Online Nonnegative Matrix Factorization with General Divergences

Renbo Zhao
National University of Singapore

Vincent Y. F. Tan
National University of Singapore

Huan Xu
Georgia Institute of Technology

Abstract

We develop a unified and systematic framework for performing online nonnegative matrix factorization under a wide variety of important divergences. The online nature of our algorithms makes them particularly amenable to large-scale data. We prove that the sequence of learned dictionaries converges almost surely to the set of critical points of the expected loss function. Experimental results demonstrate the computational efficiency and outstanding performances of our algorithms on several real-life applications, including topic modeling, document clustering and foreground-background separation.

1 Introduction

Nonnegative Matrix Factorization (NMF) has been a popular data analysis technique over recent years, due to its non-subtractive and parts-based interpretation on the learned basis [Lee and Seung, 1999]. Given a nonnegative matrix $\mathbf{V}$ with dimension $F \times N$, one seeks a nonnegative basis matrix $\mathbf{W}$ and a nonnegative coefficient matrix $\mathbf{H}$ such that $\mathbf{V} \approx \mathbf{WH}$, by solving

$$\min_{\mathbf{W}\geq 0, \mathbf{H}\geq 0} \left[D(\mathbf{V}\|\mathbf{WH}) \triangleq \sum_{n=1}^{N} d(\mathbf{v}_n \| \mathbf{W}\mathbf{h}_n) \right] \qquad (1)$$

where $\mathbf{v}_n$ (resp. $\mathbf{h}_n$) denotes the n-th column of $\mathbf{V}$ (resp. $\mathbf{H}$) and $d(\cdot\|\cdot)$ denotes a divergence between two vectors. In the NMF literature, in addition to the squared-ℓ_2 loss, i.e., $d(\mathbf{x}\|\mathbf{y}) = \frac{1}{2}\|\mathbf{x}-\mathbf{y}\|_2^2$, many other divergences have been proposed for two main purposes. First, given the observation noise of a particular distribution, there exists a divergence such that solving (1) correspond to the maximum-likelihood (ML) estimation of ground-truth data matrix under observation $\mathbf{V}$.[1] It has been shown empirically that if the divergence used in (1) does not match the distribution of the noise, the results will be inferior [Févotte et al., 2009]; thus it is imperative to use the correct divergence. Second, many robust divergences (or loss functions) have been proposed in order to overcome the well-known sensitivity of the squared-ℓ_2 loss to outliers in the data matrix $\mathbf{V}$.

Despite the successes of NMF algorithms with the aforementioned divergences, the *batch* data processing mode intrinsic to the algorithms prohibits them from being applied to *large-scale* data, i.e., data collections with a large number of data samples or even streaming data. The reasons are twofold: (i) the storage space might be insufficient to store the entire set of samples, and (ii) the high computational complexity incurred in each iteration slows down the algorithms significantly. On the other hand, although many online NMF (and other matrix factorization) algorithms have been proposed, most of them are developed for the squared-ℓ_2 loss. Thus, their applications are limited, especially when the noise is not Gaussian. Only a few works consider divergence other than the squared-ℓ_2 loss. For example, the Itakura-Saito (IS) divergence and Huber loss have been considered in Lefèvre et al. [2011] and Chen et al. [2015], Wang et al. [2013] respectively. Although these works have shown promising performances on some specific divergences, the approaches therein cannot be easily generalized to other divergences in a straightforward manner. Also, convergence guarantees (of the sequence of basis matrices) are largely lacking in these works.[2]

1.1 Challenges and Main Contributions

In this paper, we develop a framework termed *online NMF with general divergences* that learns the dictionary $\mathbf{W}$ in (1) in an online manner under a va-

Proceedings of the 20^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2017, Fort Lauderdale, Florida, USA. JMLR: W&CP volume 54.

[1] For example, if the distribution of the observation noise belongs to the exponential family, then the corresponding divergence belongs to the class of Bregman divergences [Banerjee et al., 2005].

[2] See Section S-1 for a comprehensive literature review.

riety of divergences.[3] Many prior works on online NMF with the squared-ℓ_2 loss leverage the *stochastic Majorization-Minimization (MM)* [Mairal, 2013] framework. However, this method cannot be applied to most of the general divergences we consider, since crucially, sufficient statistics in the method cannot be formed. Therefore, we leverage the *stochastic approximation* (SA) framework [Borkar, 2008] and *projected dynamical system* [Dupuis and Nagurney, 1993, Teschl, 2012] to develop an algorithm that does not need to compute the sufficient statistics, so it can effectively handle general divergences while being amenable to convergence analyses. Our analysis shows that the sequence of learned dictionaries converges almost surely to the set of critical points of the expected loss function (3). This serves as a substantial generalization of the results in previous works (e.g., Shen et al. [2014]). The analysis techniques in our work are vastly different from those in prior works, so they open new avenues for analyzing similar problems. In addition, by using robust loss functions, we indeed propose a novel way to perform *online robust NMF*. This complements a previous approach based on the ℓ_1 regularization on outliers [Zhao and Tan, 2017]. Additionally, we have conducted extensive numerical experiments on both synthetic and real datasets. The results demonstrate the computational efficiency and promising performances of our algorithms on several real-life applications, including topic modeling, document clustering and foreground-background separation.

1.2 Notations

We use boldface capital letters, boldface lowercase letters and plain lowercase letters to denote matrices, column vectors and scalars respectively. Given a matrix $\mathbf{X}$, we denote its i-th row as $\mathbf{X}_{i:}$, j-th column as $\mathbf{X}_{:j}$ and (i,j)-th entry as x_{ij}. For a column vector $\mathbf{x}$, its i-th entry is denoted by x_i. We denote the (Euclidean) projection operator onto a set $\mathcal{S}$ as $\Pi_{\mathcal{S}}$. We denote the set of nonnegative and positive real numbers and the set of natural numbers (excluding zero) as $\mathbb{R}_+$, $\mathbb{R}_{++}$ and $\mathbb{N}$ respectively. For $N \in \mathbb{N}$, define $[N] \triangleq \{1, 2, \ldots, N\}$. In this work, we use $\mathbf{v}$, $\mathbf{W}$ and $\mathbf{h}$ to denote the (nonnegative) data vector, basis matrix and coefficient vector respectively. The ambient and latent data dimensions are denoted as F and K respectively, which are *independent of time*.[4]

All the sections, figures and tables with indices containing an 'S' will appear in the supplemental material.

[3] See Section 2.1 for the divergences covered in this work.

[4] In this work, we do not simultaneously consider the data with high ambient dimensions. An attempt on this problem in the context of dictionary learning with the squared-ℓ_2 loss has been made in Mensch et al. [2016].

2 Problem Formulation

2.1 General Divergences

In general, a (vector) divergence is a bivariate function $d(\cdot\|\cdot) : \mathbb{R}^F_{++} \times \mathbb{R}^F_{++} \to \mathbb{R}_+$.[5] In this work, we consider a wide range of divergences $\mathcal{D} \triangleq \mathcal{D}_1 \cup \mathcal{D}_2$, where

$\mathcal{D}_1 \triangleq \{d(\cdot\|\cdot) \,|\, \forall\, \mathbf{x} \in \mathbb{R}^F_{++}, \forall\, \text{compact } \mathcal{Y} \subseteq \mathbb{R}^F_{++}, d(\mathbf{x}\|\cdot) \text{ is differentiable on } \mathbb{R}^F_{++} \text{ and } \nabla d(\mathbf{x}\|\cdot) \text{ is Lipschitz on } \mathcal{Y}\}$,

$\mathcal{D}_2 \triangleq \{d(\cdot\|\cdot) \,|\, \forall\, \mathbf{x} \in \mathbb{R}^F_{++}, d(\mathbf{x}\|\cdot) \text{ is convex on } \mathbb{R}^F_{++}\}$.

Remark 1. In particular, $\mathcal{D}_1$ consists of the families of α, β, α-β, γ divergences, and $\mathcal{D}_2$ consists of the α-divergences, β-divergences with $\beta \in [1, 2]$ and several robust metrics, including the ℓ_1-distance, ℓ_2-distance and Huber loss. All of these divergences have been employed in various NMF applications [Chen et al., 2015, Cichocki and Amari, 2010, Cichocki et al., 2008, 2011, Févotte and Idier, 2011, Kong et al., 2011].

2.2 Optimization Problem

We focus solely on learning the basis matrix $\mathbf{W}$ since the coefficient vectors $\{\mathbf{h}_t\}_{t\in\mathbb{N}}$ (i) cannot be stored due to limitations on the storage space and (ii) each $\mathbf{h}_t$ can be easily computed (by regression) given $\mathbf{v}_t$ and the learned $\mathbf{W}$. We assume that the data vectors $\{\mathbf{v}_t\}_{t\in\mathbb{N}} \subseteq \mathbb{R}^F_{++}$ are independently generated from a distribution $\mathbb{P}$.[6] Define the loss function of $\mathbf{W}$ given a data sample $\mathbf{v} \sim \mathbb{P}$ as

$$\ell(\mathbf{v}, \mathbf{W}) \triangleq \min_{\mathbf{h}\in\mathcal{H}} d(\mathbf{v}\|\mathbf{W}\mathbf{h}), \tag{2}$$

where $d(\cdot\|\cdot) \in \mathcal{D}$ and $\mathcal{H} \triangleq \{\mathbf{h} \in \mathbb{R}^K_+ \,|\, \epsilon' \le h_i \le U', \forall\, i \in [K]\}$ for some $0 < \epsilon' < U'$. We aim to minimize the expected loss [Bottou and Bousquet, 2008]

$$\min_{\mathbf{W}\in\mathcal{C}} \left[f(\mathbf{W}) \triangleq \mathbb{E}_{\mathbf{v}\sim\mathbb{P}}[\ell(\mathbf{v}, \mathbf{W})] \right], \tag{3}$$

where $\mathcal{C} \triangleq \{\mathbf{W} \in \mathbb{R}^{F\times K}_+ \,|\, \|\mathbf{W}_{i:}\|_1 \ge \epsilon, \|\mathbf{W}_{:j}\|_\infty \le U, \forall\,(i,j) \in [F] \times [K]\}$ for some $0 < \epsilon < U$. Note that both constraint sets $\mathcal{C}$ and $\mathcal{H}$ are convex.

Remark 2. First we notice in general, $d(\cdot\|\cdot)$ is usually asymmetric about its arguments. In this work, we only consider minimizing $d(\mathbf{v}\|\mathbf{W}\mathbf{h})$ in $\mathbf{h}$ (and $\mathbf{W}$) since this corresponds to an ML estimation of the ground-truth data from the noisy data sample $\mathbf{v}$ under various

[5] For some specific divergences, e.g., the squared-ℓ_2 loss, the domain of $d(\cdot\|\cdot)$ can be relaxed to $\mathbb{R}^F_+ \times \mathbb{R}^F_+$.

[6] Although most real data do not strictly satisfy the independence assumption, we make the i.i.d. assumption here for convenience of analysis. For a finite dataset, $\mathbb{P}$ is the empirical distribution of the data samples in this dataset. For streaming data, $\mathbb{P}$ is the population distribution.

statistical models. Next, the upper and lower bounds (ϵ, ϵ', U and U') in $\mathcal{C}$ and $\mathcal{H}$ preserve the numerical stability of our algorithm, and are set to $\epsilon = \epsilon' = 1\times10^{-8}$ and $U = U' = 1\times10^{8}$. Also, the constructions of $\mathcal{C}$ and $\mathcal{H}$ enable efficient projections onto both sets. (See Section S-2-A for details.) This ensures that our algorithms (see Algorithm 1 and 2) are efficient.

3 Algorithms

The outline of our algorithm is shown in Algorithm 1. At each time t, the subroutine for learning the coefficient vector $\mathbf{h}_t$ is shown in Algorithm 2.

3.1 Definitions

In the sequel, we let $\mathcal{X}$ be a finite-dimensional real Banach space (with norm $\|\cdot\|$). For example, $\mathcal{X}$ can be $\mathbb{R}^F$ or $\mathbb{R}^{F\times N}$. Consider a function $f : \mathcal{X} \to \mathbb{R}$.

Definition 1 ([Kruger, 2003, Section 1.1])**.** The *Fréchet subdifferential* at $x \in \mathcal{X}$, $\hat{\partial} f(x)$ is defined as

$$\hat{\partial} f(x) \triangleq \left\{ g \in \mathcal{X}^* \middle| \liminf_{y\to x, y\in\mathcal{X}} \frac{f(y) - f(x) - g(y-x)}{\|y-x\|} \geq 0 \right\},$$

where $\mathcal{X}^*$ is the topological dual space of $\mathcal{X}$.

Remark 3. If f is differentiable at $x \in \mathcal{X}$, then $\hat{\partial} f(x) = \{\nabla f(x)\}$. The Fréchet subdifferential serves as a generalization of the subdifferential in convex analysis, i.e., if f is convex on $\mathcal{X}$, then for any $x \in \mathcal{X}$, $\hat{\partial} f(x) = \partial f(x)$, where $\partial f(x) \triangleq \{g \in \mathcal{X}^* | f(x) + g(y - x) \leq f(y), \forall\, y \in \mathcal{X}\}$.

Definition 2 ([Shapiro, 1990])**.** The *(Gâteaux) directional derivative* of f at $x \in \mathcal{X}$ along direction $d \in \mathcal{X}$, $f'(x; d)$ is defined as $f'(x; d) \triangleq \lim_{\delta\downarrow 0}[f(x + \delta d) - f(x)]/\delta$. Furthermore, f is called directionally differentiable if $f'(x; d)$ exists for any $x \in \mathcal{X}$ and any $d \in \mathcal{X}$.

Definition 3 ([Razaviyayn et al., 2013])**.** Assume f to be directionally differentiable on $\mathcal{X}$. Let $\mathcal{K} \subseteq \mathcal{X}$ be a convex set. A point $x^* \in \mathcal{K}$ is a *critical point* of the constrained optimization problem $\min_{x\in\mathcal{K}} f(x)$ if $f'(x^*; x - x^*) \geq 0$, $\forall\, x \in \mathcal{K}$.

In addition, for any $t \in \mathbb{N}$, we define two important functions $\widetilde{d}_t : \mathbb{R}^{F\times K}_{++} \to \mathbb{R}$ and $\overline{d}_t : \mathbb{R}^{K}_{++} \to \mathbb{R}$ as $\widetilde{d}_t(\mathbf{W}) \triangleq d(\mathbf{v}_t\|\mathbf{W}\mathbf{h}_t)$ and $\overline{d}_t(\mathbf{h}) \triangleq d(\mathbf{v}_t\|\mathbf{W}_{t-1}\mathbf{h})$, where $\{\mathbf{v}_t, \mathbf{W}_t, \mathbf{h}_t\}_{t\in\mathbb{N}}$ are generated per Algorithm 1.

3.2 Choice of Step Sizes

In algorithm 1, the step sizes $\{\eta_t\}_{t\in\mathbb{N}}$ are chosen to satisfy $\sum_{t=1}^{\infty} \eta_t = \infty$ and $\sum_{t=1}^{\infty} \eta_t^2 < \infty$. The specific forms of $\{\eta_t\}_{t\in\mathbb{N}}$ will be given in Section 6.2. Depending on the divergences, the step sizes $\{\beta_t^k\}_{k\in\mathbb{N}}$ in Algorithm 2 can be chosen using Armijo's rule, the constant step size policy [Parikh and Boyd, 2014] or the modified Polyak's step size policy [Nedić, 2008]. See Section S-2-B for detailed discussions.

Algorithm 1 Online NMF with General Divergences

Input: Initial basis matrix[7] $\mathbf{W}_0 \in \mathcal{C}$, number of iterations T, sequence of step sizes $\{\eta_t\}_{t\in\mathbb{N}}$

for $t = 1$ to T **do**

1) Draw a data sample $\mathbf{v}_t$ from $\mathbb{P}$.

2) Learn the coefficient vector $\mathbf{h}_t$ per Algorithm 2 such that $\mathbf{h}_t$ is a critical point of

$$\min_{\mathbf{h}\in\mathcal{H}} \left[\overline{d}_t(\mathbf{h}) \triangleq d(\mathbf{v}_t\|\mathbf{W}_{t-1}\mathbf{h})\right]. \tag{4}$$

3) Update the basis matrix from $\mathbf{W}_{t-1}$ to $\mathbf{W}_t$

$$\mathbf{W}_t := \Pi_{\mathcal{C}}\Big\{\mathbf{W}_{t-1} - \eta_t \mathbf{G}_t\Big\}, \tag{5}$$

where $\mathbf{G}_t$ is any element in $\hat{\partial}\widetilde{d}_t(\mathbf{W}_{t-1})$.

end for

Output: Final basis matrix $\mathbf{W}_T$

3.3 Discussions

In some previous works on online matrix factorization (with squared ℓ_2 loss) [Feng et al., 2013, Mairal et al., 2010, Shen et al., 2014], a different approach has been employed to update the basis matrix. Namely, at each time t, $\mathbf{W}_t$ is the minimizer of $\min_{\mathbf{W}\in\mathcal{C}} f_t(\mathbf{W})$. Here $f_t : \mathcal{C} \to \mathbb{R}$ is an upper-bound function of $\widehat{f}_t : \mathcal{C} \to \mathbb{R}$, defined as $\widehat{f}_t(\mathbf{W}) \triangleq \frac{1}{t}\sum_{i=1}^{t} \ell(\mathbf{v}_i, \mathbf{W})$. At a high level, this approach belongs to the class of stochastic MM algorithms [Mairal, 2013, Razaviyayn et al., 2016]. As noted in Mairal [2013], direct minimization of f_t is possible only when f_t can be parameterized by variables of small and constant size (known as sufficient statistics in Mairal et al. [2010]) for each $t \in \mathbb{N}$. Unfortunately this condition does not hold for most divergences beyond the squared ℓ_2 loss, including those in class $\mathcal{D}$. However, if we assume for each $\mathbf{v}$, $\ell(\mathbf{v}, \cdot)$ has Lipschitz gradient on $\mathcal{C}$ and choose f_t as a quadratic upper-bound function of $\widehat{f}_t$, then the recursive update form of $\mathbf{W}_t$ via the stochastic MM approach can be regarded as *a special case* of our method. See Razaviyayn et al. [2016] for details.

4 Main Convergence Theorem

Our main convergence theorem concerns the divergences in class $\mathcal{D}_1 \cap \mathcal{D}_2$ (see Remark 1), i.e., the di-

[7] $\mathbf{W}_0$ can be chosen as any element in $\mathcal{C}$, and same for $\mathbf{h}_0$ in Algorithm 2.

[8] $\mathbf{g}_t^k$ can be chosen as any element in $\hat{\partial}\overline{d}_t(\mathbf{h}_t^{k-1})$.

Algorithm 2 Learning $\mathbf{h}_t$

Input: initial coefficient vector $\mathbf{h}_t^0 \in \mathcal{H}$, basis matrix $\mathbf{W}_{t-1}$, data sample $\mathbf{v}_t$, step sizes $\{\beta_t^k\}_{k\in\mathbb{N}}$, maximum number of iterations Υ

For $k = 1, 2, \ldots, \Upsilon$

$\mathbf{h}_t^k := \Pi_{\mathcal{H}}\Big\{\mathbf{h}_t^{k-1} - \beta_t^k \mathbf{g}_t^k\Big\}$, where[8] $\mathbf{g}_t^k \in \hat{\partial}\overline{d}_t(\mathbf{h}_t^{k-1})$

Output: Final coefficient vector $\mathbf{h}_t \triangleq \mathbf{h}_t^{\Upsilon}$

vergences $d(\cdot\|\cdot)$ that are convex and smooth in the second argument.[9]

Assumptions.

1. *The support set $\mathcal{V} \subseteq \mathbb{R}_{++}^F$ for the distribution $\mathbb{P}$ is compact.*
2. *For all $(\mathbf{v}, \mathbf{W}) \in \mathcal{V}\times\mathcal{C}$ and $d(\cdot\|\cdot)\in\mathcal{D}_2$, $d(\mathbf{v}\|\mathbf{W}\mathbf{h})$ is m-strongly convex in $\mathbf{h}$ for some constant $m > 0$.*

Remark 4. The abovementioned two assumptions are reasonable in the following sense. Assumption 1 naturally holds for real data, which are uniformly bounded entrywise. Assumption 2 is a classical assumption in literature [Mairal et al., 2010, Shen et al., 2014]. It ensures the minimizer of (4) is unique. This assumption can be satisfied by simply adding a Tikhonov regularizer $\frac{m}{2}\|\mathbf{h}\|_2^2$ to $d(\mathbf{v}\|\mathbf{W}\mathbf{h})$. Adding such regularizers can be regarded as a way to promote smoothness and avoid over-fitting on $\mathbf{h}$. Also, including the regularizers will not alter our analysis significantly, so we omit them in the objective function.

We now state our main theorem.

Theorem 1. *As $t \to \infty$, the sequence of dictionaries $\{\mathbf{W}_t\}_{t\in\mathbb{N}}$ converges almost surely to the set of critical points of* (3) *formulated with any divergence in $\mathcal{D}_1 \cap \mathcal{D}_2$.*

Remark 5. We notice that the same convergence guarantees have been proved in previous works in which the divergence term in the NMF objective function is the squared-ℓ_2 loss [Mairal et al., 2010, Shen et al., 2014]. Therefore, our result here can be considered as a substantial *generalization* of the previous results, since the class $\mathcal{D}_2$ includes many more important divergences, as discussed in Remark 1. At a higher level, our problem falls within the scope of *stochastic multi-block nonconvex optimization*. Without additional assumptions on the regularity of the problem, convergence guarantees to the global optima are in general out-of-reach. Indeed, the state-of-the-art convergence guarantees on such problems [Ghadimi and Lan, 2013, 2016, Reddi et al., 2016] are typically stated in terms of the critical points. For the matrix factorization problems, the critical points are often empirically appealing, e.g., Mairal et al. [2010], Shen et al. [2014].

[9]The efficiency and efficacy of our algorithm for the divergences in $\mathcal{D}_1\Delta\mathcal{D}_2$ will be empirically verified in Section 6. See Section S-3-F for the technical difficulties (and possible approaches) for proving such convergence results for the divergences in $\mathcal{D}_1\Delta\mathcal{D}_2$.

5 Convergence analysis

5.1 Notations and Definitions

We denote the underlying probability space for the whole stochastic process $\{\mathbf{v}_t, \mathbf{W}_t, \mathbf{h}_t\}_{t\in\mathbb{N}}$ generated by Algorithm 1 as $(\Omega, \mathscr{B}, \mu)$.[10] In the sequel, we use $t \in \mathbb{N}$ and $s \in \mathbb{R}_+$ as time indices for discrete-time and continuous-time stochastic processes respectively. For any $\omega \in \Omega$, we use $\mathbf{X}_t(\omega)$ and $\mathbf{X}(\omega, s)$ to denote the values of $\mathbf{X}_t$ and $\mathbf{X}(s)$ evaluated at ω respectively.

Definition 4 (Equicontinuity and asymptotic equicontinuity [Kushner and Yin, 2003])**.** Let $\mathcal{Y}$ be a real finite-dimensional Banach space (with norm $\|\cdot\|_{\mathcal{Y}}$). A sequence of functions $\{f_n : \mathcal{X} \to \mathcal{Y}\}_{n\in\mathbb{N}}$ is *equicontinuous* (e.c.) at $x \in \mathcal{X}$ if for any $\epsilon > 0$, there exists $\delta > 0$ such that $\sup_{n\in\mathbb{N}} \sup_{x'\in\mathcal{X}:\|x-x'\|<\delta} \|f_n(x) - f_n(x')\|_{\mathcal{Y}} < \epsilon$ and *asymptotically equicontinuous* (a.e.c.) at $x \in \mathcal{X}$ if $\limsup_{n\to\infty} \sup_{x'\in\mathcal{X}:\|x-x'\|<\delta} \|f_n(x) - f_n(x')\|_{\mathcal{Y}} < \epsilon$. If $\{f_n\}_{n\geq 1}$ is e.c. (resp. a.e.c.) at each $x \in \mathcal{X}$, then $\{f_n\}_{n\geq 1}$ is e.c. (resp. a.e.c.) on $\mathcal{X}$.

Definition 5 (Projected dynamical system, limit set and stationary points [Teschl, 2012])**.** Given a closed and convex set $\mathcal{K}$ in $\mathcal{X}$, and a continuous function $g : \mathcal{K} \to \mathcal{X}$, the projected dynamical system (PDS) (on an interval $\mathcal{I} \subseteq \mathbb{R}_+$) associated with $\mathcal{K}$ and g with initial value $x_0 \in \mathcal{K}$ is defined as

$$\frac{\mathrm{d}}{\mathrm{d}s}x(s) = \pi_{\mathcal{K}}\Big[x(s), g(x(s))\Big],\ x(0) = x_0,\ s \in \mathcal{I}, \quad (6)$$

where $\pi_{\mathcal{K}}[x, v] \triangleq \lim_{\delta\downarrow 0} (\Pi_{\mathcal{K}}(x + \delta v) - x)/\delta,\ \forall x \in \mathcal{K}, \forall\, v \in \mathcal{X}$. Denote $\mathcal{P}(g, \mathcal{K}, x_0)$ as the solution set of (6). The limit set of (6), $\mathcal{L}(g, \mathcal{K}, x_0)$ is defined as $\mathcal{L}(g, \mathcal{K}, x_0) \triangleq \bigcup_{x(\cdot)\in\mathcal{P}(g,\mathcal{K},x_0)}\{y \in \mathcal{K} \,|\, \exists\{s_n\}_{n\in\mathbb{N}} \subseteq \mathbb{R}_+, s_n \uparrow \infty, x(s_n) \to y\}$. Moreover, the set of stationary points associated with g and $\mathcal{K}$, $\mathcal{S}(g, \mathcal{K})$ is defined as $\mathcal{S}(g, \mathcal{K}) \triangleq \{x \in \mathcal{K} \,|\, \pi_{\mathcal{K}}[x, g(x)] = 0\}$.

5.2 Preliminary Lemmas

Lemma 1 and 2 below state the regularity properties of the loss function ℓ and its expectation f in (3).[11]

[10]Since $\{\mathbf{v}_t\}_{t\in\mathbb{N}}$ are drawn i.i.d. from $\mathbb{P}$, $\{\mathbf{W}_t, \mathbf{h}_t\}_{t\in\mathbb{N}}$ are also random variables.

[11]For the divergences $d(\cdot\|\cdot)$ in $\mathcal{D}_1 \cap \mathcal{D}_2$, $\hat{\partial}\widetilde{d}_t(\mathbf{W}_{t-1}) = \{\nabla_{\mathbf{W}} d(\mathbf{v}_t\|\mathbf{W}_{t-1}\mathbf{h}_t)\}$.

Lemma 1. *$\ell(\cdot,\cdot)$ is differentiable on $\mathbb{R}^F_{++}\times\mathbb{R}^{F\times K}_{++}$ and $(\mathbf{v},\mathbf{W})\mapsto\nabla_{\mathbf{W}}\ell(\mathbf{v},\mathbf{W})$ is continuous on $\mathcal{V}\times\mathcal{C}$. Moreover, let $\mathbf{h}^*(\mathbf{v},\mathbf{W})\triangleq\min_{\mathbf{h}\in\mathcal{H}}d(\mathbf{v}\|\mathbf{W}\mathbf{h})$,[12] then $\nabla_{\mathbf{W}}\ell(\mathbf{v},\mathbf{W})=\nabla_{\mathbf{W}}d(\mathbf{v}\|\mathbf{W}\mathbf{h}^*(\mathbf{v},\mathbf{W}))$.*

Lemma 2. *The expected loss (objective) function f is continuously differentiable on $\mathcal{C}$ and $\nabla f(\mathbf{W})=\mathbb{E}_{\mathbf{v}\sim\mathbb{P}}\left[\nabla_{\mathbf{W}}\ell(\mathbf{v},\mathbf{W})\right]$ for each $\mathbf{W}\in\mathcal{C}$.*

Corollary 1. *$\mathbb{E}_{\mathbf{v}_t\sim\mathbb{P}}[\nabla_{\mathbf{W}}d(\mathbf{v}_t\|\mathbf{W}_{t-1}\mathbf{h}_t)]=\nabla f(\mathbf{W}_{t-1})$, for any $t\in\mathbb{N}$. In other words, the stochastic (noisy) gradient $\nabla_{\mathbf{W}}d(\mathbf{v}_t\|\mathbf{W}_{t-1}\mathbf{h}_t)$ in Algorithm 1 acts as an unbiased estimator of the "true" gradient $\nabla f(\mathbf{W}_{t-1})$.*

Now, define the "noise" part in the stochastic gradient[13] $\nabla_{\mathbf{W}}d(\mathbf{v}_t\|\mathbf{W}_{t-1}\mathbf{h}_t)$ in (5), $\mathbf{N}_t\triangleq\nabla_{\mathbf{W}}\ell(\mathbf{v}_t,\mathbf{W}_{t-1})-\nabla f(\mathbf{W}_{t-1})$. Also, define a filtration $\{\mathscr{F}_t\}_{t\geq0}$ such that $\mathscr{F}_t\triangleq\sigma\{\mathbf{v}_i,\mathbf{W}_i,\mathbf{h}_i\}_{i=1}^t$ for all $t\geq1$ and $\mathscr{F}_0=\{\emptyset,\Omega\}$. From Lemma 1 and Corollary 1, we can show that $\{\mathbf{N}_t\}_{t\in\mathbb{N}}$ is a martingale difference sequence with uniformly bounded second moment.

5.3 Continuous-time Interpolations

Observe that (5), which lies at the core of our analysis, is a discrete-time PDS. We find it more convenient to analyze a continuous-time analogue of it, so we perform (continuous-time) constant interpolation on (5). Specifically, we first explicitly model the projection $\Pi_{\mathcal{C}}$ in (5) as an additive noise term $\mathbf{Z}_t$:

$$\mathbf{W}_t\triangleq\mathbf{W}_{t-1}-\eta_t\nabla f(\mathbf{W}_{t-1})-\eta_t\mathbf{N}_t+\eta_t\mathbf{Z}_t,\qquad(7)$$

where $\mathbf{Z}_t\triangleq\frac{1}{\eta_t}\Pi_{\mathcal{C}}\{\mathbf{W}_{t-1}-\eta_t\nabla_{\mathbf{W}}\widetilde{\ell}(\mathbf{v}_t,\mathbf{W}_{t-1})\}-\frac{1}{\eta_t}\{\mathbf{W}_{t-1}-\eta_t\nabla_{\mathbf{W}}\widetilde{\ell}(\mathbf{v}_t,\mathbf{W}_{t-1})\}$. Then we define three sequences of functions $\{F^t\}_{t\in\mathbb{N}}$, $\{N^t\}_{t\in\mathbb{N}}$ and $\{Z^t\}_{t\in\mathbb{N}}$ with common domain $\mathbb{R}_+$ as

$$F^t(s)\triangleq-\sum_{i=t}^{m(s_t+s)-1}\eta_{i+1}\nabla f(\mathbf{W}_i),$$

$$N^t(s)\triangleq-\sum_{i=t+1}^{m(s_t+s)}\eta_i\mathbf{N}_i,\quad Z^t(s)\triangleq\sum_{i=t+1}^{m(s_t+s)}\eta_i\mathbf{Z}_i,$$

for $s>0$ and $F^t(0)=N^t(0)=Z^t(0)\triangleq0$, where

$$s_t\triangleq\begin{cases}0, & t=0\\ \sum_{i=1}^t\eta_i, & t\geq1\end{cases},\quad m(s)\triangleq\begin{cases}0, & s=0\\ t, & s\in(s_{t-1},s_t]\end{cases}.$$

Define $W^t(s)\triangleq\mathbf{W}_{m(s_t+s)-1}$. By (7), for any $t\in\mathbb{N}$ and $s\in\mathbb{R}_+$, with probability one,

$$W^t(s)=W^t(0)+F^{t-1}(s)+N^{t-1}(s)+Z^{t-1}(s).\quad(8)$$

[12]The uniqueness of minimizer is ensured by Assumption 2.

[13]$\mathbf{N}_t$ is a function of both $\mathbf{v}_t$ and $\mathbf{W}_{t-1}$, but we omit such dependence to make notations uncluttered.

5.4 Key Lemmas

Our main theorem is an immediate consequence of Lemmas 4 and 5. We first present Lemma 3 since it lays the foundations for proving Lemma 4.

Lemma 3 (Almost sure asymptotic equicontinuity of important functions)**.** *For any $t\in\mathbb{N}$, define*

$$G^t(s)\triangleq-\int_0^s\nabla f(W^t(\tau))\,d\tau,\quad s\geq0,\qquad(9)$$

$$Y^t(s)\triangleq\int_0^s Z^t(\tau)\,d\tau,\quad s\geq0.\qquad(10)$$

Then we have[14] (1) $N^t\xrightarrow{\mathrm{u}}\mathbf{0}$ on $\mathbb{R}_+$ a.s. (2) $\Delta_1^t\triangleq F^t-G^t\xrightarrow{\mathrm{u}}\mathbf{0}$ on $\mathbb{R}_+$ a.s. (3) $\{G^t\}_{t\in\mathbb{N}}$ is e.c. on $\mathbb{R}_+$ a.s. (4) $\Delta_2^t\triangleq Z^t-Y^t\xrightarrow{\mathrm{u}}\mathbf{0}$ and $\{Y^t\}_{t\in\mathbb{N}}$ is e.c. on $\mathbb{R}_+$ a.s. Consequently, $\{N^t\}_{t\in\mathbb{N}}$, $\{F^t\}_{t\in\mathbb{N}}$, $\{Z^t\}_{t\in\mathbb{N}}$ and $\{W^t\}_{t\in\mathbb{N}}$ are a.e.c. on $\mathbb{R}_+$ almost surely.

Lemma 4 (Almost sure convergence to the limit set)**.** *The stochastic process $\{\mathbf{W}_t\}_{t\in\mathbb{N}}$ generated in Algorithm 1 converges almost surely to $\mathcal{L}(-\nabla f,\mathcal{C},\mathbf{W}_0)$, the limit set of the following projected dynamical system[15]*

$$\frac{\mathrm{d}}{\mathrm{d}s}W(s)=\pi_{\mathcal{C}}\Big[W(s),-\nabla f(W(s))\Big],\quad W(0)=\mathbf{W}_0,\ s\geq0.\qquad(11)$$

Lemma 5 (Characterization of the limit set)**.** *In (11), we have $\mathcal{L}(-\nabla f,\mathcal{C},\mathbf{W}_0)\subseteq\mathcal{S}(-\nabla f,\mathcal{C})$, i.e., every limit point of (11) is a stationary point associated with $-\nabla f$ and $\mathcal{C}$. Moreover, each $\mathbf{W}\in\mathcal{S}(-\nabla f,\mathcal{C})$ satisfies the following variational inequality*

$$\langle\nabla f(\mathbf{W}),\mathbf{W}'-\mathbf{W}\rangle\geq0,\ \forall\,\mathbf{W}'\in\mathcal{C}.\qquad(12)$$

This implies each stationary point in $\mathcal{S}(-\nabla f,\mathcal{C})$ is a critical point of (3).

Remark 6. Lemma 4 and 5 together imply a two-step approach to prove Theorem 1. Specifically, Lemma 4 shows $\{\mathbf{W}_t\}_{t\in\mathbb{N}}$ converges almost surely to the limit set $\mathcal{L}(-\nabla f,\mathcal{C},\mathbf{W}_0)$ and Lemma 5 characterizes every element $\mathcal{L}(-\nabla f,\mathcal{C},\mathbf{W}_0)$ as a critical point of (3).

6 Applications and Experiments

6.1 Experimental Setup

Our experiments consist of three parts. First, we tested our online algorithms on synthetic data. Next we considered three real applications of our algorithms, including topic modeling, document clustering and foreground-background separation. All the experiments were run in 64-bit Matlab® (R2015b) on a machine with 3.6 GHz CPU and 8 GB RAM.

[14]Given a sequence of functions $\{f_n\}_{n\in\mathbb{N}}$ and a function f, $f_n\xrightarrow{\mathrm{u}}f$ denotes the uniform convergence of f_n to f.

[15]Given a finite-dimensional Banach space $\mathcal{X}$, a sequence (x_n) in $\mathcal{X}$ is said to converge to a set $\mathcal{A}\subseteq\mathcal{X}$ if $\lim_{n\to\infty}\inf_{a\in\mathcal{A}}\|x_n-a\|=0$.

6.2 Heuristics and Parameter Settings

In our experiments we mainly used two popular heuristics—namely, mini-batch input and multi-pass extension. Specifically, mini-batch input refers inputing $\tau \in \mathbb{N}$ data samples at each time, and multi-pass extension refers to running online algorithms multiple times on the datasets. See Mairal et al. [2010], Wang et al. [2011] for motivations of the heuristics.

The important parameters in our experiments include the mini-batch size τ, the latent dimension K and the step sizes $\{\eta_t\}_{t\in\mathbb{N}}$. We set $\tau = 20$. The latent dimension K was determined from the domain knowledge in Section 6.4 and fixed to 40 in the other sections. The step size η_t had the form $a/(b+\tau t)$, where $a = b = 1 \times 10^4$. The setting of (τ, K, a, b) will be hereafter referred as the *canonical parameter setting*. Some discussions on the choices of these parameters are provided in Section S-4-A. We will show that our algorithms are insensitive to the values of these parameters in Section S-4-C.

6.3 Synthetic Experiments

In this section we conducted experiments of our algorithms with six important divergence in $\mathcal{D}_1 \cup \mathcal{D}_2$, including the IS, Kullback-Leibler (KL), squared-ℓ_2, Huber, ℓ_1 and ℓ_2 divergences/losses.[16] In particular, the batch NMF algorithms with these divergences have found numerous applications [Kong et al., 2011, Lee and Seung, 1999]. To generate the synthetic data matrix $\mathbf{V}$, we first generated the ground-truth data matrix $\mathbf{V}^o \triangleq \mathbf{W}^o\mathbf{H}^o \in \mathbb{R}_+^{F\times N}$, where $(F, N) = (2\times 10^3, 1\times 10^5)$ and $\mathbf{W}^o$ and $\mathbf{H}^o$ have common dimension $K^o = 40$.[17] The entries of $\mathbf{W}^o$ and $\mathbf{H}^o$ were generated independently from the shifted half-normal distribution. Next, we imposed random noise to $\mathbf{V}^o$ in two ways. For the IS, KL and squared-ℓ_2 divergences, the noises had multiplicative Gamma, Poisson and additive Gaussian distributions respectively, so that the ML estimation of $\mathbf{V}^o$ from $\mathbf{V}$ is equivalent to solving the (batch) NMF problem (1). Since the ℓ_1, ℓ_2 and Huber losses are all robust losses, we added synthetic outliers to $\mathbf{V}^o$. The detailed data generation procedures are deferred to Section S-4-B.

To demonstrate the computational efficiency of our online algorithms, we compared them with their batch counterparts. All of the batch algorithms had been derived based on the multiplicative updates in the literature [Févotte and Idier, 2011, Kong et al., 2011, Wang et al., 2013]. The set of online and batch algorithms were denoted as OL-Div and B-Div respectively, where 'Div' denotes the abbreviation of each divergence name. In addition, for the IS, squared-ℓ_2 and Huber divergences, we also compared our algorithms with the existing online algorithms, including OL-Lef [Lefèvre et al., 2011], OL-Guan [Guan et al., 2012] and OL-Wang [Wang et al., 2013].

Figure 1 shows the plots of objective values versus time for all the six divergences.[18] We observe that for each divergence, our online algorithm converges significantly faster than its batch counterpart. Moreover, it also converges faster than (for the IS and squared-ℓ_2 losses) or as fast as (for the Huber loss) the existing online algorithms. This is because the existing online algorithms typically require to solve an optimization problem in updating the basis matrix $\mathbf{W}_t$ at each time instant, in contrast, our algorithm only involves a one-step projected gradient descent to update $\mathbf{W}_t$. We also examined the sensitivity of our algorithms to the values of (τ, K, a) using this dataset. We varied each parameter over a wide range of values and kept the other two fixed as in the canonical setting. The results (shown in Section S-4-C) indicate that our algorithms are insensitive to the parameter values.

6.4 Topic Modeling and Document Clustering

In this section we applied our online algorithm with the KL divergence, OL-KL to the tasks of topic modeling and document clustering on two datasets, `Guardian` and `Wikipedia` [Greene et al., 2014]. The `Guardian` dataset consists of 5413 documents sampled from five topics, with a vocabulary size 10801. Thus the data matrix has size 10801×5413. For the `Wikipedia` dataset, the data matrix has size 17311×5738 (with six topics). For both data matrices, we transformed the raw term frequency to the term frequency–inverse document frequency (TF-IDF) statistics and set all the zero entries to 1×10^{-3}. We compared our algorithm with two other algorithms, B-KL and OL-Wang2. The OL-Wang2 algorithm [Wang et al., 2011] is an online NMF algorithm with the squared-ℓ_2 loss that is tailored to efficient document clustering.

We now briefly introduce the methodology of using NMF-based algorithms to learn topics and cluster documents. The latent dimension K is set to the number of topics (denoted as σ) in the dataset.[19] By factorizing the TF-IDF matrix $\mathbf{V}$ into $\mathbf{W}$ and $\mathbf{H}$, each column of $\mathbf{W}$ can be regarded as representing a topic so that each document (represented as a column of

[16] Among them, the IS divergence belongs to $\mathcal{D}_1 \setminus \mathcal{D}_2$, and the other divergences belong to $\mathcal{D}_2$. The KL divergence in this work is defined as in Lee and Seung [2000].

[17] See Figure S-2 for the results when $K \neq K^o$.

[18] The results were obtained using one initialization of $\mathbf{W}_0$. We observed that the results were similar when using different initializations of $\mathbf{W}_0$.

[19] For most datasets, σ is known. Some works have studied the strategies to estimate σ, e.g., Greene et al. [2014]. However, for simplicity, we use the known value of σ.

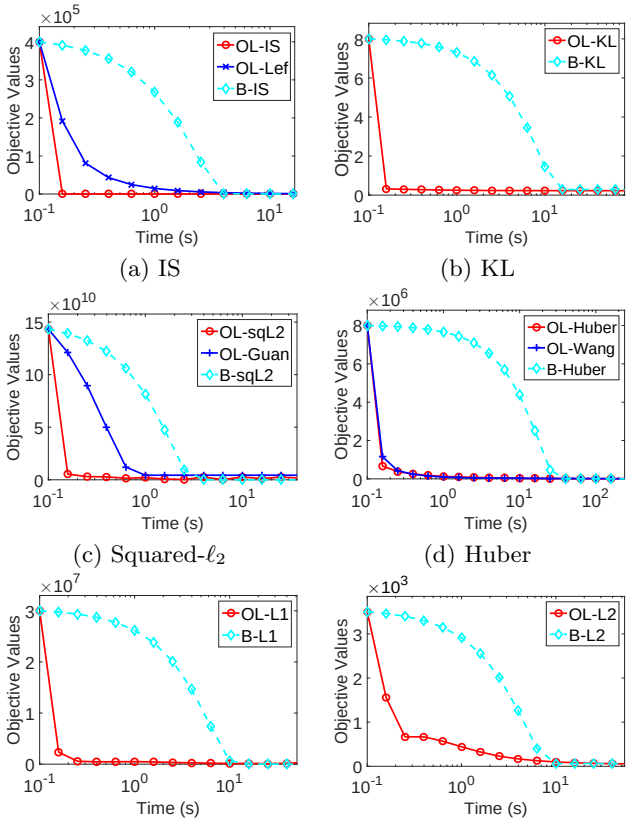

Figure 1: Plots of objective values vs time (in seconds) for all the six divergences on the synthetic dataset.

$\mathbf{V}$) can be seen as a conic combination of all the topics in the dataset. To cluster documents, in this work we assign the j-th document to the k-th topic if $k \in \arg\max_{k' \in [K]} h_{k'j}$.[20] In this work, we adopted the *Rand index* [Rand, 1971] as the quantitative measure of clustering accuracy.[21]

We ran all the algorithms using 20 randomly initializations of $\mathbf{W}_0$. Table 1 shows the topics learned by the three algorithms on the `Guardian` dataset using one initialization.[22] For each topic, we presented five most representative words, i.e., the words whose corresponding entries have the highest magnitudes in one column of $\mathbf{W}$.[23] Table 1 shows that for each topic, the representative words learned by all the three algorithms are similar and represent that topic very well. Table 2 shows the average clustering accuracies and

[20] If k is not unique, we chose the smallest one. We tried other more complex classifiers (e.g., k-means) on the set of coefficient vectors $\{\mathbf{h}_j\}_{j \in [N]}$, and obtained similar results.

[21] The Rand index has been widely used in the information retrieval literature to measure the clustering accuracy. See Manning et al. [2008, Chapter 8] for details.

[22] We observed that different initializations yielded similar results. This observation also holds for Section 6.5.

[23] The augmented results, with ten most representative words for each topic, are shown in Table S-2.

Table 1: Topics learned from the `Guardian` dataset by three algorithms: OL-KL, B-KL and OL-Wang2.

Business	Politics	Music	Fashion	Football
company	labour	music	fashion	league
sales	ultimately	album	wonder	club
market	party	band	weaves	universally
shares	government	songs	week	welsh
business	unions	vogue	war	team

(a) OL-KL

Business	Politics	Music	Fashion	Football
bank	labour	music	fashion	league
company	party	album	wonder	club
ultimately	cameron	band	weaves	universally
growth	ultimately	vogue	week	team
market	unions	songs	look	welsh

(b) B-KL

Business	Politics	Music	Fashion	Football
bank	labour	music	fashion	league
growth	party	album	week	club
shares	unions	band	wonder	welsh
company	miliband	vogue	weaves	season
market	voluntary	songs	war	universally

(c) OL-Wang2

Table 2: Average document clustering accuracies and running times of OL-KL, B-KL and OL-Wang2 on the `Guardian` dataset.

Algorithms	Accuracy	Time (s)
OL-KL	0.697 ± 0.01	29.25 ± 0.58
B-KL	0.701 ± 0.01	183.32 ± 2.09
OL-Wang2	0.643 ± 0.03	32.46 ± 0.68

running times (with standard deviations) of the three algorithms. In terms of running times, the online algorithms (OL-KL and OL-Wang2) are equally fast and are significantly faster than the batch algorithm B-KL. In terms of clustering accuracy, our online algorithm performs almost as good as its batch counterpart. In addition, both of them perform better than OL-Wang2. This suggests KL-divergence-based NMF algorithms may have superior performances on document clustering over those based on the squared-ℓ_2 loss on certain text corpora. Similar results were also obtained on the `Wikipedia` dataset. See Section S-4-D for details.

6.5 Foreground-Background Separation

Next we applied our online algorithm with the Huber loss, OL-Huber to foreground-background separation, an important task in video surveillance. We used two datasets, Hall and Escalator [Li et al., 2004]. The Hall dataset consists of 1250 8-bit gray-scale video frames with resolution 144×176, so the data matrix $\mathbf{V}$ has dimension 25344×1250. Similarly, the data matrix $\mathbf{V}$ of the Escalator dataset has dimension 20800×2000. We compared OL-Huber with three other NMF algo-

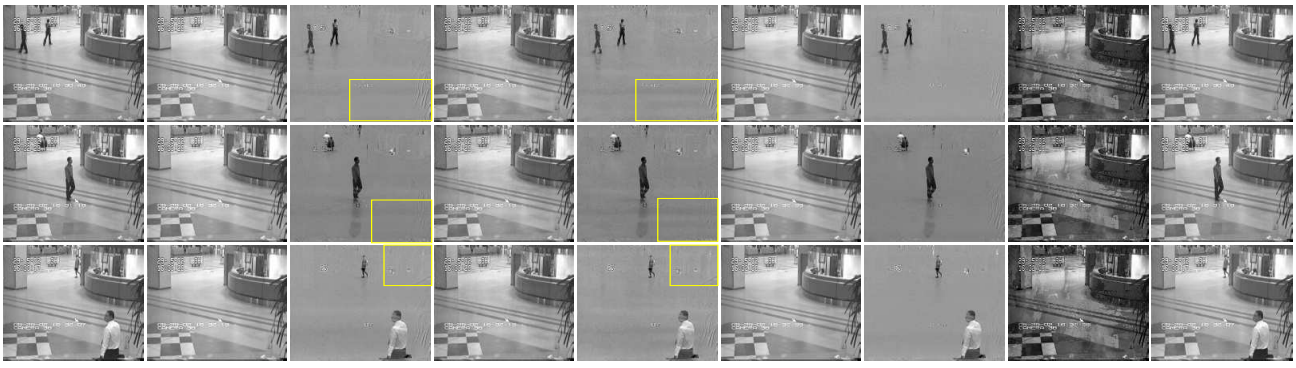

Figure 2: Foreground-background separation results on the `Hall` dataset with four algorithms: (a) OL-Huber, (b) OL-Wang, (c) B-Huber and (d) OL-Guan. The leftmost column shows the original video frames. The differences between the foreground images produced by OL-Huber and OL-Wang are highlighted in yellow boxes.[25]

rithms, namely OL-Wang, B-Huber and OL-Guan.

For (surveillance) video sequences, the background usually changes slowly over time so it can be modeled as a low-rank matrix. The foreground objects, which change rapidly, usually only occupy small areas, thus they can be modeled as sparse outliers. This naturally motivates us to learn the low-rank background (given as the product of basis and coefficient matrices) using NMF algorithms with robust loss functions, e.g., the Huber loss. After that, the foreground can be recovered by subtracting the learned background from the original video frames. For the online algorithms, we estimate the background in the t-th frame as $\overline{\mathbf{W}}\mathbf{h}_t$, where $\overline{\mathbf{W}}$ is the output basis matrix by Algorithm 1.

We ran all the algorithms using 20 randomly initializations of $\mathbf{W}_0$. All the online algorithms were run on each dataset for two passes. We set the threshold parameter of the Huber loss function to 10, for all Huber-loss-based algorithms. Figure 2 shows the foreground-background separation results for some randomly sampled video frames.[26] We observe that all the Huber-loss-based algorithms succeed in separating the foreground objects from the background. Among them, the foreground images separated by B-KL have the best visual qualities, since they have almost no artifacts in the background. Although both OL-Huber and OL-Wang produce artifacts in the foreground images, the artifacts are less severe for OL-Huber. In contrast, OL-Guan, which is based on the squared-ℓ_2 loss, completely fails on this task. This is because the squared-ℓ_2 loss is very sensitive to outliers, thus a proper background model cannot be estimated. In terms of the average running times (shown in Table 3), we observe that both OL-Huber and OL-Wang are significantly faster than B-Huber. They are also faster than OL-Guan, since the perturbations from outliers make the latter slow to converge. Similar results were obtained from the `Escalator` dataset. See Section S-4-E for details.

Table 3: Average running times of OL-Huber, OL-Wang, B-Huber and OL-Guan on the `Hall` dataset.

Algorithms	Time (s)	Algorithms	Time (s)
OL-Huber	38.79 ± 0.45	OL-Wang	45.36 ± 0.59
B-Huber	276.66 ± 1.93	OL-Guan	95.85 ± 0.82

[25] Zoom in to better observe the differences.

[26] For all the online algorithms, the results are all from the second pass. See Figure S-4 for additional results.

References

A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh. Clustering with Bregman divergences. *J. Mach. Learn. Res.*, 6:1705–1749, Dec. 2005.

V. S. Borkar. *Stochastic approximation: a dynamical systems viewpoint*. Cambridge, 2008.

L. Bottou and O. Bousquet. The tradeoffs of large scale learning. In *Proc. NIPS*, pages 161–168, Vancouver, Canada, Dec. 2008.

J. Chen, Z. J. Towfic, and A. H. Sayed. Dictionary learning over distributed models. *IEEE Trans. Signal Process.*, 63(4):1001–1016, 2015.

A. Cichocki and S. Amari. Families of Alpha-Beta-and Gamma-divergences: Flexible and robust measures of similarities. *Entropy*, 12(6):1532–1568, 2010.

A. Cichocki, H.-K. Lee, Y.-D. Kim, and S. Choi. Non-negative matrix factorization with alpha-divergence. *Pattern Recognit. Lett.*, 29(9):1433–1440, 2008.

A. Cichocki, S. Cruces, and S.-i. Amari. Generalized alpha-beta divergences and their application to robust nonnegative matrix factorization. *Entropy*, 13 (1):134–170, 2011.

P. Dupuis and A. Nagurney. Dynamical systems and variational inequalities. *Ann. Oper. Res.*, 44(1):7–42, 1993.

J. Feng, H. Xu, and S. Yan. Online robust PCA via stochastic optimization. In *Proc. NIPS*, pages 404–412, Lake Tahoe, USA, Dec. 2013.

C. Févotte and J. Idier. Algorithms for nonnegative matrix factorization with the beta-divergence. *Neural Comput.*, 23(9):2421–2456, 2011.

C. Févotte, N. Bertin, and J.-L. Durrieu. Nonnegative matrix factorization with the Itakura-Saito divergence. With application to music analysis. *Neural Comput.*, 21(3):793–830, Mar. 2009.

S. Ghadimi and G. Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM J. Optim.*, 23(4):2341–2368, 2013.

S. Ghadimi and G. Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Math. Program.*, 156(1):59–99, 2016.

D. Greene, D. O'Callaghan, and P. Cunningham. How Many Topics? Stability Analysis for Topic Models. In *Proc. ECML*, Nancy, France, 2014.

N. Guan, D. Tao, Z. Luo, and B. Yuan. Online nonnegative matrix factorization with robust stochastic approximation. *IEEE Trans. Neural Netw. Learn. Syst.*, 23(7):1087–1099, Jul. 2012.

D. Kong, C. Ding, and H. Huang. Robust nonnegative matrix factorization using ℓ_{21}-norm. In *Proc. CIKM*, pages 673–682, Glasgow, Scotland, UK, Oct. 2011.

A. Y. Kruger. On fréchet subdifferentials. *J. Math. Sci.*, 116(3):3325–3358, 2003.

H. J. Kushner and G. Yin. *Stochastic approximation and recursive algorithms and applications*. Springer, 2003.

D. D. Lee and H. S. Seung. Learning the parts of objects by nonnegative matrix factorization. *Nature*, 401:788–791, October 1999.

D. D. Lee and H. S. Seung. Algorithms for nonnegative matrix factorization. In *Proc. NIPS*, pages 556–562, Denver, USA, Dec. 2000.

A. Lefèvre, F. Bach, and C. Févotte. Online algorithms for nonnegative matrix factorization with the itakura-saito divergence. In *Proc. WASPAA*, pages 313–316, New Paltz, New York, USA, Oct 2011.

L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian. Statistical modeling of complex backgrounds for foreground object detection. *IEEE Trans. Image Process.*, 13(11):1459–1472, Nov. 2004.

J. Mairal. Stochastic majorization-minimization algorithms for large-scale optimization. In *Proc. NIPS*, pages 2283–2291, Lake Tahoe, USA, Dec. 2013.

J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *J. Mach. Learn. Res.*, 11:19–60, Mar 2010.

C. D. Manning, P. Raghavan, and H. Schutze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

A. Mensch, J. Mairal, B. Thirion, and G. Varoquaux. Dictionary learning for massive matrix factorization. In *Proc. ICML*, New York, USA, Jul. 2016.

A. Nedić. Subgradient projection method. `http://www.ifp.illinois.edu/~angelia/sgd_notes.pdf`, 2008.

N. Parikh and S. Boyd. Proximal algorithms. *Found. Trends Optim.*, 1(3):123–231, 2014.

W. M. Rand. Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc.*, 66(336):846–850, 1971.

M. Razaviyayn, M. Hong, and Z.-Q. Luo. A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM J. Optim.*, 23(2):1126–1153, 2013.

M. Razaviyayn, M. Sanjabi, and Z.-Q. Luo. A stochastic successive minimization method for nonsmooth nonconvex optimization with applications to transceiver design in wireless communication networks. *Math. Program.*, 157(2):515–545, 2016.

S. J. Reddi, S. Sra, B. Poczos, and A. Smola. Fast stochastic methods for nonsmooth nonconvex optimization. In *Proc. NIPS*, Barcelona, Spain, Dec. 2016.

A. Shapiro. On concepts of directional differentiability. *J. Optim. Theory Appl.*, 66(3):477–487, 1990.

J. Shen, H. Xu, and P. Li. Online optimization for max-norm regularization. In *Proc. NIPS*, pages 1718–1726, Montréal, Canada, Dec. 2014.

G. Teschl. *Ordinary Differential Equations and Dynamical Systems*. Amer. Math. Soc., 2012.

F. Wang, C. Tan, A. C. Konig, and P. Li. Efficient document clustering via online nonnegative matrix factorizations. In *Proc. SDM*, pages 908–919, Mesa, Arizona, USA, Apr. 2011.

N. Wang, J. Wang, and D.-Y. Yeung. Online robust non-negative dictionary learning for visual tracking. In *Proc. ICCV*, pages 657–664, Sydney, Australia, Dec. 2013.

R. Zhao and V. Y. F. Tan. Online nonnegative matrix factorization with outliers. *IEEE Trans. Signal. Process.*, pages 555–570, 2017.