NUMERIC METHODS FOR NONNEGATIVE MATRIX FACTORIZATION *

LARSON HOGSTROM†

Abstract. This report presents a framework and analysis to compare three common methods used to compute nonnegative matrix factorization (NMF). The parts-based representation obtained through NMF has become a popular approach for dimensionality reduction in a variety of scientific and technical domain. This study focuses on the algorithms and numerical aspects of NMF computation. We review the derivation and continual descent properties of multiplicative update rules which helped to first popularized the NMF approach. Alternating least squares and projection gradient methods are also presented as faster, more recent approaches. The accuracy and convergence of these algorithms is compared using simulated data and two of the methods were applied to a database gene expression profiles measuring drug responses in the MCF7 cancer cell line. Lastly, the objective function of alternating least squares was directly modified to impose sparsity during matrix factorization. The fraction of near-zero entries was compared before and after ALS modification as sparsity results are of known interest to many researchers using NMF.

Key words. dimensionality reduction, matrix factorization, least squares, nonnegativity, data representation

1. Introduction. Nonnegative matrix factorization (NMF) is a dimensionality reduction technique that can be applied in a variety of domains. Like principal component analysis (PCA), NMF seeks to describe a large matrix of data using a small set of information dense components. Unlike PCA, which imposes orthogonality constraints on components, NMF requires nonnegativity after factorization. This procedure allows for only additive combinations of positive components to estimate the input matrix. The endpoint is a parts-based representation of the starting data where each component is more easily interpreted than PCA.

A number of algorithms have been proposed in order to construct the NMF factorization with positivity constraints. Iterative algorithms, for example, build factorized components while converging on a local minimum of a specified object function.

This project compares multiple algorithms for computing NMF factorization including gradient and non gradient methods. The motivation fixed-point type methods minimization problem with bounded constraints

This project will examine two of such objective functions 1) the conventional least squares difference between the input and the factorization and 2) an objective function based on the Kullback-Leibler divergence2. The flop counts of algorithmic update rules for both approaches will both be assessed analytically and the different conditions for monotonic convergence will be examined.

For the purposes of this study, algorithms are implemented and evaluated in the python language using the 'numpy' and 'scipy' packages for scientific computation. In simulations, the algorithms are compared by their accuracy after a fixed number of iterations. To illustrate real-world applications of NMF, non-negative low-rank approximations were used to classify gene expression profiles of known pharmacological groups. This proceeding report is organized in the following structure. Section 2 defines the NMF approach and reviews commonly used objective functions. Section 3 outlines the multiplicative update rule which is one of the oldest and most widely used procedures to perform matrix decomposition with NMF. Section 4 covers the alter-

*A final project for 18.335, Spring 2015

† (hogstrom@mit.edu).

nating least squares approach which variations with better characterized convergence properties. This is followed by a discussion of sparsity and initialization conditions.

2. Problem Definition and Objective Functions. DEFINITION 2.1. *For a nonnegative matrix* $\boldsymbol{A} \in \Re^{mxn}$*, select a low-rank approximation of size k such that there are two nonnegative matrices* $\boldsymbol{W} \in \Re^{mxk}$ *and* $\boldsymbol{H} \in \Re^{kxn}$ *which minimizes a function such as*

$$f(\boldsymbol{W},\boldsymbol{H}) = \frac{1}{2}||\boldsymbol{A} - \boldsymbol{WH}||_F^2 \tag{2.1}$$

KKT optimality conditions a stationary point for

Other commonly used objective functions include Euclidian distance and Kullback-Leibler (KL) divergence. KL can be extended to a more general information-based framework using Renyi's divergence. (Devarajan, 2005). Here, a single parameter α is used to represent a continuum of distance measures and KL airises as a special case as $\alpha \to 1$.

$$KL(A||WH) = \sum_{ij} [A_{ij} \log \frac{A_{ij}}{(WH)_{ij}} - A_{ij} + (WH)_{ij}] \tag{2.2}$$

For its simplicity and widespread use, this report will focus on objective function defined in (2.1).

Regardless of the objective used to obtain them, the resulting W and H matrices are often interpreted directly. The matrix $\mathbf{W} \in \Re^{mxk}$ typically represents the k different components of signal in the original matrix. The $\mathbf{H} \in \Re^{kxn}$ matrix can be interpreted as weighting factors defining the linear combination of k components used to reconstruct a sample in the original data matrix. In the application of NMF to gene expression data for example, each of the k columns of W represent a 'metagene' response. A small number of these metagene columns can be combined to estimate an original experimental column found in the original data matrix. The H matrix defines the weighting of each metagene.

3. Multiplicative Update Rule. In their seminal paper, Lee and Seung began their analysis with a simple additive update rule for H that reduces the squared distance:

$$H_{aj} \leftarrow H_{aj} + \eta_{aj}[(W^T A)_{aj} - (W^T WH)_{aj}] \tag{3.1}$$

The authors noted that this update represents a typical gradient descent rule if each of the η_{au} terms are all equally set to a small positive number. Alternatively, the variables can be diagonally rescaled by setting

$$\eta_{aj} = \frac{H_{aj}}{(W^T WH)_{aj}} \tag{3.2}$$

and when placed into equation (3.1), this motivates the multiplicative update rules that seek to minimize the euclidian distance $||A - WH||$:

$$H_{aj} \leftarrow H_{aj} \frac{(W^T A)_{aj}}{(W^T WH)_{aj}} \qquad W_{ia} \leftarrow W_{ia} \frac{(W^T A)_{ia}}{(W^T WH)_{ia}} \tag{3.3}$$

Where the same logic for H is also applied to the update of W. Making use of auxiliary functions, Lee and Seung went on to demonstrate the non-increasing properties

of NMF under the multiplicative update rules. It has been noted that they incorrectly claimed that algorithm converges to a local minimum (Chu, 2004; Gonzalez and Zhang, 2005, Lin, 2005b). It is certainly possible that continual descent property of multiplicative updates could descend to a saddle point (Berry, 2007). Furthermore, if $W_{ia} = 0$ after the kth iteration, then this entry remains $W_{ia} = 0$ for all subsequent iterations. This property is undesirable because the algorithm could theoretically remain fixed in a non-optimal descent path.

Data: Input data matrix: $\mathbf{A} \in \Re^{\mathbf{mxn}}$
Result: nonnegative factorization of **A** using k components, creating matrices $\mathbf{W} \in \Re^{mxk}$ and $\mathbf{H} \in \Re^{kxn}$
initialization;
$\mathbf{W} \leftarrow$ random dense (m x k) matrix
$\mathbf{H} \leftarrow$ random dense (k x n) matrix
for *i = 1 to maxiter* **do**
  $\mathbf{H} = \mathbf{H}. * (\mathbf{W^T A})./(\mathbf{W^T W H})$
  $\mathbf{W} = \mathbf{W}. * (\mathbf{A H^T})./(\mathbf{W H H^T})$
end

Algorithm 1: Multiplicative update

It is important to note that this approach will breakdown if a zero appears in the denominator. In practice, it is common to add a very small floating point value to the denominator during each iteration. The required work of O(mnk) per iteration is quite high compared to other exiting algorithms and multiplicative update tends to converge slowly for real world problems. Nevertheless, with its early establishment and its straightforward implementation this algorithm has maintained popularity for NMF computation.

4. Alternating Least Squares. The alternating least squares can be viewed as a special case of block coordinate descent method. For this class of algorithms, the objective function is minimized with respect to coordinate vectors x_i^k. Multidimensional vectors are minimized in cyclical order, with one block coordinate vector minimized during each iteration (Berstekas, 1999). A single iteration takes the form:

$$x_i^{k+1} \in \text{argmin}_{\varepsilon \in X_i} = f(x_1^{k+1}, ..., x_{i-1}^{k+1}, \varepsilon, x_{i+1}^k, ..., x_n^k) \tag{4.1}$$

For the case of NMF, this scenario is especially simple as there are only two block variables, W and H. Sequentially one matrix is fixed and the other is improved.

$$\begin{aligned} W^{k+1} &= \text{argmin}_{W \geq 0} f(W, H^k) \\ H^{k+1} &= \text{argmin}_{H \geq 0} f(W^{k+1}, H) \end{aligned} \tag{4.2}$$

This takes advantage of the fact that while objective (2.1) is not convex in both W and H, it is convex in either W or H individually. One basic least ALS algorithm to accomplish this outlined by Berry et al (Berry, 2006). This strategy imposes nonnegativity by setting all nonzero entries of W and H to zero during each iteration. Unlike multiplicative update rules, however these zero entries are not forced to remain at 0 for all proceeding iterations.

Data: Input data matrix: $\mathbf{A} \in \Re^{\mathbf{mxn}}$
Result: nonnegative factorization of **A** using k components, creating matrices $\mathbf{W} \in \Re^{mxk}$ and $\mathbf{H} \in \Re^{kxn}$
initialization;
$\mathbf{W} \leftarrow$ random dense (m x k) matrix
$\mathbf{H} \leftarrow$ random dense (k x n) matrix
for *i = 1 to maxiter* **do**
  Solve for **H** in $\mathbf{W^T W H = W^T A}$
  set negative elements in **H** to 0
  Solve for **W** in $\mathbf{H H^T W^T = H A^T}$
  set negative elements in **W** to 0
end

Algorithm 2: Alternating least squares

Each subproblem from in this algorithm is solved in an unconstrained manner and the resulting W and H solutions are modified during each iteration to satisfy the desired non-negativty constraints. The simplicity of setting negative values in W and H to zero provides for an especially fast approach as demonstrated in the comparisons of this study. Theoretical evaluation of this algorithm's convergence properties, however, are difficult because each subproblem is formulated as an unconstrained least squares problem, but the solutions are directly modified and therefore do not map onto the original formulations. Other problem definitions allow for a framework to better understand the convergence properties of the two block coordinate descent method. Alternating non-negativity constrained least squares (ANLS) for instance provides a formulation that directly incorporates non-negative constraints in each block coordinate subproblem (Kim, 2006). Adding the $W \geq 0$ and $H \geq 0$ constraints onto equations () and () respectively, the Karush-Kuh-Tucker (KK) optimality conditions can be used to define stationary points of the objective function (2.1) iif

$$\begin{aligned} W &\geq 0, \\ \bigtriangledown_W f(W,H) = WHH^T - AH^T &\geq 0, \\ W. * \bigtriangledown_W f(W,H) &= 0 \end{aligned} \tag{4.3}$$

$$\begin{aligned} H &\geq 0, \\ \bigtriangledown_H f(W,H) = W^T WH - W^T A &\geq 0, \\ H. * \bigtriangledown_H f(W,H) &= 0 \end{aligned}$$

For the block coordinate descent algorithms, it has been shown that the limit point of a sequence of sub-blocks is a stationary point if each subproblem has a unique solution (Bertsekas, 1999). Unfortunately, ANLS subproblems are not unique as there exists matrices in the form $\mathbf{X} \in \Re^{kxk}$ which represent scaling and permutation that satisfy $||\mathbf{A - WH}||_F = ||\mathbf{A - WXX^{-1}H}||_F$ (Kim, 2006). In the case of two bock problems, however, stationary points can be found as the limit point of the sequence of two sub-block solutions (Grippo, 2000). One classical approach to solving the two block subploblems is by way of an *active set algorithm* (Lawson, 1974). Here, the difference between a input matrix $Y \in \Re^{lxp}$ and its resultant factorization cab be

split into vectors:

$$\min_{G\geq 0} ||BG - Y||_F^2 \rightarrow \min_{G\geq 0} ||B\mathbf{g}_1 - \mathbf{y}_1||_2^2, ..., \min_{G\geq 0} ||B\mathbf{g}_l - \mathbf{y}_m||_2^2 \tag{4.4}$$

where $B \in \Re^{lxm}$ and $y_i \in \Re^{lx1}$. Since there are m inequity constraints after the split, the ith constraint will be active if the ith regression coefficient is set to zero. Thus if the active set is known, active constraints can be treated as equality constraints instead of inequality constraints. The overall least square problem can be solved by examining only the variables in the passive set and calculating their unconstrained least squares solution (Bro, 1997). The ANLS algorithm begins with an initial feasible set of regression coefficients. The zero vector of coefficients is often used as an initial feasible set as no constraints are violated. ANLS proceeds by iteratively removing variables from the active set until the true active set is found. The variables removed from the active set are used in an unconstrained linear regression to calculate the block coordinate descent solution. While the convergence properties of ANLS are better understood, this report omits this strategy in comparative analysis due to the complexity of the algorithm and its slow computation. Recent work has succeeded in improving ANLS speed by precomputing cross-product terms during the unconstrained linear regression calculations (Bro, 1997).

5. Sparsity in Alternating least squares. Added penalties to objective function to encourage sparse solutions.

6. Projection gradient method.

7. Applications to gene expression. A classical example of NMF is the factorization of a large database of human face images. Factorization using PCA will yield a series of ?eigenfaces? which can be added or subtracted, but individual components are not easy to interpret on their own. The NMF "parts-based" representation however, yields components like noses, eyes, ears, and mouths which can be added linearly to construct a face. Parts-based representations have been helpful in identifying and interpreting patterns in a number of biological contexts.

NMF has shown to be less sensitive to *a priori* gene selection or initial conditions when identifying context-dependent patterns of gene expression (Brunet, 2004).

The same properties have been found to be helpful in a number of biological contexts.

The similar results found with ALS and gradient projection, point to the importance of other factors. The effects of sparsity were not examined with this particular dataset, but could have a significantly influence on the interpretation of the NMF results. Other factors, including smoothness, $^{and}{}_{couldalsoimpactthe}$

8. Initialization conditions.

9. Additional computational issues. -Sparsity

10. Conclusion. Despite the simplicity of defining nonnegative matrix factorization, the landscape of numeric methods used to address the problem remains complex. Recent efforts have sought to create new NMF algorithms with well defined convergence properties. Methods such as alternating non-negativity constrained least squares (ANLS) have succeeded in improving our analytical understanding of NMF convergence, but remain slow and cumbersome when compared to simpler methods that appear to converge in most real-world scenarios. When evaluating NMF tools,

researchers should be guided by the parameters of their data and the research question of interest. For especially large datasets the speed and simplicity ALS might serve as a reasonable starting point for NMF calculation. Researchers with modest datasets, might prefer . *The results presented in this report also show that researchers can manipulate the sparsity of factorization by adopting specialized objective functions.*

REFERENCES

[1] Paatero, P., *Positive matrix factorization; a non-negative factor model with optimal utilization of error estimates of data values*, Environmetrics (1994) 5, 111-126.

[2] Paatero, P., *Least squares formulation of robust non-negative factor analysis*, Chemometrics and Intell. Laboratory Syst. (1997) 37, 23-35.

[3] Paatero, P., *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. (1999) 8, 1-35.

[4] D. Lee, H. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature (1999) 401, 788-791.

[5] D. Lee, H. Seung, *Algorithms for Nonnegative Matrix Factorization*, Advances in Neural Information Processing Systems (2001) 13, 556-562.

[6] M. Berry, M. Browne, A. Langville, P. Pauca, R. Plemenos, *Algorithms and applications for approximate nonnegative matrix factorization*, Computational Statistics and Data Analysis 52 (2007) pp. 155 - 173.

[7] A. Cichocki, R. Zdunek, A. Huy Phan, S. Amari, *Nonnegative Matrix and Tensor Factorizations*, Wiley, Natick, MA, 2009.