neural Networks for Signal Processing. Martigny, Switzerland.

Hoyer, P., 2004. Non-negative matrix factorization with sparseness constraints. J. Mach. Learning Res. 5, 1457–1469.

Karvanen, J., Cichocki, A., 2003. Measuring sparseness of noisy signals. In: Proceedings of the Fourth International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003), Nara, Japan.

Keila, P., Skillicorn, D., 2005. Detecting unusual and deceptive communication in email. Technical Report, School of Computing, Queen's University, Kingston, Ont., Canada.

Keshava, N., 2003. A survey of spectral unmixing algorithms. Lincoln Laboratory J. 14 (1), 55–77.

Langville, A., Meyer, C., Albright, R., Cox, J., Duling, D., 2006. Algorithms, initializations, and convergence for the nonnegative matrix factorization, preprint.

Lawson, C., Hanson, R., 1995. Solving Least Squares Problems. SIAM, Philadelphia, PA.

Lee, D., Seung, H., 1999. Learning the parts of objects by non-negative matrix factorization. Nature 401, 788–791.

Lee, D., Seung, H., 2001. Algorithms for non-negative matrix factorization. Adv. Neural Inform. Process. Systems 13, 556–562.

Lin, C.-J., 2005a. On the convergence of multiplicative update algorithms for non-negative matrix factorization. Technical Report Information and Support Services Technical Report, Department of Computer Science, National Taiwan University.

Lin, C.-J., 2005b. Projected gradient methods for non-negative matrix factorization. Technical Report Information and Support Services Technical Report ISSTECH-95-013, Department of Computer Science, National Taiwan University.

Luu, K., Matson, C., Snodgrass, J., Giffin, M., Hamada, K., Lambert, J., 2003. Object characterization from spectral data. In: Proceedings of the 2003 AMOS Technical Conference, Maui, HI, September.

McLean, B., Elkind, P., 2003. The Smartest Guys in the Room: The Amazing Rise and Scandalous Fall of Enron. Portfolio.

Nagy, J.G., Strakos, Z., 2000. Enforcing nonnegativity in image reconstruction algorithms. Math. Modeling, Estimation, and Imaging 4121, 182–190.

Nocedal, J., Wright, S., 1999. Numerical Optimization. Springer, Berlin.

Paatero, P., 1997. Least squares formulation of robust non-negative factor analysis. Chemometrics and Intell. Laboratory Syst. 37, 23–35.

Paatero, P., 1999. The multilinear engine—a table-driven least squares program for solving multilinear problems, including the n-way parallel factor analysis model. J. Comput. Graphical Statist. 8 (4), 1–35.

Paatero, P., Tapper, U., 1994. Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values. Environmetrics 5, 111–126.

Pauca, V., Shahnaz, F., Berry, M., Plemmons, R., 2004. Text mining using non-negative matrix factorizations. In: Proceedings of the Fourth SIAM International Conference on Data Mining, April 22–24. SIAM, Lake Buena Vista, FL.

Pauca, P., Piper, J., Plemmons, R., 2006a. Nonnegative matrix factorization for spectral data analysis. Linear Algebra Appl. 416 (1), 29–47.

Pauca, V., Plemmons, R., Abercromby, K., 2006b. Nonnegative matrix factorization methods with physical constraints for spectral unmixing, in preparation.

Piper, J., Pauca, V., Plemmons, R., Giffin, M., 2004. Object characterization from spectral data using nonnegative factorization and information theory. In: Proceedings of the 2004 AMOS Technical Conference, Maui, HI, September.

Plaza, A., Martinez, P., Perez, R., Plaza, J., 2004. A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data. IEEE Trans. on Geoscience and Remote Sensing 42 (3), 650–663.

Polak, E., 1971. Computational Methods in Optimization: A Unified Approach. Academic Press, New York.

Powell, M., 1964. An efficient method for finding the minimum of a function of several variables without calculating derivatives. Comput. J. 7, 155–162.

Powell, M., 1973. On search directions for minimization. Math. Programming 4, 193–201.

Salakhutdinov, R., Roweis, S.T., Ghahramani, Z., 2003. On the convergence of bound optimization algorithms. Uncertainty in Artificial Intelligence 19, 509–516.

Shahnaz, F., Berry, M., Pauca, V., Plemmons, R., 2006. Document clustering using nonnegative matrix factorization. Inform. Process. Manage. 42 (2), 373–386.

Smilde, A., Bro, R., Geladi, P., 2004. Multi-way Analysis. Wiley, West Sussex, England.

Torczon, V., 1997. On the convergence of pattern search algorithms. SIAM J. Optim. 7, 1–25.

Wang, Y., Jiar, Y., Hu, C., Turk, M., 2004. Fisher non-negative matrix factorization for learning local features. In: Asian Conference on Computer Vision, Korea, January 27–30.

Wild, S., 2002. Seeding non-negative matrix factorization with the spherical k-means clustering. Master's Thesis, University of Colorado.

Wild, S., Curry, J., Dougherty, A., 2003. Motivating non-negative matrix factorizations. In: Proceedings of the Eighth SIAM Conference on Applied Linear Algebra, July 15–19. SIAM, Williamsburg, VA, ⟨http://www.siam.org/meetings/la03/proceedings⟩.

Wold, H., 1966. Nonlinear estimation by iterative least squares procedures. In: David, F. (Ed.), Research Papers in Statistics. Wiley, New York, pp. 411–444.

Wold, H., 1975. Soft modelling by latent variables: nonlinear iterative partial least squares approach. In: Gani, J. (Ed.), Perspectives in Probability and Statistics. Academic Press, London, pp. 411–444.

Xu, W., Liu, X., Gong, Y., 2003. Document-clustering based on non-negative matrix factorization. In: Proceedings of SIGIR'03, July 28–August 1, Toronto, CA, pp. 267–273.

Zangwill, W., 1967. Minimizing a function without calculating derivatives. Comput. J. 10, 293–296.

Zdunek, R., Cichocki, A., 2006. Non-negative matrix factorization with quasi-newton optimization. In: Proceedings of the Eighth International Conference on Artificial Intelligence and Soft Computing, ICAISC, Zakopane, Poland, June 25–29.