

Near-separable Non-negative Matrix Factorization with ℓ_1 and Bregman Loss Functions

Abhishek Kumar*

Vikas Sindhwani†

Abstract

Recently, a family of tractable NMF algorithms have been proposed under the assumption that the data matrix satisfies a separability condition (Donoho & Stodden, 2003; Arora et al., 2012). Geometrically, this condition reformulates the NMF problem as that of finding the extreme rays of the conical hull of a finite set of vectors. In this paper, we develop separable NMF algorithms with ℓ_1 loss and Bregman divergences, by extending the conical hull procedures proposed in our earlier work (Kumar et al., 2013). Our methods inherit all the advantages of (Kumar et al., 2013) including scalability and noise-tolerance. We show that on foreground-background separation problems in computer vision, robust near-separable NMFs match the performance of Robust PCA, considered state of the art on these problems, with an order of magnitude faster training time. We also demonstrate applications in exemplar selection settings.

1 Introduction

The problem of non-negative matrix factorization (NMF) is to express a non-negative matrix \mathbf{X} of size $m \times n$, either exactly or approximately, as a product of two non-negative matrices, \mathbf{W} of size $m \times r$ and \mathbf{H} of size $r \times n$. Approximate NMF attempts to minimize a measure of divergence between the matrix \mathbf{X} and the factorization \mathbf{WH} . The inner-dimension of the factorization r is usually taken to be much smaller than m and n to get interpretable part-based representation of data (Lee & Seung, 1999). NMF is used in a wide range of applications, e.g., topic modeling and text mining, hyper-spectral image analysis, audio source separation, and microarray data analysis (Cichocki et al., 2009).

The exact and approximate NMF problem is NP-hard. Hence, traditionally, algorithmic work in NMF has focused on treating it as an instance of non-convex optimization (Cichocki et al., 2009; Lee & Seung, 1999; Lin, 2007; Hsieh & Dhillon, 2011) leading to algorithms lacking optimality guarantees beyond convergence to

a stationary point. Promising alternative approaches have emerged recently based on a *separability* assumption on the data (Esser et al., 2012; Arora et al., 2012; Bittorf et al., 2012; Gillis & Vavasis, 2012; Kumar et al., 2013; Gillis, 2014; Gillis & Luce, 2014) which enables the NMF problem to be solved efficiently and exactly. Under this assumption, the data matrix \mathbf{X} is said to be r -separable if all columns of \mathbf{X} are contained in the conical hull generated by a subset of r columns of \mathbf{X} . In other words, if \mathbf{X} admits a factorization \mathbf{WH} then the separability assumption states that the columns of \mathbf{W} are present in \mathbf{X} at positions given by an unknown index set A of size r . Equivalently, the corresponding columns of the right factor matrix \mathbf{H} constitute the $r \times r$ identity matrix, i.e., $\mathbf{H}_A = \mathbf{I}$. We refer to these columns indexed by A as *anchor columns*.

The separability assumption was first investigated by Donoho & Stodden (2003) in the context of deriving conditions for uniqueness of NMF. NMF under separability assumption has been studied for topic modeling in text (Kumar et al., 2013; Arora et al., 2013) and hyper-spectral imaging (Gillis & Vavasis, 2012; Esser et al., 2012), and separability has turned out to be a reasonable assumption in these two applications. In the context of topic modeling where \mathbf{X} is a document-word matrix and \mathbf{W} , \mathbf{H} are document-topic and topic-word associations respectively, it translates to assuming that there is at least one word in every topic that is unique to itself and is not present in other topics.

Our starting point in this paper is the family of conical hull finding procedures called XRAY introduced in our earlier work (Kumar et al., 2013) for near-separable NMF problems with Frobenius norm loss. XRAY finds anchor columns one after the other, incrementally expanding the cone and using exterior columns to locate the next anchor. XRAY has several appealing features: (i) it requires no more than r iterations each of which is parallelizable, (ii) it empirically demonstrates noise-tolerance, (iii) it admits efficient model selection, and (iv) it does not require normalizations or preprocessing needed in other methods. However, in the presence of outliers or different noise characteristics, the use of Frobenius norm approximations is not optimal.

*IBM TJ Watson Research Center, Yorktown Heights, NY. Email: abhishk@us.ibm.com

†Google Research, NY. Email: sindhwani@google.com. This work was done when the author was at IBM Research.



Figure 1: RobustXRAY applied to video background-foreground separation problem

In fact, none of the existing near-separable NMF algorithms works with ℓ_1 and Bregman loss functions. On the other hand, there exist local search based NMF algorithms for ℓ_1 loss and Bregman divergences (Sra & Dhillon, 2005; Guan et al., 2012).

In this paper, we fill this gap and extend XRAY to provide robust factorizations with respect to ℓ_1 loss, and approximations with respect to the family of Bregman divergences. Figure 1 shows a motivating application from computer vision. Given a sequence of video frames, the goal is to separate a near-stationary background from the foreground of moving objects that are relatively more dynamic across frames but span only a few pixels. In this setting, it is natural to seek a low-rank background matrix \mathbf{B} that minimizes $\|\mathbf{X} - \mathbf{B}\|_1$ where \mathbf{X} is the frame-by-pixel video matrix, and the ℓ_1 loss imposes a sparsity prior on the residual foreground. Unlike the case of low-rank approximations in Frobenius or spectral norms, this problem does not admit an SVD-like tractable solution. The Robust Principal Component Analysis (RPCA), considered state of the art for this application, uses a nuclear-norm convex relaxation of the low-rank constraints. In this paper, we instead recover tractability by imposing the separable NMF assumption on the background matrix. This implies that the variability of pixels across the frames can be "explained" in terms of observed variability in a small set of anchor pixels. Under a more restrictive setting, this can be shown to be equivalent to median filtering on the video frames, while a full near-separable NMF model imparts more degrees of freedom to model the background. We show that the proposed near-separable NMF algorithms with ℓ_1 loss are competitive with RPCA in separating foreground from background while outperforming it in terms of computational efficiency.

Our algorithms are empirically shown to be robust to noise (deviations from the pure separability assumption). In addition to the background-foreground problem, we also demonstrate our algorithms on the exemplar selection problem. For identifying exemplars in a data set, the proposed algorithms are evaluated on text documents with classification accuracy as a performance metric and are shown to outperform the recently proposed method of Elhamifar et al. (2012).

Related Work: Existing separable NMF methods work either with only a limited number of loss functions on the factorization error such as Frobenius norm loss (Kumar et al., 2013), $\ell_{1,\infty}$ norm loss (Bittorf et al., 2012), or maximize proxy criteria such as volume of the convex polyhedron with anchor columns as vertices (Gillis & Vavasis, 2012) and distance between successive anchors (Arora et al., 2013) to select the anchor columns. On the other hand, local search based NMF methods (Cichocki et al., 2009) have been proposed for a wide variety of loss functions on the factorization error including ℓ_1 norm loss (Guan et al., 2012; Kim & Park, 2007) and instances of Bregman divergence (Li et al., 2012; Sra & Dhillon, 2005). In this paper, we close this gap and propose algorithms for near-separable NMF that minimize ℓ_1 loss and Bregman divergence for the factorization.

2 Geometric Intuition

The goal in exact NMF is to find a matrix \mathbf{W} such that the cone generated by its columns (i.e., their non-negative linear combinations) contains all columns of \mathbf{X} . Under separability assumption, the columns of matrix \mathbf{W} are to be picked directly from \mathbf{X} , also known as anchor columns. The algorithms in this paper build the cone incrementally by picking a column from \mathbf{X} in every iteration. The algorithms execute r such iterations for constructing a factorization of inner-dimension r . Figure 2 shows the cone after three iterations of the algorithm when three anchor columns have been identified. An extreme ray $\{t\mathbf{x} : t > 0\}$ is associated with every anchor point \mathbf{x} . The points on an extreme ray cannot be expressed as conic combinations of other points in the cone that do not themselves lie on that extreme ray. To identify the next anchor column, the algorithm picks a point outside the current cone (a green point) and projects it to the current cone so that the distance between the point and the projection is minimized in terms of the desired measure of distance. This projection is then used to setup a specific simple criteria which when maximized over the data points, identifies a new anchor. This new anchor is then added to the current set of anchors and the cone is expanded iteratively until all anchors have been picked.

These geometric intuitions are inspired by Clarkson (1994); Dula et al. (1998) who present linear programming (LP) based algorithms for general convex and conical hull problems. Their algorithms use ℓ_2 projections of exterior points to the current cone and are also applicable in our NMF setting if the data matrix \mathbf{X} satisfies r -separability exactly. In this case, the ℓ_2 projection and corresponding residual vector of *any* single exterior point can be used to expand the cone and *all* r anchors

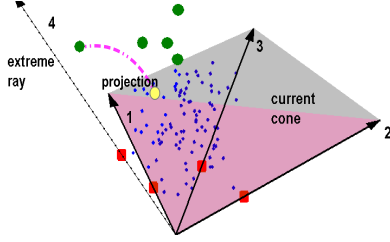


Figure 2: Geometrical illustration of the algorithm

will be recovered correctly at the end of the algorithm. When \mathbf{X} does not satisfy r -separability exactly, anchor selection criteria derived from *multiple* residuals demonstrate superior noise robustness as empirically shown by Kumar et al. (2013) who consider the case of Gaussian i.i.d. noise. However, the algorithms of Kumar et al. (2013) are not suitable for noise distributions other than Gaussian (e.g., other members of the exponential family, sparse noise) as they minimize $\|\mathbf{X} - \mathbf{X}_A \mathbf{H}\|_F^2$. In the following sections, we present algorithms for near-separable NMF that are targeted precisely towards this goal and empirically demonstrate their superiority over existing algorithms under different noise distributions.

3 Near-separable NMF with ℓ_1 loss

This section considers the case when the pure separable structure is perturbed by sparse noise. Hence our aim is to minimize $\|\mathbf{X} - \mathbf{X}_A \mathbf{H}\|_1$ for $\mathbf{H} \geq 0$ where $\|\cdot\|_1$ denotes element-wise ℓ_1 norm of the matrix and \mathbf{X}_A are the columns of \mathbf{X} indexed by set $A \subset \{1, 2, \dots, n\}$. We denote i th column of \mathbf{X} by \mathbf{X}_i . The proposed algorithm proceeds by identifying one anchor column in each iteration and adding it to the current set of anchors, thus expanding the cone generated by anchors. Each iteration consists of two steps: (i) *anchor selection* step that finds the column of \mathbf{X} to be added as an anchor, and (ii) a *projection* step where all data points (columns of \mathbf{X}) are projected to the current cone in terms of minimizing the ℓ_1 norm. Algorithm 1 outlines the steps of the proposed algorithm.

Selection Step: In the selection step, we normalize all the points to lie on the hyperplane $\mathbf{p}^T \mathbf{x} = 1$ ($\mathbf{Y}_j = \frac{\mathbf{X}_j}{\mathbf{p}^T \mathbf{X}_j}$) for a strictly positive vector \mathbf{p} and evaluate the selection criterion of Eq. 3.3 to select the next anchor column. Note that any exterior point ($i : \|\mathbf{R}_i\|_1 \geq 0$) can be used in the selection criterion – Algorithm 1 shows two possibilities for choosing the exterior point. Taking the point with maximum residual ℓ_1 norm to be the exterior point turns out to be far more robust to noise than randomly choosing the exterior point, as observed in our numerical simulations.

Algorithm 1 RobustXRAY: Near-separable NMF with ℓ_1 loss

Input: $\mathbf{X} \in \mathbb{R}_+^{m \times n}$, inner dimension r

Output: $\mathbf{W} \in \mathbb{R}_{m \times r}$, $\mathbf{H} \in \mathbb{R}_{r \times n}$, r indices in A
such that: $\mathbf{X} = \mathbf{W}\mathbf{H}$, $\mathbf{W} = \mathbf{X}_A$

Initialize: $\mathbf{R} \leftarrow \mathbf{X}$, $\mathbf{D}^* \leftarrow \mathbf{X}$, $A \leftarrow \{\}$

while $|A| < r$ **do**

1. **Anchor Selection step:**

First, pick any point exterior to the current cone. Two possible criteria are

$$(3.1) \quad \text{rand} : \quad \text{any random } i : \|\mathbf{R}_i\|_1 > 0$$

$$(3.2) \quad \text{max} : \quad i = \arg \max_k \|\mathbf{R}_k\|_1$$

Choose a suitable $\mathbf{D}_i^* \in \mathbf{D}_i$ where $\mathbf{D}_{ji} = \text{sign}(\mathbf{R}_{ji})$ if $\mathbf{R}_{ji} \neq 0$, else $\mathbf{D}_{ji} \in [-1, 1]$ (see Remark (1)).

Select an anchor as follows (\mathbf{p} is a strictly positive vector, not collinear with \mathbf{D}_i^* (see Remark (3))):

$$(3.3) \quad j^* = \arg \max_j \frac{\mathbf{D}_i^{*T} \mathbf{X}_j}{\mathbf{p}^T \mathbf{X}_j}$$

2. Update: $A \leftarrow A \cup \{j^*\}$ (see Remark (2))

3. **Projection step:** Project onto current cone.

$$(3.4) \quad \mathbf{H} = \arg \min_{\mathbf{B} \geq 0} \|\mathbf{X} - \mathbf{X}_A \mathbf{B}\|_1 \quad (\text{ADMM})$$

4. Update Residuals: $\mathbf{R} = \mathbf{X} - \mathbf{X}_A \mathbf{H}$

end while

Projection Step: The projection step, Eq. 3.4, involves solving a multivariate least absolute deviations problems with non-negativity constraints. We use alternating direction method of multipliers (ADMM) (Boyd et al., 2011) and reformulate the problem as

$$\min_{\mathbf{B} \geq 0, \mathbf{Z}} \|\mathbf{Z}\|_1, \text{ such that } \mathbf{X}_A \mathbf{B} + \mathbf{Z} = \mathbf{X}.$$

Thus the non-negativity constraints are decoupled from the ℓ_1 objective and the ADMM optimization proceeds by alternating between two sub-problems – a standard ℓ_1 penalized ℓ_2 proximity problem in variable \mathbf{Z} which has a closed form solution using the soft-thresholding operator, and a non-negative least squares problem in variable \mathbf{B} that is solved using a cyclic coordinate descent approach (cf. Algorithm 2 in (Kumar et al., 2013)). The standard primal and dual residuals based criteria is used to declare convergence (Boyd et al., 2011). The ADMM procedure converges to the global optimum since the problem is convex.

We now show that Algorithm 1 correctly identifies all the anchors in pure separable case.

LEMMA 3.1. *Let \mathbf{R} be the residual matrix obtained after ℓ_1 projection of columns of \mathbf{X} onto the current cone*

and \mathbf{D} be the set of matrices such $\mathbf{D}_{ij} = \text{sign}(\mathbf{R}_{ij})$ if $\mathbf{R}_{ij} \neq 0$ else $\mathbf{D}_{ij} \in [-1, 1]$. Then, there exists at least one $\mathbf{D}^* \in \mathbf{D}$ such that $\mathbf{D}^{*T} \mathbf{X}_A \leq 0$, where \mathbf{X}_A are anchor columns selected so far by Algorithm 1.

Proof. Residuals are given by $\mathbf{R} = \mathbf{X} - \mathbf{X}_A \mathbf{H}$, where $\mathbf{H} = \arg \min_{\mathbf{B} \geq 0} \|\mathbf{X} - \mathbf{X}_A \mathbf{B}\|_1$.

Forming the Lagrangian for Eq. 3.4, we get $\mathcal{L}(\mathbf{B}, \mathbf{\Lambda}) = \|\mathbf{X} - \mathbf{X}_A \mathbf{B}\|_1 - \text{tr}(\mathbf{\Lambda}^T \mathbf{B})$, where the matrix $\mathbf{\Lambda}$ contains the non-negative Lagrange multipliers. The Lagrangian is not smooth everywhere and its sub-differential is given by $\partial \mathcal{L} = -\mathbf{X}_A^T \mathbf{D} - \mathbf{\Lambda}$ where \mathbf{D} is as defined in the lemma. At the optimum $\mathbf{B} = \mathbf{H}$, we have $0 \in \partial \mathcal{L} \Rightarrow -\mathbf{\Lambda} \in \mathbf{X}_A^T \mathbf{D}$. Since $\mathbf{\Lambda} \geq 0$, this means that there exists at least one $\mathbf{D}^* \in \mathbf{D}$ for which $\mathbf{D}^{*T} \mathbf{X}_A \leq 0$.

LEMMA 3.2. *For any point \mathbf{X}_i exterior to the current cone, there exists at least one $\mathbf{D}^* \in \mathbf{D}$ such that it satisfies the previous lemma and $\mathbf{D}_i^{*T} \mathbf{X}_i > 0$.*

Proof. Let $\mathbf{R} = \mathbf{X} - \mathbf{X}_A \mathbf{H}$, where $\mathbf{H} = \arg \min_{\mathbf{B} \geq 0} \|\mathbf{X} - \mathbf{X}_A \mathbf{B}\|_1$ and \mathbf{X}_A are the current set of anchors. From the proof of previous lemma, $-\mathbf{\Lambda}^T \in \mathbf{D}^T \mathbf{X}_A$. Hence, $-\mathbf{\Lambda}_i^T \in \mathbf{D}_i^T \mathbf{X}_A$ (i th row of both left and right side matrices). From the complementary slackness condition, we have $\mathbf{\Lambda}_{ji} \mathbf{H}_{ji} = 0 \forall j, i$. Hence, $-\mathbf{\Lambda}_i^T \mathbf{H}_i = 0 \in \mathbf{D}_i^T \mathbf{X}_A \mathbf{H}_i$.

Since all KKT conditions are met at the optimum, there is at least one $\mathbf{D}^* \in \mathbf{D}$ that satisfies previous lemma and for which $\mathbf{D}_i^{*T} \mathbf{X}_A \mathbf{H}_i = 0$. For this \mathbf{D}^* , we have $\mathbf{D}_i^{*T} \mathbf{X}_i = \mathbf{D}_i^{*T} (\mathbf{R}_i + \mathbf{X}_A \mathbf{H}_i) = \mathbf{D}_i^{*T} \mathbf{R}_i = \|\mathbf{R}_i\|_1 > 0$ since $\mathbf{R}_i \neq 0$ for an exterior point.

Using the above two lemmas, we prove the following theorem regarding the correctness of Algorithm 1 in pure separable case.

THEOREM 3.1. *If the maximizer in Eq. 3.3 is unique, the data point \mathbf{X}_{j^*} added at each iteration in the Selection step of Algorithm 1, is an anchor that has not been selected in one of the previous iterations.*

Proof. Let the index set A denote all the anchor columns of \mathbf{X} . Under the separability assumption, we have $\mathbf{X} = \mathbf{X}_A \mathbf{H}$. Let the index set A^t identify the current set of anchors.

Let $\mathbf{Y}_j = \frac{\mathbf{X}_j}{\mathbf{p}^T \mathbf{X}_j}$ and $\mathbf{Y}_A = \mathbf{X}_A [\text{diag}(\mathbf{p}^T \mathbf{X}_A)]^{-1}$ (since \mathbf{p} is strictly positive, the inverse exists). Hence $\mathbf{Y}_j = \mathbf{Y}_A \frac{[\text{diag}(\mathbf{p}^T \mathbf{X}_A)] \mathbf{H}_j}{\mathbf{p}^T \mathbf{X}_j}$. Let $\mathbf{C}_j = \frac{[\text{diag}(\mathbf{p}^T \mathbf{X}_A)] \mathbf{H}_j}{\mathbf{p}^T \mathbf{X}_j}$. We also have $\mathbf{p}^T \mathbf{Y}_j = 1$ and $\mathbf{p}^T \mathbf{Y}_A = \mathbf{1}^T$. Hence, we have $1 = \mathbf{p}^T \mathbf{Y}_j = \mathbf{p}^T \mathbf{Y}_A \mathbf{C}_j = \mathbf{1}^T \mathbf{C}_j$.

Using Lemma 3.1, Lemma 3.2 and the fact that \mathbf{p} is strictly positive, we have $\max_{1 \leq j \leq n} \mathbf{D}_i^{*T} \mathbf{Y}_j =$

$\max_{j \notin A^t} \mathbf{D}_i^{*T} \mathbf{Y}_j$. Indeed, for all $j \in A^t$ we have $\mathbf{D}_i^{*T} \mathbf{Y}_j \leq 0$ using Lemma 3.1 and there is at least one $j = i \notin A^t$ for which $\mathbf{D}_i^{*T} \mathbf{Y}_j > 0$ using Lemma 3.2. Hence the maximum lies in the set $\{j : j \notin A^t\}$.

Further, we have $\max_{j \notin A^t} \mathbf{D}_i^{*T} \mathbf{Y}_j = \max_{j \notin A^t} \mathbf{D}_i^{*T} \mathbf{Y}_A \mathbf{C}_j = \max_{j \in (A \setminus A^t)} \mathbf{D}_i^{*T} \mathbf{Y}_j$. The second equality is the result of the fact that $\|\mathbf{C}_j\|_1 = 1$ and $\mathbf{C}_j \geq 0$. This implies that if there is a unique maximum at a $j^* = \arg \max_{j \notin A^t} \mathbf{D}_i^{*T} \mathbf{Y}_j$, then \mathbf{X}_{j^*} is an anchor that has not been selected so far.

Remarks:

(1) For the correctness of Algorithm 1, the anchor selection step requires choosing a $\mathbf{D}_i^* \in \mathbf{D}_i$ for which Lemma 3.1 and Lemma 3.2 hold true. Here we give a method to find one such \mathbf{D}_i^* using linear programming. Using KKT conditions, the \mathbf{D}_i^* satisfying $-\mathbf{X}_A^T \mathbf{D}_i^* = \mathbf{\Lambda}_i \in \mathbb{R}_+^{|A|}$ is a candidate. We know $\mathbf{\Lambda}_{ji} = 0$ if $\mathbf{H}_{ji} > 0$ and $\mathbf{\Lambda}_{ji} > 0$ if $\mathbf{H}_{ji} = 0$ (complementary slackness). Let $Z = \{j : \mathbf{H}_{ji} > 0\}$ and $\tilde{Z} = \{j : \mathbf{H}_{ji} = 0\}$. Let $I = \{j : \mathbf{R}_{ji} = 0\}$. Let \mathbf{u} represent the elements of \mathbf{D}_i^* that we need to find, i.e., $\mathbf{u} = \{\mathbf{D}_{ji}^* : j \in I\}$. Finding \mathbf{u} is a feasibility problem that can be solved using an LP. Since there can be multiple feasible points, we can choose a dummy cost function $\sum_k \mathbf{u}_k$ (or any other random linear function of \mathbf{u}) for the LP. More formally, the LP takes the form:

$$\begin{aligned} & \min_{-1 \leq \mathbf{u} \leq 1} \mathbf{1}^T \mathbf{u}, \quad \text{such that} \\ & -\mathbf{X}_A^T \mathbf{D}_i^* = \mathbf{\Lambda}_i, \mathbf{\Lambda}_{ji} = 0 \forall j \in Z, \mathbf{\Lambda}_{ji} > 0 \forall j \in \tilde{Z} \end{aligned}$$

In principle, the number of variables in this LP is the number of zero entries in residual vector \mathbf{R}_i which can be as large as $m - 1$. In practice, we always have the number of zeros in \mathbf{R}_i much less than m since we always pick the exterior point with maximum ℓ_1 norm in the Anchor Selection step of Algorithm 1. The number of constraints in the LP is also very small ($= |A| < r$). In our implementation, we simply set $\mathbf{u} = -1$ which, in practice, almost always satisfies Lemma 3.1 and Lemma 3.2. The LP is called whenever Lemma 3.2 is violated which happens rarely (note that Lemma 3.1 will never violate with this setting of \mathbf{u}).

(2) If the maximum of Eq. 3.3 occurs at more than one point, it is clear from the proof of Theorem 3.1 (last paragraph in the proof) that at least two anchors that are not selected so far should attain the maximum, i.e., $\max_{j \notin A^t} \mathbf{D}_i^{*T} \mathbf{Y}_j = \max_{j \in (A \setminus A^t)} \mathbf{D}_i^{*T} \mathbf{Y}_j = \mathbf{D}_i^{*T} \mathbf{Y}_k = \mathbf{D}_i^{*T} \mathbf{Y}_l$ for $k, l \in (A \setminus A^t), k \neq l$. Hence, if the maximum of Eq. 3.3 occurs at exactly two points j_1^* and j_2^* , both these points $\mathbf{X}_{j_1^*}$ and $\mathbf{X}_{j_2^*}$ are anchor points and both are added to anchor set A . If the maximum occurs at

more than two points, some of these are the anchors and others are conic combinations of these anchors. We can identify the anchors of this subset of points by calling Algorithm 1 recursively.

(3) In Algorithm 1, the vector \mathbf{p} needs to satisfy $\mathbf{p}^T \mathbf{x}_i > 0, i = 1 \dots n$. In our implementation, we simply used $\mathbf{p} = \mathbf{1} + \delta \in \mathbb{R}^m$ where δ is small perturbation vector with entries i.i.d. according to a uniform distribution $\mathcal{U}(0, 10^{-5})$. This is done to avoid the possibility of \mathbf{p} being collinear with \mathbf{D}_i^* .

4 Near-separable NMF with Bregman divergence

Let $\phi : \mathcal{S} \mapsto \mathbb{R}$ be a strictly convex function on domain $\mathcal{S} \subseteq \mathbb{R}$ which is differentiable on its non-empty relative interior $ri(\mathcal{S})$. Bregman divergence is then defined as $D_\phi(x, y) = \phi(x) - \phi(y) - \phi'(y)(x - y)$ where $\phi'(y)$ is the continuous first derivative of $\phi(\cdot)$ at y . Here we will also assume $\phi'(\cdot)$ to be smooth which is true for most Bregman divergences of interest. A Bregman divergence is always convex in the first argument. Some instances of Bregman divergence are also convex in the second argument (e.g., KL divergence). For two matrices \mathbf{X} and \mathbf{Y} , we work with divergence of the form $D_\phi(\mathbf{X}, \mathbf{Y}) := \sum_{ij} D_\phi(\mathbf{X}_{ij}, \mathbf{Y}_{ij})$.

Here we consider the case when the entries of data matrix \mathbf{X} are generated from an exponential family distribution with parameters satisfying the separability assumption, i.e., $\mathbf{X}_{ij} \sim \mathcal{P}_\phi(\mathbf{W}^i \mathbf{H}_j)$, $\mathbf{W} \in \mathbb{R}_+^{m \times r}$, $\mathbf{H} = [\mathbf{I} \mathbf{H}'] \in \mathbb{R}_+^{r \times n}$ (\mathbf{W}^i and \mathbf{H}_j denote the i th row of \mathbf{W} and the j th column of \mathbf{H} , respectively). Every member distribution \mathcal{P}_ϕ of the exponential family has a unique Bregman divergence $D_\phi(\cdot, \cdot)$ associated with it (Banerjee et al., 2005), and solving $\min_{\mathbf{Y}} D_\phi(\mathbf{X}, \mathbf{Y})$ is equivalent to maximum likelihood estimation for parameters \mathbf{Y}_{ij} of the distribution $\mathcal{P}_\phi(\mathbf{Y}_{ij})$. Hence, the projection step in Algorithm 1 is changed to $\mathbf{H} = \arg \min_{\mathbf{B} \geq 0} D_\phi(\mathbf{X}, \mathbf{X}_A \mathbf{B})$. We use the coordinate descent based method of Li et al. (2012) to solve the projection step. To select the anchor columns with Bregman projections $\mathbf{X}_A \mathbf{H}$, we modify the selection criteria as

$$(4.5) \quad j^* = \arg \max_j \frac{(\phi''(\mathbf{X}_A \mathbf{H}_i) \odot \mathbf{R}_i)^T \mathbf{X}_j}{\mathbf{p}^T \mathbf{X}_j}$$

for any $i : \|\mathbf{R}_i\| > 0$, where $\mathbf{R} = \mathbf{X} - \mathbf{X}_A \mathbf{H}$ and $\phi''(\mathbf{x})$ is the vector of second derivatives of $\phi(\cdot)$ evaluated at individual elements of the vector \mathbf{x} (i.e., $[\phi''(\mathbf{x})]_j = \phi''(\mathbf{x}_j)$), and \odot denotes element-wise product of vectors. We can show the following result regarding the anchor selection property of this criteria. Recall that an anchor is a column that can not be expressed as conic combination of other columns in \mathbf{X} .

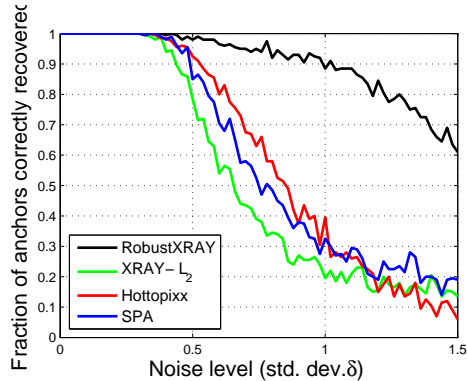


Figure 3: Sparse noise case: anchor recovery rate versus noise level (best viewed in color)

THEOREM 4.1. *If the maximizer of Eq. 4.5 is unique, the data point \mathbf{X}_{j^*} added at each iteration in the Selection step, is an anchor that has not been selected in one of the previous iterations.*

The proof is provided in the Supplementary Material. Again, any exterior point i can be chosen to select the next anchor but our simulations show that taking exterior point to be $i = \arg \max_k D_\phi(\mathbf{X}_k, \mathbf{X}_A \mathbf{H}_k)$ gives much better performance under noise than randomly choosing the exterior point. Note that for the Bregman divergence induced by function $\phi(x) = x^2$, the selection criteria of Eq. 4.5 reduces to the selection criteria of XRAY proposed in (Kumar et al., 2013).

Since Bregman divergence is not generally symmetric, it is also possible to have the projection step as $\mathbf{H} = \arg \min_{\mathbf{B} \geq 0} D_\phi(\mathbf{X}_A \mathbf{B}, \mathbf{X})$. In this case, the selection criteria will change to $j^* = \arg \max_j \frac{(\phi'(\mathbf{X}_i) - \phi'(\mathbf{X}_A \mathbf{H}_i))^T \mathbf{X}_j}{\mathbf{p}^T \mathbf{X}_j}$ for any point i exterior to the current cone, where $\phi'(\mathbf{x})$ operates element-wise on vector \mathbf{x} . However this variant does not have as meaningful a probabilistic interpretation as the one discussed earlier.

5 Empirical Observations

In this section, we present experiments on synthetic and real datasets to demonstrate the effectiveness of the proposed algorithms under noisy conditions. In addition to comparing our algorithms with existing separable NMF methods (Bittorf et al., 2012; Gillis & Vavasis, 2012; Kumar et al., 2013), we also benchmark them against Robust PCA and local-search based low-rank factorization methods, wherever applicable, for the sake of providing a more complete picture.

5.1 Anchor recovery under noise Here we test the proposed algorithms for recovery of anchors when the separable structure is perturbed by noise. Since there are no existing separable NMF algorithms that work

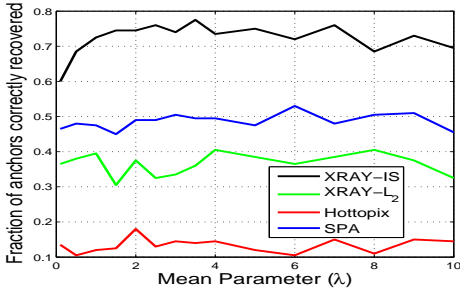


Figure 4: Data matrix generated from exponential distribution: anchor recovery rate versus mean parameter (best viewed in color)

with ℓ_1 and Bregman loss functions, we compare with methods proposed in Gillis & Vavasis (2012) (abbrv. as SPA for Successive Projection Approximation), Bittorf et al. (2012) (abbrv. as *Hottopix*) and Kumar et al. (2013) (abbrv. as XRAY- ℓ_2) to highlight that the choice of a suitable loss function plays a crucial role in column-selection. There exist local search methods for ℓ_1 - and Bregman NMF (Guan et al., 2012; Sra & Dhillon, 2005) but these are not comparable here since they do not recover the anchor columns.

First, we consider the case when the separable structure is perturbed by addition of a sparse noise matrix, i.e., $\mathbf{X} = \mathbf{W}\mathbf{H} + \mathbf{N}$, $\mathbf{H} = [\mathbf{I} \mathbf{H}']$. Each entry of matrix $\mathbf{W} \in \mathbb{R}_+^{200 \times 20}$ is generated i.i.d. from a uniform distribution between 0 and 1. The matrix $\mathbf{H} \in \mathbb{R}^{20 \times 210}$ is taken to be $[\mathbf{I}_{20 \times 20} \mathbf{H}'_{20 \times 190}]$ where each column of \mathbf{H}' is sampled i.i.d. from a Dirichlet distribution whose parameters are generated i.i.d. from a uniform distribution between 0 and 1. It is clear from the structure of matrix \mathbf{H} that first twenty columns are the anchors. The data matrix \mathbf{X} is generated as $\mathbf{W}\mathbf{H} + \mathbf{N}$ with $\mathbf{N} = \max(\mathbf{N}_1, 0) \in \mathbb{R}_+^{200 \times 210}$, where each entry of \mathbf{N}_1 is generated i.i.d. from a Laplace distribution having zero mean and δ standard deviation. Since Laplace distribution is symmetric around mean, almost half of the entries in matrix \mathbf{N} are 0 due to the max operation. The std. dev. δ is varied from 0 to 1.5 with a step size of 0.02. Fig. 3 plots the fraction of correctly recovered anchors averaged over 10 runs for each value of δ . The proposed RobustXRAY (Algorithm 1) outperforms all other methods including XRAY- ℓ_2 by a huge margin as the noise level increases. This highlights the importance of using the right loss function in the projection step that is suitable for the noise model (in this case ℓ_1 loss of Eq. 3.4).

Next, we consider the case where the non-negative data matrix is generated from an exponential family distribution other than the Gaussian, i.e., $\mathbf{X}_{ij} \sim \mathcal{P}_\phi(\mathbf{W}^i \mathbf{H}_j)$, $\mathbf{W} \in \mathbb{R}_+^{m \times r}$, $\mathbf{H} = [\mathbf{I} \mathbf{H}'] \in \mathbb{R}_+^{r \times n}$ (\mathbf{W}^i and

\mathbf{H}_j denote the i th row of \mathbf{W} and the j th column of \mathbf{H} , respectively). As mentioned earlier, every member distribution \mathcal{P}_ϕ of the exponential family has a unique Bregman divergence D_ϕ associated with it. Hence we minimize the corresponding Bregman divergence in the projection step of the algorithm as discussed in Section 4, to recover the anchor columns. Two most commonly used Bregman divergences are generalized KL-divergence and Itakura-Saito (IS) divergence (Sra & Dhillon, 2005; Banerjee et al., 2005; Févotte et al., 2009) that correspond to Poisson and Exponential distributions, respectively. We do not report results with generalized KL-divergence here since they were not very informative in highlighting the differences among various algorithms that are considered. The reason is that Poisson distribution with parameter λ has a mean of λ and std. dev. of $\sqrt{\lambda}$, and increasing the noise (std. dev.) actually increases the signal to noise ratio¹. Hence anchor recovery gets better with increasing λ (perfect recovery after certain value) and almost all algorithms perform as well as XRAY-KL for the full λ range. The anchor recovery results with IS-divergence are shown in Fig. 4. The entries of data matrix \mathbf{X} are generated as $\mathbf{X}_{ij} \sim \exp(\lambda \mathbf{W}^i \mathbf{H}_j)$ $\mathbf{W} \in \mathbb{R}_+^{200 \times 20}$, $\mathbf{H} = [\mathbf{I}_{20 \times 20} \mathbf{H}'] \in \mathbb{R}_+^{20 \times 210}$. The matrices \mathbf{W} and \mathbf{H} are generated as described in the previous paragraph. The parameter λ is varied from 0 to 10 in the steps of 0.5 and we report the average over 10 runs for each value of λ . The XRAY-IS algorithm significantly outperforms other methods including XRAY- ℓ_2 (Kumar et al., 2013) in correctly recovering the anchor column indices. The recovery rate does not change much with increasing λ since exponential distribution with mean parameter λ has a std. dev. of λ and the signal-to-noise ratio practically stays almost same with varying λ .

5.2 Exemplar Selection The problem of exemplar selection is concerned with finding a few representatives from a dataset that can summarize the dataset well. Exemplar selection can be used in many applications including summarizing a video sequence, selecting representative images or text documents (e.g., tweets) from a collection, etc. If \mathbf{X} denotes the data matrix where each *column* is a data point, the exemplar selection problem translates to selecting a few columns from \mathbf{X} that can act as representatives for all the columns. The separable NMF algorithms can be used for this task, working under the assumption that all data points (columns of \mathbf{X}) can be expressed as non-negative linear combinations

¹Poisson distribution with parameter λ closely resembles a Gaussian distribution with mean λ and std. dev. $\sqrt{\lambda}$, for large values of λ .

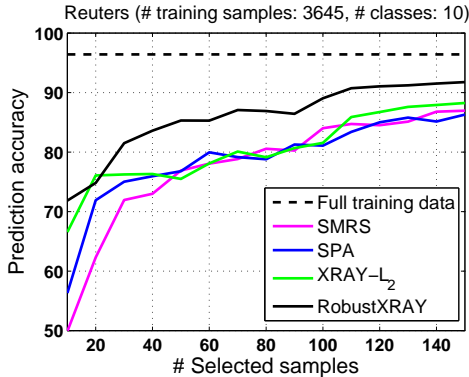


Figure 5: Accuracy of SVM trained with selected exemplars on *Reuters data* (best viewed in color)

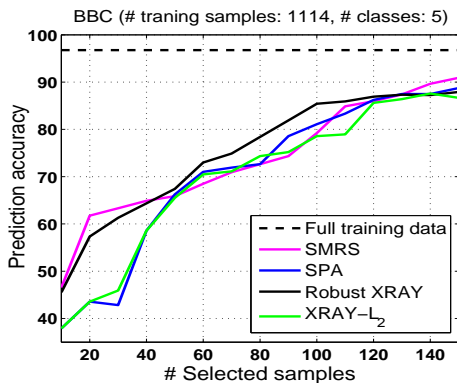


Figure 6: Accuracy of SVM trained with selected exemplars on *BBC data* (best viewed in color)

of the exemplars (the anchor columns). To be able to compare the quality of the selected exemplars by different algorithms in an objective manner, we test them on a classification task (assuming that every data point has an associated label). We randomly partition the data in training and test sets, and use only training set in selecting the exemplars. We train a multiclass SVM classifier (Fan et al., 2008) with the selected exemplars and look at its accuracy on the held-out test set. The accuracy of the classifier trained with the full training set is taken as a benchmark and is also reported. We also compare with Elhamifar et al. (2012) who recently proposed a method for exemplar selection, named as Sparse Modeling Representative Selection (SMRS). They assume that the data points can be expressed as a convex linear combination of the exemplars and minimize $\|\mathbf{X} - \mathbf{XC}\|_{\text{F}}^2 + \lambda \|\mathbf{C}\|_{1,2}$ s.t. $\mathbf{1}^T \mathbf{C} = \mathbf{1}^T$. The columns of \mathbf{X} corresponding to the non-zero rows of \mathbf{C} are selected as exemplars. We use the code provided by the authors for SMRS. There are multiple possibilities for anchor selection criteria in the proposed RobustXRAY and XRAY- l_2 (Kumar et al., 2013) and we use *max* criterion for both the algorithms.

We report results with two text datasets:

Reuters (Reuters) and BBC (Greene & Cunningham, 2006). We use a subset of Reuters data corresponding to the most frequent 10 classes which amounts to 7285 documents and 18221 words ($\mathbf{X} \in \mathbb{R}_+^{18221 \times 7285}$). The BBC data consists of 2225 documents and 9635 words with 5 classes ($\mathbf{X} \in \mathbb{R}_+^{9635 \times 2225}$). For both datasets, we evenly split the documents into training and test set, and select the exemplars from the training set using various algorithms. Fig. 5 and Fig. 6 show the plot of SVM accuracy on the test set against the number of selected exemplars that are used for training the classifier. The number of selected anchors is varied from 10 to 150 in the steps of 10. The accuracy using the full training set is also shown (dotted black line). For Reuters data, the proposed RobustXRAY algorithm outperforms other methods by a significant margin for the whole range of selected anchors. All methods seem to perform comparably on BBC data. An advantage of SPA and XRAY family of methods is that there is no need for a cleaning step to remove near-duplicate exemplars as needed in SMRS (Elhamifar et al., 2012). Another advantage is of computational speed – in all our experiments, SPA and XRAY methods are about 3–10 times faster than SMRS. It is remarkable that even a low number of selected exemplars give reasonable classification accuracy for all methods – SMRS gives 50% accuracy for Reuters data using 10 exemplars (on average 1 training sample per class) while RobustXRAY gives more than 70%.

5.3 Foreground-background Separation In this section, we consider the problem of foreground-background separation in video. The camera position is assumed to be *almost* fixed throughout the video. In all video frames, the camera captures the background scene superimposed with a *limited* foreground activity (e.g., movement of people or objects). Background is assumed to be stationary or slowly varying across frames (variations in illumination and shadows due to lighting or time of day) while foreground is assumed to be composed of objects that move across frames but span only a few pixels. If we vectorize all video frames and stack them as rows to form the matrix \mathbf{X} , the foreground-background separation problem can be modeled as decomposing \mathbf{X} into a low-rank matrix \mathbf{L} (modeling the background) and a sparse matrix \mathbf{S} (modeling the foreground).

Connection to Median Filtering: Median filtering is one of the most commonly used background modeling techniques (Sen-Ching & Kamath, 2004), which simply models the background as the pixel-wise median of the video frames. The assumption is that each pixel location stays in the background for more than half of the video frames. Consider the NMF

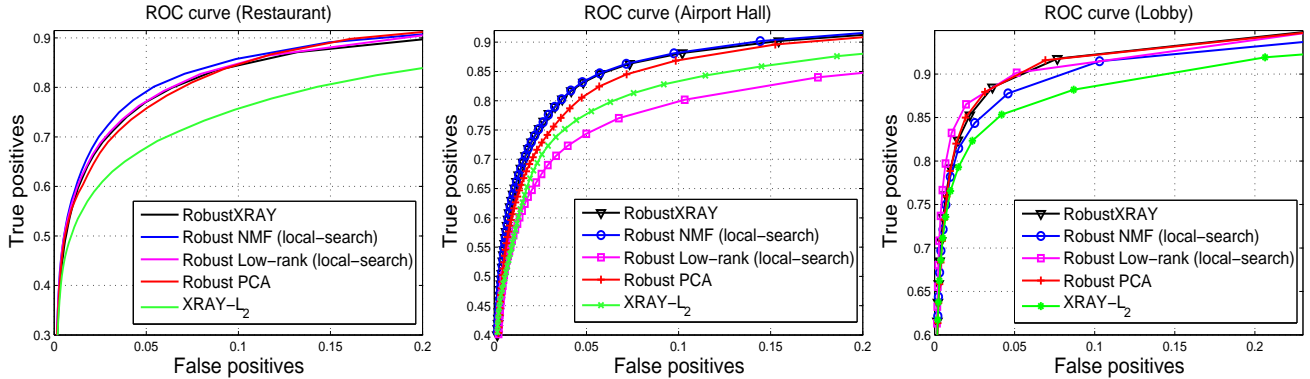


Figure 7: Foreground-background separation: ROC curves with various methods for *Restaurant*, *Airport Hall* and *Lobby* video sequences. The ranges for X and Y axes are chosen to better highlight the differences among the ROC curves. (best viewed in color)

of inner-dimension 1: $\min_{\mathbf{w} \geq 0, \mathbf{h} \geq 0} \|\mathbf{X} - \mathbf{w}\mathbf{h}\|_1$. If we constrain the vector \mathbf{w} to be all ones, the solution $\mathbf{h}^* = \arg \min_{\mathbf{w}=1, \mathbf{h} \geq 0} \|\mathbf{X} - \mathbf{w}\mathbf{h}\|_1$ is nothing but the element-wise median of all rows of \mathbf{X} . More generally, if \mathbf{w} is constrained to be such that $w_i = c > 0 \forall i$, the solution \mathbf{h}^* is a scaled version of the element-wise median vector. Hence Robust NMF under this very restrictive setting is equivalent to median filtering on the video frames, and we can hope that loosening this assumption and allowing for higher inner-dimension in the factorization can help in modeling more variations in the background.

We use three video sequences for evaluation: *Restaurant*, *Airport Hall* and *Lobby* (Li et al., 2004). *Restaurant* and *Airport Hall* are videos taken at a buffet restaurant and at a hall of an airport, respectively. The lighting are distributed from the ceilings and significant shadows of moving persons cast on the ground surfaces from different directions can be observed in the videos. *Lobby* video sequence was captured from a lobby in an office building and has background changes due to lights being switched on/off. The ground truth (whether a pixel belongs to foreground or background) is also available for these video sequences and we use it to generate the ROC curves. We mainly compare RobustXRAY with Robust PCA which is widely considered state of the art methodology for this task in the Computer Vision community. In addition, we also compare with two local-search based approaches: (i) *Robust NMF (local-search)* which solves $\min_{\mathbf{W} \geq 0, \mathbf{H} \geq 0} \|\mathbf{X} - \mathbf{WH}\|_1$ using local search, and (ii) *Robust Low-rank (local-search)* which solves $\min_{\mathbf{W}, \mathbf{H}} \|\mathbf{X} - \mathbf{WH}\|_1$ using local search. We use an ADMM based optimization procedure for both these local-search methods. We also show results with XRAY- ℓ_2 of Kumar et al. (2013) to highlight the importance of having near-separable NMFs with ℓ_1 loss for this problem. For both XRAY- ℓ_2

and RobustXRAY, we do 1 to 2 refitting steps to refine the solution (i.e., solve $\mathbf{H} = \min_{\mathbf{B} \geq 0} \|\mathbf{X} - \mathbf{X}_A \mathbf{B}\|_1$ then solve $\mathbf{W} = \min_{\mathbf{C} \geq 0} \|\mathbf{X} - \mathbf{CH}\|_1$). For all the methods, we do a grid search on the parameters (inner-dimension or rank parameter for the factorization methods and λ parameter for Robust PCA) and report the best results for each method.

Fig. 7 shows the ROC plots for the three video datasets. For *Restaurant* data, all robust methods (those with ℓ_1 penalty on the foreground) perform almost similarly. For *Airport Hall* data, RobustXRAY is tied with local-search based Robust NMF and these two are better than other methods. Surprisingly, XRAY- ℓ_2 performs better than local-search based Robust Low-rank which might be due to bad initialization. For *Lobby* data, local-search based Robust low-rank, Robust PCA and RobustXRAY perform almost similarly, and are better than local-search based Robust NMF. The results on these three datasets show that RobustXRAY is a promising method for the problem of foreground-background separation which has a huge advantage over Robust PCA in terms of speed. Our MATLAB implementation was at least 10 times faster than the inexact Augmented Lagrange Multiplier (i-ALM) implementation of Lin et al. (2010).

6 Conclusion and Future Work

We have proposed generalized conical hull algorithms to extend near-separable NMFs to robust (ℓ_1) loss function and Bregman divergences. Empirical results on exemplar selection and video background-foreground modeling problems suggest that this is a promising methodology. Avenues for future work include formal theoretical analysis of noise robustness and applications to online settings.

References

- Arora, Sanjeev, Ge, Rong, Kannan, Ravi, and Moitra, Ankur. Computing a nonnegative matrix factorization – provably. In *STOC*, 2012.
- Arora, Sanjeev, Ge, Rong, Halpern, Yoni, Mimno, David, Moitra, Ankur, Sontag, David, Wu, Yichen, and Zhu, Michael. A practical algorithm for topic modeling with provable guarantees. 2013.
- Banerjee, Arindam, Merugu, Srujana, Dhillon, Inderjit S, and Ghosh, Joydeep. Clustering with bregman divergences. *Journal of Machine Learning Research*, 6:1705–1749, 2005.
- Bittorf, Victor, Recht, Benjamin, Re, Christopher, and Tropp, Joel A. Factoring nonnegative matrices with linear programs. In *NIPS*, 2012.
- Boyd, Stephen, Parikh, Neal, Chu, Eric, Peleato, Borja, and Eckstein, Jonathan. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.
- Cichocki, A., Zdunek, R., Phan, A. H., and Amari, S. *Non-negative Matrix and Tensor Factorizations*. Wiley, 2009.
- Clarkson, K. More output-sensitive geometric algorithms. In *FOCS*, 1994.
- Donoho, D. and Stodden, V. When does non-negative matrix factorization give a correct decomposition into parts? In *NIPS*, 2003.
- Dula, J. H., Hegalson, R. V., and Venugopal, N. An algorithm for identifying the frame of a pointed finite conical hull. *INFORMS Jour. on Comp.*, 10(3):323–330, 1998.
- Elhamifar, Ehsan, Sapiro, Guillermo, and Vidal, Rene. See all by looking at a few: Sparse modeling for finding representative objects. In *CVPR*, 2012.
- Esser, Ernie, Mller, Michael, Osher, Stanley, Sapiro, Guillermo, and Xin, Jack. A convex model for non-negative matrix factorization and dimensionality reduction on physical space. *IEEE Transactions on Image Processing*, 21(10):3239 – 3252, 2012.
- Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. Liblinear: A library for large linear classification. *JMLR*, 2008.
- Févotte, Cédric, Bertin, Nancy, and Durrieu, Jean-Louis. Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis. *Neural computation*, 21(3):793–830, 2009.
- Gillis, Nicolas. Successive nonnegative projection algorithm for robust nonnegative blind source separation. *SIAM Journal on Imaging Sciences*, 7(2):1420–1450, 2014.
- Gillis, Nicolas and Luce, Robert. Robust near-separable nonnegative matrix factorization using linear optimization. *Journal of Machine Learning Research*, 15(Apr): 1249–1280, 2014.
- Gillis, Nicolas and Vavasis, Stephen A. Fast and robust recursive algorithms for separable nonnegative matrix factorization. *arXiv:1208.1237v2*, 2012.
- Greene, Derek and Cunningham, Pádraig. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *ICML*, 2006.
- Guan, Naiyang, Tao, Dacheng, Luo, Zhigang, and Shawe-Taylor, John. Mahnmf: Manhattan non-negative matrix factorization. *CoRR abs/1207.3438*, 2012.
- Hsieh, C. J. and Dhillon, I. S. Fast coordinate descent methods with variable selection for non-negative matrix factorization. In *KDD*, 2011.
- Kim, Hyunsoo and Park, Haesun. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 23:1495–1502, 2007.
- Kumar, Abhishek, Sindhvani, Vikas, and Kambadur, Prabhajan. Fast conical hull algorithms for near-separable non-negative matrix factorization. In *ICML*, 2013.
- Lee, D. and Seung, S. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.
- Li, Liangda, Lebanon, Guy, and Park, Haesun. Fast bregman divergence nmf using taylor expansion and coordinate descent. In *KDD*, 2012.
- Li, Liyuan, Huang, Weimin, Gu, Irene Yu-Hua, and Tian, Qi. Statistical modeling of complex backgrounds for foreground object detection. *Image Processing, IEEE Transactions on*, 13(11):1459–1472, 2004.
- Lin, C.-J. Projected gradient methods for non-negative matrix factorization. *Neural Computation*, 2007.
- Lin, Zhouchen, Chen, Minming, and Ma, Yi. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*, 2010.
- Reuters. archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection.
- Sen-Ching, S Cheung and Kamath, Chandrika. Robust techniques for background subtraction in urban traffic video. In *Electronic Imaging 2004*, pp. 881–892, 2004.
- Sra, Suvrit and Dhillon, Inderjit S. Generalized nonnegative matrix approximations with bregman divergences. In *Advances in neural information processing systems*, pp. 283–290, 2005.