Stochastic Approximation and Memory-Limited Subspace Tracking for Poisson Streaming Data

Liming Wang, *Member, IEEE,* and Yuejie Chi, *Senior Member, IEEE*

***Abstract*—Poisson count data is ubiquitously encountered in applications such as optical imaging, social networks and traffic monitoring, where the data is typically modeled after a Poisson distribution and presented in a streaming fashion. Therefore it calls for techniques to efficiently extract and track the useful information embedded therein. We consider the problem of recovering and tracking the underlying Poisson rate, where the rate vectors are assumed to lie in an unknown low-dimensional subspace, from streaming Poisson data with possibly missing entries. The recovery of the underlying subspace is posed as an expected loss minimization problem under nonnegative constraints, where the loss function is a penalized Poisson log-likelihood function. A stochastic approximation (SA) algorithm is proposed and can be implemented in an online manner. Two theoretical results are established regarding the convergence of the SA algorithm. The SA algorithm is guaranteed almost surely to converge to the same point as the original expected loss minimization problem, and the estimate converges to a local minimum. To further reduce the memory requirement and handle missing data, the SA algorithm is modified via lower bounding the log-likelihood function in a form that is decomposable and can be implemented in a memory-limited manner without storing history data. Numerical experiments are provided to demonstrate the superior performance of the proposed algorithms, compared to existing algorithms. The memory-limited SA algorithm is shown to empirically yield similar performance as the original SA algorithm at a much lower memory requirement.**

***Index Terms*—Poisson data, Poisson noise, count data, stochastic approximation, subspace estimation and tracking.**

I. Introduction

There is an increasing interest in exploring and interpreting high-dimensional streaming count data, which has appeared ubiquitously in numerous areas such as social networks [1], medical imaging [2], [3], photonics [4], traffic monitoring and surveillance [5]. The characteristics of count data significantly differ from traditional data defined in the continuous domain, and call for new processing techniques. Consequently, as the Gaussian model is generally inappropriate for count data, the Poisson model starts to serve a pivotal role in modeling such data.

Consider a high-dimensional count data stream, where the data vector $\mathbf{y}_n \in \mathbb{Z}_+^N$ at each time n is typically modeled as $\mathbf{y}_n \sim \mathrm{Pois}(\mathbf{z}_n)$, where $\mathrm{Pois}(\cdot)$ denotes the vector Poisson distribution, and $\mathbf{z}_n \in \mathbb{R}_+^N$ is the rate vector. Moreover, in many applications, the data vectors may not be fully observed due to packet loss, privacy considerations or missing data. Therefore, it is vital to propose online algorithms that can accurately learn and track the underlying structure of the Poisson model, e.g. changes in the rates, in both computational- and memory-efficient manners, as well as being robust to missing data.

Efforts have been devoted to achieve the aforementioned goals for the Gaussian model, where the data vectors $\mathbf{y}_n$'s are assumed drawn from the Gaussian distribution, from two perspectives. The first approach is to design memory-efficient online algorithms tailored for streaming data, which do not require storing all the previous data samples. These include online versions of classical algorithms such as Principal Component Analysis (PCA) [6] and dictionary learning [7]. The other approach is to effectively reduce the data dimensionality by exploring its hidden structure. To be more specific, data in very high dimensional ambient space can often be effectively represented by a much lower-dimensional structure, and processing of the original data can be efficiently carried out on this low-dimensional structure. Nonnegative matrix factorization (NMF) is an eminent example along this direction [8], which seeks to approximately decompose a nonnegative data matrix into a product of two nonnegative matrices of smaller sizes. This also makes it possible to recover the data even when it is highly subsampled, for example completing a low-rank matrix [9]. By assuming the data vectors lie approximately in a low-dimensional subspace, a series of recent work [10]–[14] have developed low-complexity subspace estimation and tracking algorithms under the Gaussian model, using partially observed or even corrupted streaming data. In [15], the authors proposed a sequential optimization framework which extends the Gaussian model to a binary model. Very recently, extensions of the Gaussian model to categorical data via Probit, Tobit and Logit models are considered in [16], and several algorithms are proposed to accommodate large-scale categorical data that are incomplete and online. However, these approaches do not generalize straightforwardly to the Poisson model in a memory-efficient manner.

Inverting a Poisson measurement model in a batch setting has been studied from various perspectives. In the Poisson compressed sensing (CS) framework [17], the rate vector is modeled as $\mathbf{z}_n = \mathbf{A}\mathbf{x}_n$, where the aim is to recover a sparse vector $\mathbf{x}_n$ whose dimension is much higher than that of $\mathbf{y}_n$ when the sensing matrix $\mathbf{A}$ is known *a priori*. When $\mathbf{A}$ satisfies certain desirable properties such as the restricted isometry property, performance bounds for sparse recovery are

L. Wang is with Department of Electrical and Computer Engineering, The Ohio State University, Columbus, OH 43210, USA. Email: wang.8482@osu.edu.

Y. Chi is with Department of Electrical and Computer Engineering and Department of Biomedical Informatics, The Ohio State University, Columbus, OH 43210, USA. Email: chi.97@osu.edu.

This work is supported in part by AFOSR under the grant FA9550-15-1-0205, by ONR under the grant N00014-15-1-2387, and by NSF under the grants CAREER ECCS-1650449, ECCS-1462191 and CCF-1704245.

developed in [18], [19] for the single measurement case. The impact of a designed sensing matrix under the Poisson model has been investigated in [20]. In [3], the Poisson CS framework is extended to the multiple measurement setting, where it proposes a batch algorithm to recover multiple sparse vectors $\{\mathbf{x}_n\}$. Similarly, [21], [22] developed batch algorithms for the Poisson matrix completion problem, which aims to recover the rate vectors $\{\mathbf{z}_n\}$, assuming it lies in a low-dimensional subspace. These batch algorithms become highly inefficient in terms of computational cost and storage complexity for large-scale data streams, and do not adapt to changes in an online setting.

Motivated by [11], this paper assumes that the rate vectors $\mathbf{z}_n$ lie in a low-dimensional subspace that may change over time. The goal is to develop efficient online algorithms for estimating and tracking the low-dimensional subspace, and recovering the rate vectors, from streaming observations in a low-complexity manner. To begin with, we formulate the subspace estimation problem as an expected loss minimization problem with *nonnegative* constraints, which minimizes the expectation of a Tikhonov-penalized negative Poisson log-likelihood function. The stochastic approximation (SA) framework [23] is invoked to develop the online algorithm, which follows two steps at each time to solve the optimization problem. In the first *non-negative encoding* step, the nonnegative coefficient of the rate vector in the subspace is estimated by using the previously learned subspace representation. During the second step named *subspace update*, the subspace is updated using the previously estimated coefficients under a nonnegative constraint. Under a few mild assumptions, we establish the convergence of the proposed SA algorithm. Namely, the SA algorithm is guaranteed almost surely to converge to the same point as the original expected loss minimization problem, and the estimate converges to a local minimum.

However, distinct from the Gaussian model, the above SA algorithm is not memory-efficient, namely, it requires a memory space that grows with the size of the data stream, due to the form of the Poisson model. Alternatively, we derive lower bounds of the likelihood function (i.e., upper bounds of the objective function) and optimize the obtained bounds instead to mitigate the issue, so that the subspace can be estimated by using some surrogates whose sizes do not grow with time and can be updated efficiently in an online manner. This is denoted as the memory-limited SA algorithm. Moreover, we provide extensions even when the data vectors are only partially observed. Numerical simulations are provided to show that the memory-limited SA algorithm empirically yields similar performance as the original SA algorithm. Furthermore, we conduct experiments using both synthetic and video data to demonstrate the performance of the proposed algorithms, where they significantly outperform state-of-the-art subspace tracking algorithms that naively apply Gaussian models to count data.

It is worth noting that several other approaches exist in the literature for tracking streaming data that can be applied to count data, such as online convex optimization [5] and nonparametric Bayesian factor analysis [24]. They can be thought as complementary to ours, as they make different assumptions about the underlying structures of the rate vectors, as well as adopt different analytical frameworks. Our focus is on estimating and tracking the (possibly abrupt) changes of the underlying low-dimensional subspace, which can be used to model many data types of interest. Numerical experiments on real data are provided to compare our approach against these alternatives.

The paper is organized as follows. In Section II, we first introduce our model and the problem formulation. In Section III, we propose the stochastic approximation algorithms, and develop memory-limited modifications to the SA algorithm, which also allow for the missing data case. We present the convergence analysis of the proposed SA algorithm in Section IV. Several numerical experiments for synthetic and real datasets are presented in Section V. We conclude the paper in Section VI.

Notations: Bold upper and lower case letters are used to denote matrices and vectors, respectively, *e.g.,* $\mathbf{D}$ is a matrix and $\mathbf{x}$ is a vector. $\|\cdot\|_F$ and $\|\cdot\|_2$ denote the Frobenius norm and ℓ_2 norm, respectively. $\text{Tr}(\cdot)$ denotes the trace of an argument matrix, and $\odot$ denotes the Hadamard product (entry-wise product). In addition, we follow the convention that $\log 0 = 0$.

II. Signal Model and Problem Statement

Consider the following Poisson streaming data model:

$$\mathbf{y}_n \sim \text{Pois}(\mathbf{z}_n), \qquad n = 1, 2, \ldots, \tag{1}$$

where the observation at time n is given as $\mathbf{y}_n = [y_{n,1}, \ldots, y_{n,N}]^T \in \mathbb{Z}_+^N$, $\mathbf{z}_n = [z_{n,1}, \ldots, z_{n,N}]^T \in \mathbb{R}_+^N$ is the Poisson rate vector, and $\text{Pois}(\cdot)$ denotes the vector-Poisson distribution, *i.e.,* $\text{Pois}(\mathbf{z}_n) = \prod_{i=1}^N \text{pois}(z_{n,i})$, where $\text{pois}(\cdot)$ is the common scalar Poisson distribution with parameter $z_{n,i}$. Moreover, we assume that the rate vector $\mathbf{z}_n$ lies in an K-dimensional subspace that possibly changes over time, given as

$$\mathbf{z}_n = \mathbf{D}_n \mathbf{a}_n, \tag{2}$$

where $\mathbf{D}_n \in \mathbb{R}_+^{N\times K}$, $\mathbf{a}_n \in \mathbb{R}_+^K$ and K is the dimension of the subspace, $K \ll N$. For simplicity, we have assumed that the subspace dimension K does not change with time, which can be thought as an upper bound of all possible dimensions if it indeed changes with time.

Moreover, in many applications, we may only observe a subset of entries in the data vectors. Let $\mathbf{p}_n = [p_{n,1}, \ldots, p_{n,N}]^T \in \{0,1\}^N$ denote a binary mask at time n, where $p_{n,i} = 1$ if the i-th entry of $\mathbf{y}_n$ is observed, and $p_{n,i} = 0$ otherwise. Clearly, in the missing data case, it is possible that the subspace may not be identifiable if the observation masks $\{\mathbf{p}_n\}$ are not well posed. For example, if all $\mathbf{p}_{n,l} = 0$, then it is not possible to recover the lth row of the Poisson rate nor the subspace. In [25], observation patterns that allow finite and unique completion of a low-rank matrix are characterized, and it is shown that a uniform random sampling scheme guarantees uniqueness as long as each entry is observed with a sufficiently large probability.

Given the sequential observations $\{\mathbf{y}_n\}_{n=1}^M$ (in the full observation case) or $\{\mathbf{p}_n \odot \mathbf{y}_n, \mathbf{p}_n\}_{n=1}^M$ (in the partial observation case), the goal is to recover and track the unknown subspace matrix $\{\mathbf{D}_n\}_{n=1}^M$, and corresponding rate vectors $\{\mathbf{z}_n\}_{n=1}^M$ in an online fashion. Clearly, the problem is not uniquely identifiable, due to the fact that we can always rewrite $\mathbf{D}_n\mathbf{a}_n = (k\mathbf{D}_n\mathbf{Q})(k^{-1}\mathbf{Q}^T\mathbf{a}_n)$, where k is an arbitrary positive scalar and $\mathbf{Q} \in \mathbb{R}^{K\times K}$ is an orthonormal matrix. Hence, subspace recovery should be interpreted in the sense that the subspace spanned by the columns of $\mathbf{D}_n$ is accurately recovered. In particular, we wish that the developed algorithms have small computational and memory footage, whose complexities do not grow with time.

III. Proposed Algorithms based on Stochastic Approximation

We start with the full observation case, and will develop the extension to the partial observation case in Section III-C. At each time n, given the Poisson model, we manifest a loss function with respect to $\mathbf{y}_n$ and $\mathbf{D} \in \mathbb{R}_+^{N\times K}$, defined as

$$\ell(\mathbf{y}_n, \mathbf{D}) := \min_{\mathbf{a}_n\in\mathbb{R}_+^K} -\log \mathrm{Pois}(\mathbf{y}_n; \mathbf{D}\mathbf{a}_n) + \lambda\|\mathbf{D}\|_F^2 + \mu\|\mathbf{a}_n\|_2^2, \tag{3}$$

where $\log \mathrm{Pois}(\mathbf{y}_n; \mathbf{D}\mathbf{a}_n)$ denotes the vector Poisson log-likelihood function with the rate vector $\mathbf{D}\mathbf{a}_n$, i.e.

$$\log \mathrm{Pois}(\mathbf{y}_n; \mathbf{D}\mathbf{a}_n) = \sum_{i=1}^N \log \mathrm{pois}(y_{n,i}; \mathbf{d}_i^T\mathbf{a}_n), \tag{4}$$

and $\log \mathrm{pois}(\cdot)$ denotes the log-likelihood function of the scalar Poisson distribution, $\mathbf{d}_i^T \in \mathbb{R}_+^{1\times K}$ is the ith row of $\mathbf{D}$, $i = 1,\ldots,N$, $\lambda, \mu > 0$ are preset regularization parameters, and the terms $\|\mathbf{D}\|_F^2$ and $\|\mathbf{a}_n\|_2^2$ are Tikhonov regularization terms that control the Frobenius norm of the subspace and ℓ_2 norm of the associated coefficients. We note that these Tikhonov regularization terms will help to fix the scaling ambiguity in the model.

Motivated by the formulation in [23], let us first formulate the expected loss minimization problem by assuming $\mathbf{D}_n = \mathbf{D}$ is fixed throughout time n. This will be used to motivate the stochastic approximation algorithms below and to benchmark performances. Define the expected loss defined as

$$f(\mathbf{D}) := \mathbb{E}_{\mathbf{y}_n}[\ell(\mathbf{y}_n, \mathbf{D})], \tag{5}$$

where the expectation is evaluated over the Poisson distribution of $\mathbf{y}_n$. We then aim to recover $\mathbf{D}$ by the following expected loss minimization problem under a nonnegative constraint:

$$\hat{\mathbf{D}} = \operatorname*{argmin}_{\mathbf{D}\in\mathbb{R}_+^{N\times K}} f(\mathbf{D}). \tag{6}$$

Once $\hat{\mathbf{D}}$ is obtained from (6), the coefficient $\hat{\mathbf{a}}_n$ can be derived via

$$\hat{\mathbf{a}}_n = \operatorname*{argmin}_{\mathbf{a}_n\in\mathbb{R}_+^K} -\log \mathrm{Pois}(\mathbf{y}_n; \hat{\mathbf{D}}\mathbf{a}_n) + \mu\|\mathbf{a}_n\|_2^2, \tag{7}$$

and the rate vector can be estimated as $\hat{\mathbf{z}}_n = \hat{\mathbf{D}}\hat{\mathbf{a}}_n$.

A. A Stochastic Approximation Algorithm

The problem in (6) is a non-convex stochastic programming, and we seek its solution via leveraging the stochastic approximation (SA) framework [23]. We first define the empirical loss at time t as

$$f_t(\mathbf{D}) := \frac{1}{t}\sum_{n=1}^t \ell(\mathbf{y}_n, \mathbf{D}). \tag{8}$$

By the strong law of large number, $f_t(\mathbf{D}) \to f(\mathbf{D})$ almost surely (a.s.) as $t \to \infty$.

At each time t, we aim to approximate the problem (6) via replacing the objective function $f(\mathbf{D})$ by the empirical loss $f_t(\mathbf{D})$. Hence, problem (6) can be approximated as

$$\hat{\mathbf{D}}_t = \operatorname*{argmin}_{\mathbf{D}\in\mathbb{R}_+^{N\times K}} \frac{1}{t}\sum_{n=1}^t \min_{\mathbf{a}_n\in\mathbb{R}_+^K} \left[-\log \mathrm{Pois}(\mathbf{y}_n; \mathbf{D}\mathbf{a}_n) + \mu\|\mathbf{a}_n\|_2^2\right] + \lambda\|\mathbf{D}\|_F^2. \tag{9}$$

Invoking the stochastic approximation framework, we aim to solve problem (9) by alternating between two steps, *nonnegative encoding* and *subspace update*. Specifically, at time t, we first learn the coefficient vector $\hat{\mathbf{a}}_t$, given the new data $\mathbf{y}_t$ and previously learned subspace $\hat{\mathbf{D}}_{t-1}$. Namely, the estimate $\hat{\mathbf{a}}_t$ is obtained by minimizing the loss function:

$$\hat{\mathbf{a}}_t = \operatorname*{argmin}_{\mathbf{a}\in\mathbb{R}_+^K} -\log \mathrm{Pois}(\mathbf{y}_t; \hat{\mathbf{D}}_{t-1}\mathbf{a}) + \mu\|\mathbf{a}\|_2^2. \tag{10}$$

Once we obtain $\hat{\mathbf{a}}_t$, the subspace $\hat{\mathbf{D}}_t$ is then updated by minimizing, based on previous estimates $\{\hat{\mathbf{a}}_n\}_{n=1}^t$ and observations $\{\mathbf{y}_n\}_{n=1}^t$, the following:

$$\hat{\mathbf{D}}_t = \operatorname*{argmin}_{\mathbf{D}\in\mathbb{R}_+^{N\times K}} \left\{-\frac{1}{t}\sum_{n=1}^t \log \mathrm{Pois}(\mathbf{y}_n; \mathbf{D}\hat{\mathbf{a}}_n) + \lambda\|\mathbf{D}\|_F^2\right\}. \tag{11}$$

Owing to (4), (11) can be decomposed into a set of smaller problems for each row of the subspace matrix. Specifically, the ith row of $\mathbf{D}$ can be updated in parallel as

$$\hat{\mathbf{d}}_{t,i} = \operatorname*{argmin}_{\mathbf{d}_i\in\mathbb{R}_+^K} -\frac{1}{t}\sum_{n=1}^t \log \mathrm{pois}(y_{n,i}; \mathbf{d}_i^T\hat{\mathbf{a}}_n) + \lambda\|\mathbf{d}_i\|_2^2. \tag{12}$$

We summarize the proposed stochastic approximate (SA) algorithm in Algorithm 1. Both (10) and (12) can be solved efficiently via projected gradient descent. Below we discuss the details for solving (10), and (12) can be solved similarly. Specifically, we find $\hat{\mathbf{a}}_t$ iteratively and at the $(k+1)$-th iteration, the update is calculated as

$$\hat{\mathbf{a}}_t^{(k+1)} = \mathrm{Proj}\left(\hat{\mathbf{a}}_t^{(k)} - \alpha_k\nabla g(\hat{\mathbf{a}}_t^{(k)})\right),$$

where $g(\mathbf{a}) = -\log \mathrm{Pois}(\mathbf{y}_t; \hat{\mathbf{D}}_{t-1}\mathbf{a}) + \mu\|\mathbf{a}\|_2^2$ and $\mathrm{Proj}(\mathbf{a}) := \max\{\mathbf{a}, 0\}$ is the projection operator, where $\max$ operator denotes the entry-wise maximization. Moreover, α_k is the step size and can be set as [26]

$$\alpha_k = \frac{\left(\hat{\mathbf{a}}_t^{(k)} - \hat{\mathbf{a}}_t^{(k-1)}\right)^T \left[\nabla g(\hat{\mathbf{a}}_t^{(k)}) - \nabla g(\hat{\mathbf{a}}_t^{(k-1)})\right]}{\left\|\nabla g(\hat{\mathbf{a}}_t^{(k)}) - \nabla g(\hat{\mathbf{a}}_t^{(k-1)})\right\|_2^2},$$

and a random initial point $\hat{\mathbf{a}}_t^{(0)}$ of the gradient descent is employed. The regularization parameters λ, μ can be empirically

Algorithm 1 Stochastic Approximation (SA) for Poisson Streaming Data

```
Input: Data $\{\mathbf{y}_n\}_{n=1}^M$, $\lambda$, $\mu$, initialization $\mathbf{D}_0$
Output: Subspace estimates $\{\hat{\mathbf{D}}_t\}_{t=1}^M$ and $\{\hat{\mathbf{a}}_t\}_{t=1}^M$
for $t = 1$ to $M$ do
    Estimate the coefficient $\hat{\mathbf{a}}_t$ by the following optimization via projected gradient descent:
        $\hat{\mathbf{a}}_t = \operatorname*{argmin}_{\mathbf{a} \in \mathbb{R}_+^K} -\log \mathrm{Pois}(\mathbf{y}_t; \hat{\mathbf{D}}_{t-1}\mathbf{a}) + \mu \|\mathbf{a}\|_2^2;$
    Update each row of the subspace $\hat{\mathbf{D}}_t$ by the following optimization via projected gradient descent:
        $\hat{\mathbf{d}}_{t,i} = \operatorname*{argmin}_{\mathbf{d}_i \in \mathbb{R}_+^K} -\frac{1}{t}\sum_{n=1}^{t} \log \mathrm{pois}(y_{n,i}; \mathbf{d}_i^T \hat{\mathbf{a}}_n) + \lambda \|\mathbf{d}_i\|_2^2.$
end for
```

determined via cross-validation and a random initialization $\mathbf{D}_0$ can be utilized.

B. Memory-Limited Stochastic Approximation

The SA algorithm in Algorithm 1 allows us to update the subspace in an online fashion as new data arrives. However, a closer examination suggests that its implementation requires storing all previous $\{\hat{\mathbf{a}}_n\}_{n=1}^t$ and $\{\mathbf{y}_n\}_{n=1}^t$ in order to perform the subspace update (12), yielding a significant memory overhead that grows linearly as t increases. This is in sharp contrast to the Gaussian case [11], [12], where the log-likelihood is a quadratic term that can be efficiently implemented by only storing sufficient statistics of previous data whose size does not grow with time.

Our goal in this section is to derive a memory-limited SA algorithm for Poisson data that only demands storing sufficient statistics of previous data, which is more appealing for streaming applications. Unfortunately, the Poisson log-likelihood function prohibits such an easy adaptation. Rather than dealing with the original Poisson log-likelihood function, we will establish its upper bound which is more amenable for memory-limited implementations. In order to facilitate the derivation, we make the following two technical assumptions:

A1) $\mathbf{D} \in \mathcal{C}_2$ where $\mathcal{C}_2 \subset \mathbb{R}_+^{N \times K}$ is a compact set.

A2) There exist positive constants a and b such that $0 < a \le \mathbf{d}_i^T \mathbf{a}_n \le b$, for all n and i. In other words, we assume that entries of the rate vector $\mathbf{z}_n = \mathbf{D}\mathbf{a}_n$ are bounded.

The first one essentially assumes that $\mathbf{D}$ has bounded entries and this is a very mild assumption and almost always valid for real applications. The second assumption is used to exclude the singular case, where some Poisson rates approach zero. Similar assumptions have also been utilized in [3], [18] under the Poisson CS framework. Note that by our assumptions, the minimization in (3) is only obtained on a compact domain of $\mathbf{a}_n$. Therefore, we can further restrict $\mathbf{a}_n$ to be supported on the compact set $\mathcal{C}_3$, which will not alter the solution to the minimization.

We can now upper bound the Poisson log-likelihood function in the following proposition whose proof is presented in Appendix A.

Algorithm 2 Memory-Limited Stochastic Approximation for Poisson Streaming Data

```
Input: Data $\{\mathbf{y}_n\}_{n=1}^M$, $\lambda$, $\mu$, initialization $\mathbf{D}_0$, $\mathbf{s}_0 = 0$, $\beta_{0,i} = 0$ and $\mathbf{r}_{0,i} = 0$ for all $1 \le i \le N$.
Output: Subspace estimates $\{\hat{\mathbf{D}}_t\}_{t=1}^M$ and $\{\hat{\mathbf{a}}_t\}_{t=1}^M$
for $t = 1$ to $M$ do
    Estimate the coefficient $\hat{\mathbf{a}}_t$ by the following optimization via projected gradient descent
        $\hat{\mathbf{a}}_t = \operatorname*{argmin}_{\mathbf{a} \in \mathbb{R}_+^K} -\log \mathrm{Pois}(\mathbf{y}_t; \hat{\mathbf{D}}_{t-1}\mathbf{a}) + \mu \|\mathbf{a}\|_2^2.$
    Update the sufficient statistics, for $1 \le i \le N$, as
        $\mathbf{s}_t = \frac{t-1}{t}\mathbf{s}_{t-1} + \frac{1}{t}\hat{\mathbf{a}}_t,$   (15)
        $\beta_{t,i} = \frac{t-1}{t}\beta_{t,i} + \frac{1}{t}y_{t,i},$   (16)
        $\mathbf{r}_{t,i} = \mathbf{r}_{t-1,i} + \hat{\mathbf{a}}_t y_{t,i};$   (17)
    Update each row of the subspace $\hat{\mathbf{D}}_t$ by the following optimization via projected gradient descent
        $\hat{\mathbf{d}}_{t,i} = \operatorname*{argmin}_{\mathbf{d}_i \in \mathbb{R}_+^K} \mathbf{d}_i^T \mathbf{s}_t - \beta_{t,i} \log(\mathbf{d}_i^T \mathbf{r}_{t,i}) + \lambda \|\mathbf{d}_i\|_2^2.$
end for
```

Proposition 1. *Under assumptions A1) and A2), we have the following bound for every $1 \le i \le N$ and t:*

$$\begin{aligned} -\sum_{n=1}^{t} \log \mathrm{pois}(y_{n,i}; \mathbf{d}_i^T \mathbf{a}_n) &\le \mathbf{d}_i^T \left(\sum_{n=1}^{t} \mathbf{a}_n\right) \\ &- \left(\sum_{n=1}^{t} y_{n,i}\right) \log\left[\mathbf{d}_i^T \left(\sum_{n=1}^{t} \mathbf{a}_n y_{n,i}\right)\right] + \sum_{n=1}^{t} \log(y_{n,i}!) \\ &+ \left(\sum_{n=1}^{t} y_{n,i}\right) \cdot \left[\log\left(\sum_{n=1}^{t} y_{n,i}\right) + T\right], \end{aligned} \tag{13}$$

where T is a constant only depending on a and b, as assumed in A2.

Replacing the log-likelihood term by the above upper bound, at each time t, we propose to update the ith row of $\mathbf{D}$ via the following optimization problem:

$$\hat{\mathbf{d}}_{t,i} = \operatorname*{argmin}_{\mathbf{d}_i \in \mathbb{R}_+^K} \mathbf{d}_i^T \mathbf{s}_t - \beta_{t,i} \log(\mathbf{d}_i^T \mathbf{r}_{t,i}) + \lambda \|\mathbf{d}_i\|_2^2. \tag{14}$$

where $\mathbf{s}_t = \frac{1}{t}\sum_{n=1}^{t} \hat{\mathbf{a}}_n$, $\beta_{t,i} = \frac{1}{t}\sum_{n=1}^{t} y_{n,i}$, and $\mathbf{r}_{t,i} = \sum_{n=1}^{t} \hat{\mathbf{a}}_n y_{n,i}$. To implement (14), it is sufficient to update $\mathbf{s}_t$, $\beta_{t,i}$ and $\mathbf{r}_{t,i}$ as new data arrives in a low-complexity fashion as done in (15), (16), therefore it can be implemented in a memory-limited manner. Putting everything together, we obtain the memory-limited SA algorithm for Poisson streaming data, summarized in Algorithm 2.

Remark: If we apply the Jensen's inequality to $\sum_{n=1}^{t} \log \mathrm{pois}(y_{n,i}; \mathbf{d}_i^T \mathbf{a}_n)$, we obtain a lower bound of the Poisson log-likelihood function:

$$-\frac{1}{t}\sum_{n=1}^{t}\log\mathrm{pois}(y_{n,i};\mathbf{d}_i^T\mathbf{a}_n) \geq \frac{1}{t}G_t - T\left(\frac{1}{t}\sum_{n=1}^{t} y_{n,i}\right),$$

where G_t is a short-hand notation for the right-hand side of (13). Putting the above lower bound and the upper bound (13) in Proposition 1 together, it is straightforward to observe that the gap of the bounds is $T\left(\frac{1}{t}\sum_{n=1}^{t} y_{n,i}\right)$, which approaches to a constant when t goes to infinity. As we present in Section V, it is observed that the performance of the memory-limited SA algorithm is close to the original SA algorithm, suggesting that these bounds are empirically tight for most numerical experiments.

C. Extension to Handle Missing Data

In this section, we discuss how to extend the proposed SA algorithms to handle missing data, when the data stream is only partially observed. To begin with, we modify the empirical loss minimization problem in (9) as

$$\hat{\mathbf{D}}_t = \operatorname*{argmin}_{\mathbf{D}\in\mathbb{R}_+^{N\times K}} \frac{1}{t}\sum_{n=1}^{t}\min_{\mathbf{a}_n}\Big[-\sum_{i=1}^{N} p_{n,i}\log\mathrm{pois}(y_{n,i};\mathbf{d}_i^T\mathbf{a}_n) + \mu\|\mathbf{a}_n\|_2^2\Big] + \lambda\|\mathbf{D}\|_F^2. \quad (18)$$

The above can be solved in a similar fashion as described in Algorithm 1, where the coefficient $\hat{\mathbf{a}}_t$ is estimated via

$$\hat{\mathbf{a}}_t = \operatorname*{argmin}_{\mathbf{a}\in\mathbb{R}_+^K} -\sum_{i=1}^{N} p_{t,i}\log\mathrm{Pois}(y_{t,i};\mathbf{d}_i^T\mathbf{a}) + \mu\|\mathbf{a}\|_2^2, \quad (19)$$

i.e. only the observed entries contribute to the loss function. To obtain the memory-limited SA algorithm, we still take the two assumptions as in previous section and have the following upper bound.

Proposition 2. *With previous assumptions, we have the following bound for every $1 \leq i \leq N$ and t:*

$$\begin{aligned}-\sum_{n=1}^{t} p_{n,i}\log\mathrm{pois}(y_{n,i};\mathbf{d}_i^T\mathbf{a}_n) &\leq \mathbf{d}_i^T\left(\sum_{n=1}^{t} p_{n,i}\mathbf{a}_n\right)\\ &- \left(\sum_{n=1}^{t} p_{n,i}y_{n,i}\right)\log\left[\mathbf{d}_i^T\left(\sum_{n=1}^{t} p_{n,i}y_{n,i}\mathbf{a}_n\right)\right]\\ &+ \sum_{n=1}^{t} p_{n,i}\log(y_{n,i}!)\\ &+ \left(\sum_{n=1}^{t} p_{n,i}y_{n,i}\right)\left[\log\left(\sum_{n=1}^{t} p_{n,i}y_{n,i}\right) + T\right], \quad (20)\end{aligned}$$

where T is a constant depending on a and b, as assumed in A2.

Replacing the log-likelihood function by the above upper bound, then the rows of the subspace $\mathbf{D}$ can be similarly updated in parallel as

$$\hat{\mathbf{d}}_{t,i} = \operatorname*{argmin}_{\mathbf{d}_i\in\mathbb{R}_+^K} \mathbf{d}_i^T\widetilde{\mathbf{s}}_{t,i} - \widetilde{\beta}_{t,i}\log(\mathbf{d}_i^T\widetilde{\mathbf{r}}_{t,i}) + \lambda\|\mathbf{d}_i\|_2^2. \quad (21)$$

where $\widetilde{\mathbf{s}}_{t,i} = \frac{1}{t}\sum_{n=1}^{t} p_{n,i}\hat{\mathbf{a}}_n$, $\widetilde{\beta}_{t,i} = \frac{1}{t}\sum_{n=1}^{t} p_{n,i}y_{n,i}$, and $\widetilde{\mathbf{r}}_{t,i} = \sum_{n=1}^{t}\hat{\mathbf{a}}_n p_{n,i}y_{n,i}$. Hence, we can formulate a memory-limited SA algorithm with missing data that alternates between estimating $\hat{\mathbf{a}}_t$ and updating $\hat{\mathbf{D}}_t$, which is summarized in Algorithm 3.

Algorithm 3 Memory-Limited Stochastic Approximation for Poisson Streaming Data with Missing Data

```
Input: Data $\{\mathbf{y}_n\}_{n=1}^M$, $\lambda$, $\mu$, initialization $\mathbf{D}_0$, $\widetilde{\mathbf{s}}_{0,i} = 0$, $\widetilde{\beta}_{0,i} = 0$ and $\widetilde{\mathbf{r}}_{0,i} = 0$ for all $1 \leq i \leq N$.
Output: Subspace estimates $\{\hat{\mathbf{D}}_t\}_{t=1}^M$ and $\{\hat{\mathbf{a}}_t\}_{t=1}^M$
for $t = 1$ to $M$ do
    Estimate the coefficient $\hat{\mathbf{a}}_t$ by the following optimization via projected gradient descent
        $\hat{\mathbf{a}}_t = \operatorname*{argmin}_{\mathbf{a}\in\mathbb{R}_+^K} -\sum_{i=1}^{N} p_{t,i}\log\mathrm{pois}(y_{t,i};\mathbf{d}_i^T\mathbf{a}) + \mu\|\mathbf{a}\|_2^2;$
    Update the sufficient statistics, for $1 \leq i \leq N$, as
        $\widetilde{\mathbf{s}}_{t,i} = \frac{t-1}{t}\widetilde{\mathbf{s}}_{t-1,i} + \frac{1}{t}p_{t,i}\hat{\mathbf{a}}_t,$   (22)
        $\widetilde{\beta}_{t,i} = \frac{t-1}{t}\beta_{t,i} + \frac{1}{t}p_{t,i}y_{t,i},$   (23)
        $\widetilde{\mathbf{r}}_{t,i} = \widetilde{\mathbf{r}}_{t-1,i} + \hat{\mathbf{a}}_t p_{t,i}y_{t,i};$   (24)
    Update each row of the subspace $\hat{\mathbf{D}}_t$ by the following optimization via projected gradient descent,
        $\hat{\mathbf{d}}_{t,i} = \operatorname*{argmin}_{\mathbf{d}_i\in\mathbb{R}_+^K} \mathbf{d}_i^T\widetilde{\mathbf{s}}_{t,i} - \widetilde{\beta}_{t,i}\log(\mathbf{d}_i^T\widetilde{\mathbf{r}}_{t,i}) + \lambda\|\mathbf{d}_i\|_2^2.$
end for
```

IV. Convergence Analysis

In this section, we provide a convergence analysis for the proposed SA algorithm in Algorithm 1 assuming $\mathbf{D}_t = \mathbf{D}$ is fixed. We first define

$$\ell'(\mathbf{y}_n, \mathbf{D}, \mathbf{a}) := -\log\mathrm{Pois}(\mathbf{y}_n; \mathbf{D}\mathbf{a}) + \lambda\|\mathbf{D}\|_F^2 + \mu\|\mathbf{a}\|_2^2, \quad (25)$$

and

$$f_t'(\mathbf{D}) := \frac{1}{t}\sum_{n=1}^{t}\ell'(\mathbf{y}_n, \mathbf{D}, \hat{\mathbf{a}}_n), \quad (26)$$

where $\hat{\mathbf{a}}_n$ is the output of Algorithm 1. Note that $f_t'(\mathbf{D})$ captures the empirical loss of the SA algorithm.

In order to facilitate the convergence analysis, in addition to the assumptions A1) and A2) made in Section III-B, we make an additional assumption:

A3) The observations $\{\mathbf{y}_n\}$ are supported on a compact set $\mathcal{C}_1$.

The assumption A3) essentially assumes that the observed data is bounded. Since $\mathbf{D}\mathbf{a}_n$ is bounded above by A1), it is straightforward to check that a realization of $\{\mathbf{y}_n\}$ are upper bounded with a probability controlled by the bound. When the upper bound is set large enough, A3) holds with a high probability. Therefore, one can apply a hard-threshold on the data $\{\mathbf{y}_n\}$ with a preset large upper bound. In addition, such a bounded assumption is naturally satisfied for real data.

Our first theorem states the almost sure convergence of the empirical loss $\{f_t'(\hat{\mathbf{D}}_t)\}$ of Algorithm 1, the objective function $\{f_t(\hat{\mathbf{D}}_t)\}$ of Algorithm 1, and the expected loss $\{f(\hat{\mathbf{D}}_t)\}$ in (5) converge to the same limit, where $\{\hat{\mathbf{D}}_t\}$ is the sequence output of Algorithm 1. All the proofs are presented in Appendix B.

Theorem 1. *With all previous assumptions, the stochastic processes* $\{f_t(\hat{\mathbf{D}}_t)\}$,$\{f_t'(\hat{\mathbf{D}}_t)\}$ *and* $\{f(\hat{\mathbf{D}}_t)\}$ *converge a.s. to the same limit.*

In addition, our second theorem states that the estimated subspace $\hat{\mathbf{D}}_t$ also almost surely converges to a local minimum of $f(\mathbf{D})$.

Theorem 2. *With all previous assumptions, consider a sequence* $\{\hat{\mathbf{D}}_t\}$ *such that Theorem 1 holds. Then with probability* 1*,* $\hat{\mathbf{D}}_t$ *converges to a local minimum of the expected loss* $f(\mathbf{D})$.

Different from previous works [12], [27] where the proof directly aims for the stochastic sequence $\{\hat{\mathbf{D}}_t\}$, inspired by [28], we show the convergence of $\{\hat{\mathbf{D}}_t\}$ by charactering all convergent subsequences of $\{\hat{\mathbf{D}}_t\}$.

Unfortunately, the proof techniques for Theorem 1 and 2 can only be applied to Algorithm 1, and cannot be easily adapted to the memory-limited versions. However, Theorem 1 and 2 still serve as a convergence indicator of the memory-limited versions, provided that they yield similar performance in the numerical experiments in Section V.

V. Numerical Experiments

In this section, we showcase the performance of the proposed two algorithms, *i.e.,* SA and the memory-limited SA algorithms, for both the full observation and partial observation cases. We conduct experiments on both synthetic and real video data, as well as document analysis.

A. Experiments with Synthetic Data

Let $N = 100$ and $K = 10$. We generate synthetic data $\mathbf{y}_n \sim \text{Pois}(\mathbf{D}\mathbf{a}_n)$ *i.i.d.*, where the entries of $\mathbf{D}$ and $\{\mathbf{a}_n\}$ are randomly drawn from the uniform distribution on $[0, 1]$. Furthermore, assume the entries of the partial observation mask $\mathbf{p}_n$ are also generated i.i.d. using the Bernoulli distribution with the parameter $0 < p \leq 1$. The normalized subspace reconstruction error is used to measure the performance, and it is calculated as $\|P_{\hat{\mathbf{D}}_t^\perp}\mathbf{D}\|_F/\|\mathbf{D}\|_F$ where $P_{\hat{\mathbf{D}}_t^\perp}$ is the projection operator onto the orthogonal complement of the subspace estimate $\hat{\mathbf{D}}_t$ at time t. We randomly pick the initialization and set the regularization parameters $\lambda = 0.2$, $\mu = 0.1$ and the length of the data stream $M = 800$.

We compare our results with the state-of-the-art subspace tracking algorithm in [13], referred as recursive projected compressive sensing (ReProCS) algorithm, that does not assume any Poisson noise for incoming data. Furthermore, we also compare our results to a Bayesian Poisson factor analysis (BPFA) algorithm [24], which assumes the data is of the form $\text{Pois}(\mathbf{\Phi}\mathbf{\Theta})$ where $\mathbf{\Phi}$ is the factor loading matrix and $\mathbf{\Theta}$ is the factor score matrix. A Bayesian model is considered in [24] for inference of $\mathbf{\Phi}$ and $\mathbf{\Theta}$. In addition, we also compare our algorithms to the batch Poisson matrix completion algorithm in [22] that adopts a similar low-rank assumption on the rate vectors. Since BPFA [24] and the batch algorithm in [22] are not online algorithms, we only present their performance using the entire data stream as the input. We treat the Poisson observations as the input to all algorithms.

Fig. 1 shows the normalized subspace reconstruction error with respect to the data stream index of the proposed SA, memory-limited SA, ReProCS, BPFA and batch algorithms when the data stream is fully observed. Fig. 2 shows the normalized subspace reconstruction error with respect to the data stream index for various algorithms when the data is partially observable with different probabilities of observations. Note that BPFA cannot handle missing data and is omitted in this case. It can be seen that for both cases, the subspace estimates of both SA and memory-limited SA algorithms improve with the increase of time index, and provides much better estimates than the ReProCS algorithm, which produces very poor results. Moreover, the memory-limited SA algorithm yields a very similar performance towards the SA algorithm with a smaller complexity. The proposed algorithms also achieves better performance than the BPFA algorithm and yields similar performances of the batch algorithm when the data index is large enough. Furthermore, the running time is plotted against the normalized subspace error in Fig. 3. It can be found that the proposed algorithms are at least competitive, compared to other algorithms under considerations.

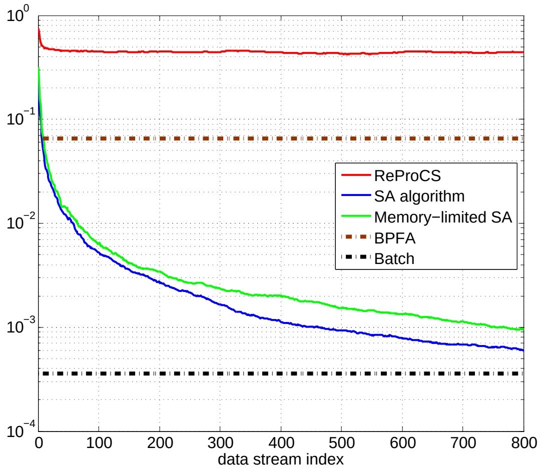
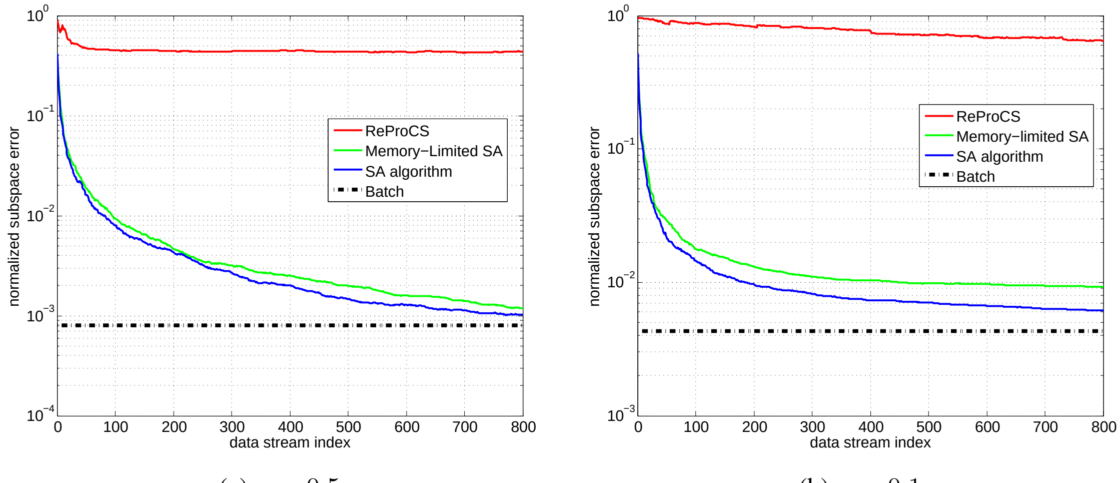


Figure 1. The normalized subspace reconstruction errors for SA, memory-limited SA, ReProCS, BPFA and batch algorithms when $N = 100$, $K = 10$ and data is fully-observable.

We further examine the performance of the proposed algorithms when the rank of the subspace K is over-specified. This is common since the subspace rank might not be perfectly known. In practice, it is common to have an over-estimation of the rank. Fig. 4 shows the normalized subspace errors with respect to the data stream index for the SA algorithm when the subspace rank is over-estimated, under the same setup of the previous experiments. Similarly, Fig. 5 shows the performance of the memory-limited SA algorithm. It can be found that both of the proposed algorithms still converge nicely even when the



(a) $p = 0.5$

(b) $p = 0.1$

Figure 2. The normalized subspace reconstruction errors for SA, memory-limited SA, ReProCS and batch algorithms when $N = 100$, $K = 10$, and the probability of observing each entry is (a) $p = 0.5$, and (b) $p = 0.1$.

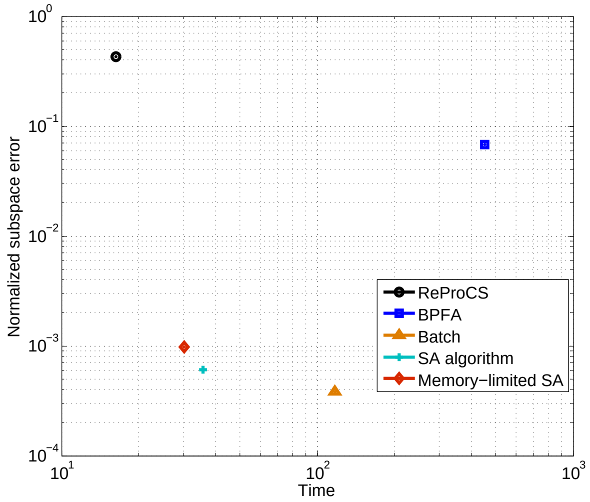


Figure 3. The running time versus the normalized subspace reconstruction errors for SA, memory-limited SA, ReProCS, BPFA and batch algorithms when $N = 100$, $K = 10$ and the data stream is fully observed.

rank is not perfectly known, and the error increases gracefully as we increase the rank.

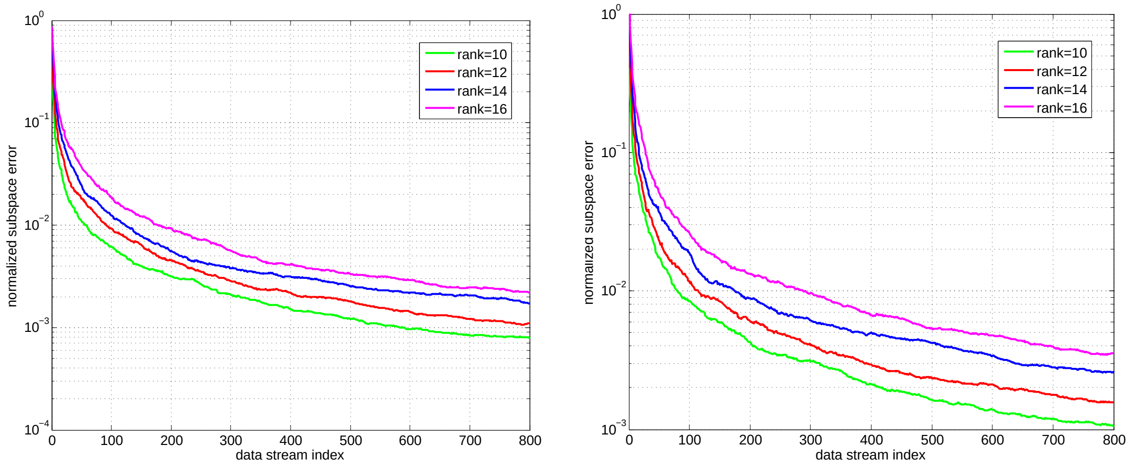
We also present the performance of the proposed algorithms when the underlying subspace changes abruptly. Let $N = 100$, $K = 10$ and $M = 2600$. We generate the data $\mathbf{y}_n \sim \text{Pois}(\mathbf{D}\mathbf{a}_n)$ similarly as earlier. From time index 1 to 700, a realization of $\mathbf{D}$ is used first, and we constitute new realizations of $\mathbf{D}$ at time steps 701, 1201 and 1801, representing sudden changes of the underlying subspace. Fig. 6 demonstrates the normalized subspace error with respect to the data stream index for both the SA and memory-limited SA algorithms. It can be seen that both algorithms have successfully tracked the subspace when changes occur.

In order to examine the effect of random initialization to the proposed algorithms, we showcase the performance of the proposed algorithms where $N = 200$, $K = 20$, $M = 800$ and 50 Monte-Carlo simulations are manifested. Fig. 7 presents the mean and variance of the normalized subspace error under the Monte-Carlo simulations. It can be seen that the overall performance is not very sensitive to the random initialization.

B. *Experiments with Real Video Data*

We apply the proposed algorithms on real video sequences under Poisson noise. The gray-scale video is of a resolution 50×50 with total 250 frames and the nth frame is regarded as a 2500-dimensional vector $\mathbf{z}_n$ of its gray scale. In order to determine the rank of the data $[\mathbf{z}_1, \ldots, \mathbf{z}_{250}]$, we use SVD to calculate the approximate rank. Hence, we set $N = 2500$, rank $K = 40$, $M = 250$, $\mu = 0.1$ and $\lambda = 0.2$. The observations are the Poisson counts $\mathbf{y}_n \sim \text{Pois}(\mathbf{z}_n)$, where each entry of $\mathbf{y}_n$ is observed independently with probability $0 < p \leq 1$. We compute the relative video reconstruction error at the nth frame as $\|\hat{\mathbf{D}}_n \hat{\mathbf{a}}_n - \mathbf{z}_n\|_2 / \|\mathbf{z}_n\|_2$. In addition to the aforementioned ReProCS [13], BPFA [24] and batch [22] algorithms, we compare our results to a Poisson dynamic model referred as dynamic mirror descent (DMD) in [5] and the online Robust PCA (RPCA) algorithm in [12].

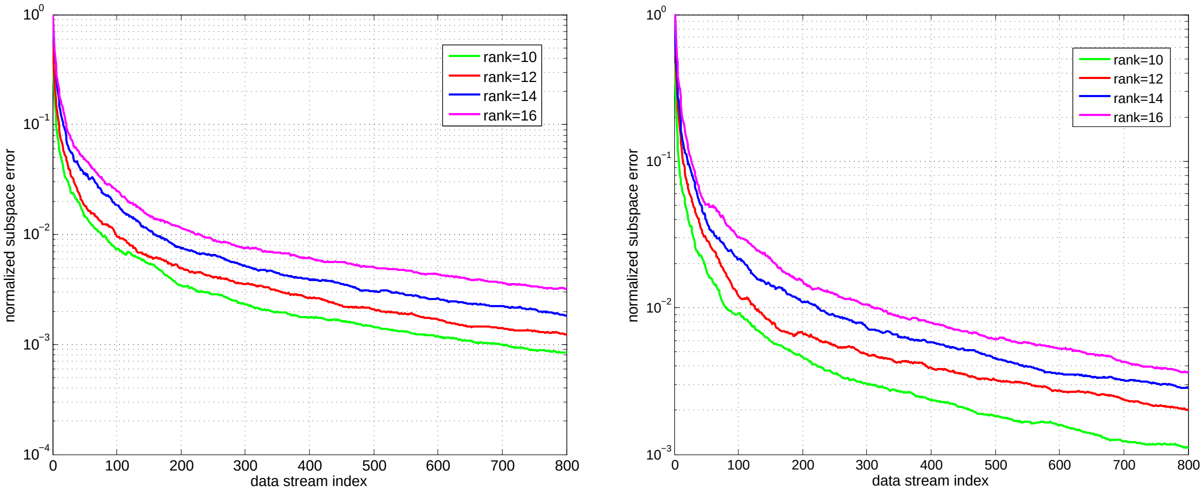
In Fig. 8, we illustrate the original video frame, its observation and recovery by various algorithms. Fig. 9 presents the relative errors of the recovered video frames via the SA, the memory-limited SA, DMD, online RPCA, BPFA and batch algorithms when $p = 1$ and $p = 0.5$, respectively. We note that BPFA, online RPCA, and DMD algorithms cannot directly handle the missing data scenarios and are omitted for the case with missing data. It is demonstrated that the performance improves with the increase of the data stream index, and approaches the performance of the batch algorithm by only processing each data vector once. We do not show the performance of the ReProCS algorithm here, since its performance is so poor that the relative error is significantly



(a) $p = 1$

(b) $p = 0.5$

Figure 4. The normalized subspace reconstruction errors for the SA algorithm under various settings of input rank, where $N = 100$ and the true rank $K = 10$, for (a) fully observed data with $p = 1$ and (b) partially observed data with $p = 0.5$.



(a) $p = 1$

(b) $p = 0.5$

Figure 5. The normalized subspace reconstruction errors for the memory-limited SA algorithm under various settings of input rank, where $N = 100$ and the true rank $K = 10$, for (a) fully observed data with $p = 1$ and (b) partially observed data with $p = 0.5$

larger and does not improve with the increase of the time index.

C. Application on Document Analysis

In addition to previous experiments on imagery application, we apply the proposed SA algorithms to document analysis. Specifically, we consider the *State of the Union* addresses from year 1790 to 2014, total $M = 225$ transcripts. For each address of year n, we first convert it to a vector $\mathbf{y}_n$ with each entry representing the counts of words, from a vocabulary V. Akin to [29], we pre-process the data by removing stop words and terms which appear less than 20 times in the address, yielding a V of size $N = 2216$. Hence, each address $\mathbf{y}_n$ is represented by Poisson counts via a linear combination of columns of the underlying topics $\mathbf{D}$, *i.e.*, $\mathbf{y}_n \sim \text{Pois}(\mathbf{D}\mathbf{a}_n)$. We wish to infer the underlying topic matrix $\mathbf{D}$ as well as the associated topic weight $\mathbf{a}_n$. Throughout this experiment, we set $K = 50$, $\mu = 0.1$ and $\lambda = 0.15$.

In Table I, we list 4 examples of inferred topics, and their associated top words. For each topic i, represented by the i-th column of $\mathbf{D}$, its presence is reflected via the intensity $\frac{\mathbf{a}_{n,i}}{\sum_i \mathbf{a}_{n,i}}$. In Fig. 10, we plot how the intensities of these 4 topics change over time. We can see that topic 1 seems to associate with the Afghanistan and Iraq wars and its intensity strongly presents after year 2001. Topic 2 is related to two world wars and its intensity reaches the highest point accordingly. Topic 3 is with the National Energy Program and topic 4 focuses on the Philippine-American war. It is straightforward to find that the inferred intensity dynamically tracks the topics mentioned in these addresses.

VI. Conclusion

We have considered the problem of recovering and tracking the underlying Poisson rate from streaming count data, where

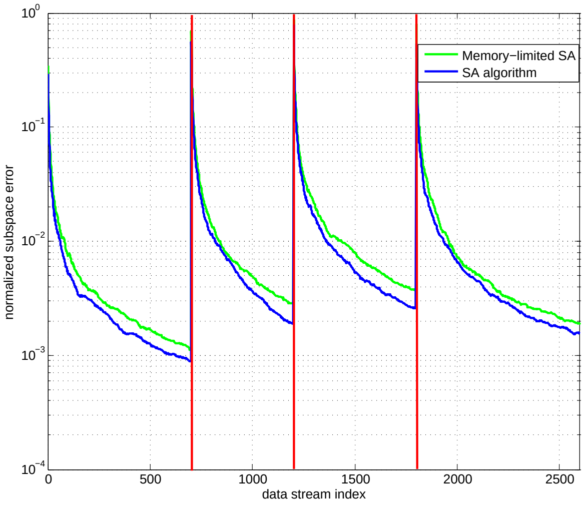


Figure 6. The normalized subspace reconstruction errors of the proposed SA and memory-limited algorithms when abrupt changes of the subspace occur. The 3 red vertical lines mark the time steps when the underlying subspace changes.

Table I
4 EXAMPLES OF INFERRED TOPICS AND THEIR ASSOCIATED TOP WORDS

Topic 1	Topic 2	Topic 3	Topic 4
Iraq	war	energy	island
terrorists	enemy	nuclear	islands
terror	Japanese	development	Philippine
Afghanistan	German	policy	military

the rate vectors have been posed to lie in a low-dimensional subspace. A stochastic programming approach has been proposed to recover the underlying subspace as well as the rate vectors. A stochastic approximation algorithm has first been derived. The SA algorithm has been decomposed into two steps where the subspace and its coefficients are iteratively updated as new data arrives. Theoretical convergence guarantees have been established for the SA algorithm under certain mild assumptions. The SA algorithm has been proved to converge to the same point as the original expected loss minimization problem. In addition, the estimated subspace has been shown to converge to a local minimum of the original expected loss minimization problem. In order to reduce the memory requirement and handle missing data, the SA algorithm has been modified to allow a memory-limited implementation. We have demonstrated that the memory-limited SA algorithms yield similar performance to the SA algorithm. All algorithms have been showcased to achieve promising performances over both synthetic and real data.

Appendix A
Proofs of Propositions 1 and 2

We first introduce a lemma [30] which is useful later.

Lemma 1. *Let $f:\mathbb{R}\to\mathbb{R}$ be a concave function on interval $[a,b]$. Let $\{x_i\}$ and $\{p_i\}$ be collections of finite points such that $x_i \in [a,b]$, for all i and $\sum_i p_i = 1$ with all $p_i > 0$. Assuming $c,d>0$ with $c+d=1$, we have*

$$\sum p_i f(x_i) \ge f\left(\sum p_i x_i\right) - \max_c [f(ca+db) - cf(a) - df(b)]$$

It is easy to check that $[f(ca+db) - cf(a)) - df(b)]$ is a concave function, and a unique maximization depending only on a and b exists. Denote $0 \le T_{[a,b]} = \max_c [f(ca+db) - cf(a) - df(b)]$.

Proof of Proposition 1. First we write the log-likelihood function as

$$\begin{aligned}
&\sum_{n=1}^{t} \log \mathrm{pois}(y_{n,i}; \mathbf{d}_i^T \mathbf{a}_n) \\
&= \sum_{n=1}^{t} \log \frac{e^{-(\mathbf{d}_i^T \mathbf{a}_n)} (\mathbf{d}_i^T \mathbf{a}_n)^{y_{n,i}}}{y_{n,i}!} \\
&= \sum_{n=1}^{t} \left[-\mathbf{d}_i^T \mathbf{a}_n + y_{n,i} \log(\mathbf{d}_i^T \mathbf{a}_n) - \log(y_{n,i}!) \right] \\
&= -\mathbf{d}_i^T \left(\sum_{n=1}^{t} \mathbf{a}_n \right) + \sum_{n=1}^{t} y_{n,i} \log(\mathbf{d}_i^T \mathbf{a}_n) - \sum_{n=1}^{t} \log(y_{n,i}!).
\end{aligned} \tag{27}$$

The second term in (27) can be bounded as follows:

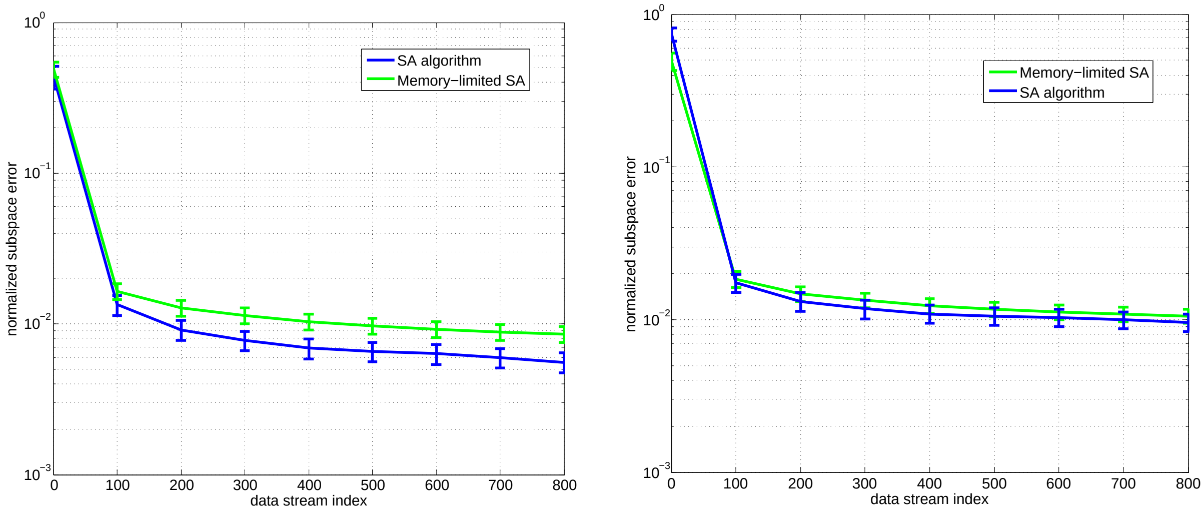
$$\begin{aligned}
&\sum_{n=1}^{t} y_{n,i} \log(\mathbf{d}_i^T \mathbf{a}_n) \\
&= \left(\sum_{n=1}^{t} y_{n,i} \right) \cdot \left[\sum_{n=1}^{t} \left(\frac{y_{n,i}}{\sum_{n=1}^{t} y_{n,i}} \right) \log(\mathbf{d}_i^T \mathbf{a}_n) \right] \\
&\ge \left(\sum_{n=1}^{t} y_{n,i} \right) \cdot \left\{ \log \left[\sum_{n=1}^{t} \frac{y_{n,i} (\mathbf{d}_i^T \mathbf{a}_n)}{\sum_{n=1}^{t} y_{n,i}} \right] - T \right\},
\end{aligned} \tag{28}$$

where we have applied Lemma 1. By our assumptions, the minimization is only obtained in a compact domain of $\mathbf{D}$ and $\mathbf{a}_n$. Therefore, $(\mathbf{d}_i^T \mathbf{a}_n)_i < b$ for some constant b for all i and n, and above inequality follows from Lemma 1 where $f(x) = \log x$. Denote the constant $T_{[a,b]}$ in Lemma 1 as $T := T_{[a,b]}$. Therefore, (13) follows by rearranging the terms in (28). □

Proof of Proposition 2. Similar to (27), we have

$$\begin{aligned}
&\sum_{n=1}^{t} p_{n,i} \log \mathrm{pois}(y_{n,i}; \mathbf{d}_i^T \mathbf{a}_n) \\
&= \sum_{n=1}^{t} p_{n,i} \log \frac{e^{-(\mathbf{d}_i^T \mathbf{a}_n)} (\mathbf{d}_i^T \mathbf{a}_n)^{y_{n,i}}}{y_{n,i}!} \\
&= \sum_{n=1}^{t} p_{n,i} \left[-\mathbf{d}_i^T \mathbf{a}_n + y_{n,i} \log(\mathbf{d}_i^T \mathbf{a}_n) - \log(y_{n,i}!) \right] \\
&= -\mathbf{d}_i^T \left(\sum_{n=1}^{t} p_{n,i} \mathbf{a}_n \right) + \sum_{n=1}^{t} p_{n,i} y_{n,i} \log(\mathbf{d}_i^T \mathbf{a}_n) \\
&\quad - \sum_{n=1}^{t} p_{n,i} \log(y_{n,i}!)
\end{aligned} \tag{29}$$

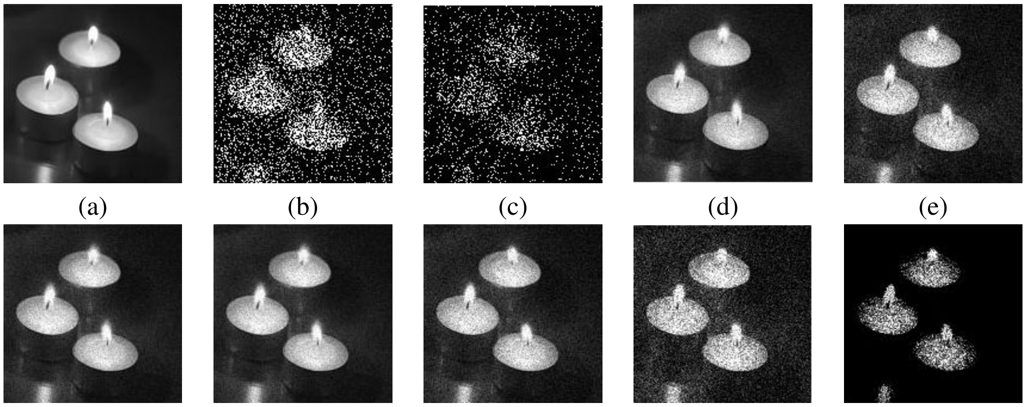
Akin to the derivation of (28), the second term in (29) can be



(a) $p = 1$

(b) $p = 0.5$

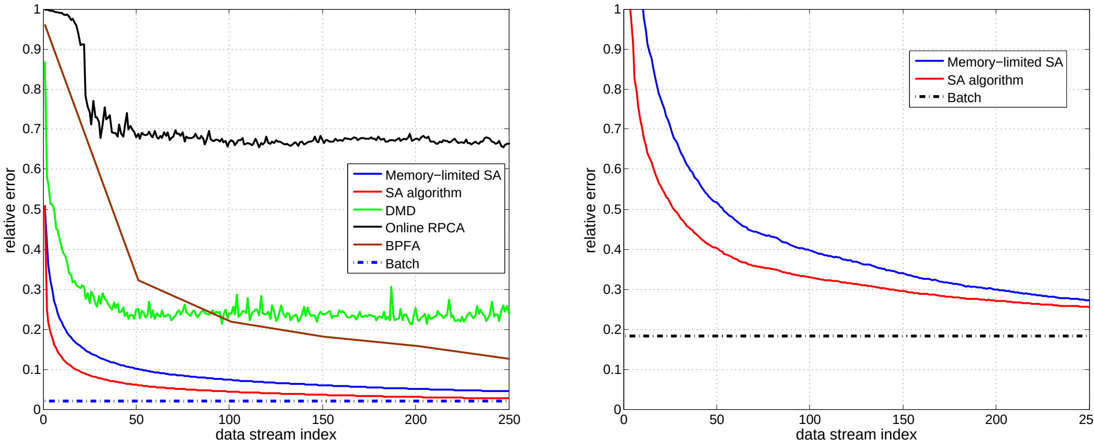
Figure 7. The mean and variance of normalized subspace reconstruction errors for the SA and Memory-limited SA algorithms with 50 Monte-Carlo simulation, where $N = 200$ and the true rank $K = 20$, for (a) fully observed data with $p = 1$ and (b) partially observed data with $p = 0.5$.



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 8. Poisson video reconstruction. (a) original video frame. (b) Poisson observation when $p = 1$. (c) Poisson observation when $p = 0.5$. (d) SA algorithm recovered video when $p = 1$. (e) Memory-limited SA algorithm recovered video frame when $p = 0.5$. (f) Batch algorithm recovered video when $p = 1$. (g) Batch algorithm recovered video when $p = 0.5$. (h) BPFA recovered video. (i) DMD recovered video. (j) Online RPCA recovered video.



(a)

(b)

Figure 9. Relative reconstruction error of various algorithms. (a) The relative reconstruction errors for SA, memory-limited SA, DMD, BPFA, Online RPCA and batch algorithms when $p = 1$. (b) The relative reconstruction errors for SA, memory-limited SA and batch algorithms when $p = 0.5$.

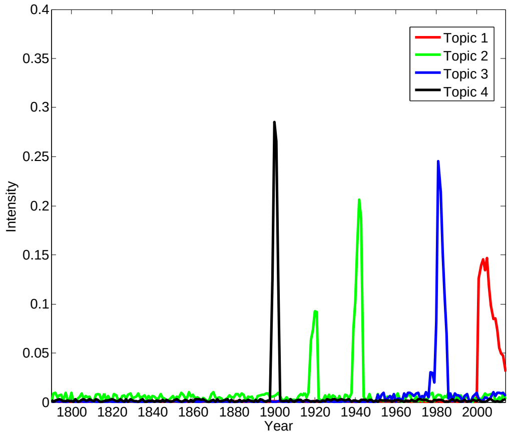


Figure 10. The intensities of associated topics inferred by SA algorithm.

bounded as follows:

$$\begin{aligned}&\sum_{n=1}^{t} p_{n,i}y_{n,i}\log(\mathbf{d}_i^T\mathbf{a}_n)\\ =&\left(\sum_{n=1}^{t} p_{n,i}y_{n,i}\right)\cdot\left[\sum_{n=1}^{t}\left(\frac{p_{n,i}y_{n,i}}{\sum_{n=1}^{t} p_{n,i}y_{n,i}}\right)\log(\mathbf{d}_i^T\mathbf{a}_n)\right]\\ \geq&\left(\sum_{n=1}^{t} p_{n,i}y_{n,i}\right)\cdot\left\{\log\left[\sum_{n=1}^{t}\frac{p_{n,i}y_{n,i}(\mathbf{d}_i^T\mathbf{a}_n)}{\sum_{n=1}^{t} p_{n,i}y_{n,i}}\right]-T\right\}.\end{aligned}\tag{30}$$

Equation (20) then follows by rearranging terms. □

Appendix B
Proofs of Theorems 1 and 2

We first introduce several theorems in the forms which are useful later.

Theorem 3 ([31])**.** *Let $\{u_t\}$ be a nonnegative discrete-time stochastic process on a probability space $(\Omega, \mathcal{A}, \mathbb{P})$. Let $\mathcal{F}_t$ be a filtration adapted to $\{u_t\}$. Define a binary process δ_t such that $\delta_t = 1$ if $\mathbb{E}[u_{t+1} - u_t|\mathcal{F}_t] > 0$ and $\delta_t = 0$ otherwise. If $\sum_{t=1}^{\infty}\mathbb{E}[\delta_t(u_{t+1} - u_t)] < \infty$, then $u_t \to u$ a.s., where u is integrable on $(\Omega, \mathcal{A}, \mathbb{P})$.*

Theorem 4 (Donsker's Theorem and Glivenko-Cantelli Theorem [32])**.** *Let $X_1, \dots, X_n$ be i.i.d. from a distribution $\mathbb{P}$. Define the empirical distribution $\mathbb{P}_n = \frac{1}{n}\sum_{i=1}^{n}\delta_{X_i}$. For a measurable function f, define $\mathbb{P}f$ and $\mathbb{P}_n f$ as the expectations of f under $\mathbb{P}$ and $\mathbb{P}_n$, respectively. Define $G_n(f) = \sqrt{n}(\mathbb{P}_n f - \mathbb{P}f)$, where $f \in \mathcal{F}$ and $\mathcal{F}$ is a collection of measurable functions. We call $\mathcal{F}$ is $\mathbb{P}$-Donsker if $\{G_n\}$ converges in distribution to a zero-mean Gaussian process G. Moreover, in that case, we have $\mathbb{E}\|G_n\|_\infty \to \mathbb{E}\|G\|_\infty$, where $\|\cdot\|_\infty$ denotes the supnorm on $\mathcal{F}$. Additionally, $\mathcal{F}$ is called $\mathbb{P}$-Glivenko-Cantelli if $\|\mathbb{P}_n f - \mathbb{P}f\|_\infty \to 0$ a.s..*

In particular, the following theorem provides a sufficient condition to verify $\mathcal{F}$ is both $\mathbb{P}$-Donsker and $\mathbb{P}$-Glivenko-Cantelli.

Theorem 5 ([32])**.** *Define a probability space $(\mathcal{X}, \mathcal{A}, \mathbb{P})$. Let $\mathcal{F} = \{f_\theta : \mathcal{X} \to \mathbb{R}|\theta \in \Theta\}$, where $\Theta \subset \mathbb{R}^d$ is bounded. If there exists a constant K such that*

$$|f_{\theta_1}(x) - f_{\theta_2}(x)| \leq \|\theta_1 - \theta_2\|, \forall\theta_1, \theta_2 \in \Theta, \forall x \in \mathcal{X}, \tag{31}$$

where $\|\cdot\|$ denotes arbitrary vector norm in $\mathbb{R}^d$, then $\mathcal{F}$ is both $\mathbb{P}$-Donsker and $\mathbb{P}$-Glivenko-Cantelli.

We now establish the following lemmas.

Lemma 2. *There exists a unique minimizer $\mathbf{a}$ of $\ell'(\mathbf{y}_n, \mathbf{D}, \mathbf{a})$, for any $(\mathbf{y}_n, \mathbf{D}) \in \mathcal{C}_1 \times \mathcal{C}_2$.*

Proof. It is enough to show that $\ell'(\mathbf{y}_n, \mathbf{D}, \mathbf{a})$ is a strictly convex function of $\mathbf{a}$, for any $(\mathbf{y}_n, \mathbf{D}) \in \mathcal{C}_1 \times \mathcal{C}_2$. Since $-\log\mathrm{Pois}(\mathbf{y}_n; \mathbf{D}\mathbf{a})$ is a convex function of $\mathbf{a}$ and $\nabla^2_{\mathbf{a}}\mu\|\mathbf{a}\|_2^2 \succeq 2\mu\mathbf{I}_{K\times K}$, where $\mathbf{I}_{K\times K}$ denotes the identity matrix of size $K \times K$. Hence, $\ell'(\mathbf{y}_n, \mathbf{D}, \mathbf{a})$ is a strictly convex function of $\mathbf{a}$, for any $(\mathbf{y}_n, \mathbf{D}) \in \mathcal{C}_1 \times \mathcal{C}_2$. □

Lemma 3. *$f'_t(\cdot)$, $f_t(\cdot)$ and $\ell(\mathbf{y}_n, \cdot)$ are Lipschitz on $\mathcal{C}_2$.*

Proof. It is straightforward to check f'_t is continuously differentiable on the compact set $\mathcal{C}_2$. Therefore, f'_t is Lipschitz on $\mathcal{C}_2$. By the definition of f'_t, it is enough to show $\ell(\mathbf{y}_n, \cdot)$ is Lipschitz. It is easy to check $\ell'(\mathbf{y}_n, \mathbf{D}, \mathbf{a}_n)$ is continuously differentiable on $\mathcal{C}_1 \times \mathcal{C}_2$. Via Danskin's theorem [32], along with Lemma 2, we conclude $\ell(\mathbf{y}_n, \cdot)$ is continuously differentiable on $\mathcal{C}_2$. Hence, $\ell(\mathbf{y}_n, \cdot)$ and $f_t(\cdot)$ are Lipschitz on $\mathcal{C}_2$. □

Lemma 4. *For all $t \geq 1$, $f'_t(\mathbf{D})$ is strongly-convex on $\mathcal{C}_2$, for any $(\mathbf{y}_n, \mathbf{a}_n) \in \mathcal{C}_1 \times \mathcal{C}_3$.*

Proof. By the definiton of $f'_t(\mathbf{D})$, it is enough to check whether $\ell'(\mathbf{y}_n, \mathbf{D}, \mathbf{a}_n)$ is strongly convex. Let's check the Hessian of $\ell'(\mathbf{y}_n, \mathbf{D}, \mathbf{a}_n)$. Since $-\log\mathrm{Pois}(\mathbf{y}_n; \mathbf{D}\mathbf{a}_n)$ is a convex function of $\mathbf{D}$ and $\nabla^2_{\mathbf{D}}\lambda\|\mathbf{D}\|_F^2 \succeq 2\lambda\mathbf{I}_{NK\times NK}$, where $\mathbf{I}_{NK\times NK}$ denotes the identity matrix of size $NK \times NK$. We have $\nabla^2_{\mathbf{D}}\ell'(\mathbf{y}_n, \mathbf{D}, \mathbf{a}_n) \succeq 2\lambda\mathbf{I}_{NK\times NK}$. Hence, $f'_t(\mathbf{D})$ is strongly-convex on $\mathcal{C}_2$. □

Lemma 5. *For Algorithm 1, we have $\|\hat{\mathbf{D}}_{t+1} - \hat{\mathbf{D}}_t\|_F \leq c/t$ for all sufficient large t, where c is a constant.*

Proof. By Lemma 4 and the property of strong convexity, we have

$$f'_t(\hat{\mathbf{D}}_{t+1}) - f'_t(\hat{\mathbf{D}}_t) \geq \frac{c'}{2}\|\hat{\mathbf{D}}_{t+1} - \hat{\mathbf{D}}_t\|_F^2, \tag{32}$$

where c' is a constant. We also have

$$\begin{aligned}&f'_t(\hat{\mathbf{D}}_{t+1}) - f'_t(\hat{\mathbf{D}}_t)\\ =&f'_t(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_{t+1}) + f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t)\\ +&f'_{t+1}(\hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t) &(33)\\ \leq&g_t(\hat{\mathbf{D}}_{t+1}) - g_t(\hat{\mathbf{D}}_t), &(34)\end{aligned}$$

where $g_t := f'_t - f'_{t+1}$ and (34) follows since $\hat{\mathbf{D}}_{t+1}$ is the minimizer of f'_{t+1}. Note that $g_t(\mathbf{D})$ can be expressed as

$$\begin{aligned}g_t(\mathbf{D}) &= \frac{1}{t}\sum_{n=1}^{t}\ell'(\mathbf{y}_n, \mathbf{D}, \hat{\mathbf{a}}_n) - \frac{1}{t+1}\sum_{n=1}^{t+1}\ell'(\mathbf{y}_n, \mathbf{D}, \hat{\mathbf{a}}_n)\\ &= \frac{(t+1)\sum_{n=1}^{t}\ell'(\mathbf{y}_n, \mathbf{D}, \hat{\mathbf{a}}_n) - t\sum_{n=1}^{t+1}\ell'(\mathbf{y}_n, \mathbf{D}, \hat{\mathbf{a}}_n)}{t(t+1)}\end{aligned}$$

$$= \frac{\left(\sum_{n=1}^{t} \ell'(\mathbf{y}_n, \mathbf{D}, \hat{\mathbf{a}}_n) - t\ell'(\mathbf{y}_{t+1}, \mathbf{D}, \hat{\mathbf{a}}_{t+1})\right)}{t(t+1)}$$

$$= \frac{1}{t(t+1)} \sum_{n=1}^{t} \left[\ell'(\mathbf{y}_n, \mathbf{D}, \hat{\mathbf{a}}_n) - \ell'(\mathbf{y}_{t+1}, \mathbf{D}, \hat{\mathbf{a}}_{t+1})\right]. \quad (35)$$

Let $h(\mathbf{y}, \mathbf{y}', \mathbf{D}, \mathbf{a}, \mathbf{a}') := \ell'(\mathbf{y}, \mathbf{D}, \mathbf{a}) - \ell'(\mathbf{y}', \mathbf{D}, \mathbf{a}')$. It is easy to check that h is continuously differentiable and $\|\nabla h\|_2$ is bounded on the compact domain $\mathcal{C}_1 \times \mathcal{C}_1 \times \mathcal{C}_2 \times \mathcal{C}_3 \times \mathcal{C}_3$. By the properties of Lipschitz function, h is Lipschitz and thus g_t is also Lipschitz. Hence, we have that there exists some constant c'' such that

$$g_t(\hat{\mathbf{D}}_{t+1}) - g_t(\hat{\mathbf{D}}_t) \leq \frac{1}{t(t+1)} t c'' \|\hat{\mathbf{D}}_{t+1} - \hat{\mathbf{D}}_t\|_F$$

$$= \frac{c''}{(t+1)} \|\hat{\mathbf{D}}_{t+1} - \hat{\mathbf{D}}_t\|_F. \quad (36)$$

Combing (36) with (32), we prove that

$$\|\hat{\mathbf{D}}_{t+1} - \hat{\mathbf{D}}_t\|_F \leq \frac{c}{t+1} \leq \frac{c}{t}. \quad (37)$$

where c is a constant. □

Proof of Theorem 1. The proof is established in the following manner. We first show the a.s. convergence of $\{f'_t(\hat{\mathbf{D}}_t)\}$ via Theorem 3 by proving the convergence of the expected positive variation series of $u_t := f'_t(\hat{\mathbf{D}}_t)$, in which Theorem 4 is employed to bound each term in the aforementioned series. Via Lemma 3 and Theorem 4, we prove that $f_t(\hat{\mathbf{D}}_t) - f(\hat{\mathbf{D}}_t) \to 0$ a.s.. We then show that $f_t(\hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t) \to 0$ a.s. by proving $\sum_{t=1}^{\infty} \frac{f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)}{t+1} < \infty$. Finally, the proof is concluded by combining these convergence results.

Define $u_t := f'_t(\hat{\mathbf{D}}_t)$ and binary δ_t such that $\delta_t = 1$ if $\mathbb{E}[u_{t+1} - u_t | \mathcal{F}_t] > 0$ and $\delta_t = 0$ otherwise. We have

$$u_{t+1} - u_t$$

$$= f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t) + f'_{t+1}(\hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t) \quad (38)$$

$$= f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t)$$

$$+ \frac{1}{t+1} \ell'(\mathbf{y}_{t+1}, \hat{\mathbf{D}}_t, \hat{\mathbf{a}}_{t+1}) + \frac{t}{t+1} f'_t(\hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t). \quad (39)$$

Note that $\ell'(\mathbf{y}_{t+1}, \hat{\mathbf{D}}_t, \hat{\mathbf{a}}_{t+1}) = \ell(\mathbf{y}_{t+1}, \hat{\mathbf{D}}_t)$, which is followed from the definition. We obtain

$$u_{t+1} - u_t$$

$$= f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t) + \frac{\ell(\mathbf{y}_{t+1}, \hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t)}{t+1} \quad (40)$$

$$= f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t)$$

$$+ \frac{\ell(\mathbf{y}_{t+1}, \hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)}{t+1} + \frac{f_t(\hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t)}{t+1}. \quad (41)$$

By the definitions of f'_t and f_t, it is straightforward to see that $f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t) \leq 0$ and $f_t(\hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t) \leq 0$. We will employ the Donsker's theorem to prove the convergence. Let us define a filtration $\{\mathcal{F}_t\}$ where $\mathcal{F}_t$ is the minimal σ-algebra such that $(\mathbf{y}_t, \hat{\mathbf{D}}_t, \hat{\mathbf{a}}_t)$ are measurable for all t. Hence, we have

$$\mathbb{E}[u_{t+1} - u_t | \mathcal{F}_t] \leq \frac{\mathbb{E}[\ell(\mathbf{y}_{t+1}, \hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t) | \mathcal{F}_t]}{t+1}$$

$$= \frac{f(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)}{t+1}. \quad (42)$$

By the definition of δ_t, we obtain

$$\mathbb{E}[\delta_t \mathbb{E}[u_{t+1} - u_t | \mathcal{F}_t]] \leq \frac{\mathbb{E}[|f(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)|]}{t+1} \quad (43)$$

$$\leq \frac{\mathbb{E}[\|f - f_t\|_\infty]}{t+1} \quad (44)$$

By Lemma 3, $\ell(\mathbf{y}, \cdot\,)$ is Lipschitz on $\mathcal{C}_2$. Hence, via Theorem 5, $\{\ell(\cdot\,, \mathbf{D})\}_{\mathbf{D} \in \mathcal{C}_2}$ is $\mathbb{P}$-Donsker and $\mathbb{E}[\|\sqrt{t}(f - f_t)\|_\infty] < Q$ for all sufficiently large t and Q is a constant. Therefore,

$$\frac{\mathbb{E}[\|f - f_t\|_\infty]}{t+1} = \frac{\mathbb{E}[\sqrt{t}\|f - f_t\|_\infty]}{\sqrt{t}(t+1)}$$

$$\leq \frac{Q}{\sqrt{t}(t+1)}. \quad (45)$$

Combining (44) and (45), we have

$$\sum_{i=1}^{\infty} \mathbb{E}[\delta_t[u_{t+1} - u_t]] = \sum_{i=1}^{\infty} \mathbb{E}[\delta_t \mathbb{E}[u_{t+1} - u_t | \mathcal{F}_t]] \quad (46)$$

$$\leq \sum_{t=1}^{\infty} \frac{M}{\sqrt{t}(t+1)} < \infty. \quad (47)$$

By Theorem 3, we have $\{f'_t(\hat{\mathbf{D}}_t)\}$ converges a.s..

Next we show that $\{f(\hat{\mathbf{D}}_t)\}$ converges a.s.. By Lemma 3 and Theorem 4, f_t is Lipschitz and hence $\mathbb{P}$-Glivenko-Cantelli. Therefore we have that $f_t(\hat{\mathbf{D}}_t) - f(\hat{\mathbf{D}}_t) \to 0$ a.s.. In order to show the convergence of $f(\hat{\mathbf{D}}_t)$, it is enough to show that $\{f_t(\hat{\mathbf{D}}_t)\}$ converges a.s.. Via (41) and the fact that $f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t) \leq 0$, we have

$$\frac{f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)}{t+1} = \mathbb{E}\left[\frac{f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)}{t+1} | \mathcal{F}_t\right] \quad (48)$$

$$= \mathbb{E}[f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t) | \mathcal{F}_t]$$

$$+ \mathbb{E}\left[\frac{\ell(\mathbf{y}_{t+1}, \hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)}{t+1} | \mathcal{F}_t\right] - \mathbb{E}[u_{t+1} - u_t | \mathcal{F}_t] \quad (49)$$

$$\leq \frac{f(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)}{t+1} - \mathbb{E}[u_{t+1} - u_t | \mathcal{F}_t] \quad (50)$$

$$\leq \frac{\|f - f_t\|_\infty}{t+1} - \mathbb{E}[u_{t+1} - u_t | \mathcal{F}_t]. \quad (51)$$

Via the previous arguments, we know that $\sum_{t=1}^{\infty} |\mathbb{E}[u_{t+1} - u_t | \mathcal{F}_t]| < \infty$. Furthermore, we have shown that $\mathbb{E}[\|\sqrt{t}(f - f_t)\|_\infty]$ is bounded. Therefore, we have $\|\sqrt{t}(f - f_t)\|_\infty$ is bounded a.s. for sufficiently large t and $\sum_{t=1}^{\infty} \frac{\|f - f_t\|_\infty}{t+1} \leq \sum_{t=1}^{\infty} \frac{\sqrt{t}\|f - f_t\|_\infty}{\sqrt{t}(t+1)} < \infty$. Hence, $\sum_{t=1}^{\infty} \frac{f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)}{t+1} < \infty$ a.s.. Via the result in [27, Lemma 8], in order to show $f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t)$ converges, it is enough to show $|(f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f_{t+1}(\hat{\mathbf{D}}_{t+1})) - (f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t))| \leq \frac{R}{t}$ for all t sufficiently large where R is a constant. We have

$$|(f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f_{t+1}(\hat{\mathbf{D}}_{t+1})) - (f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t))|$$

$$\leq |f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_t(\hat{\mathbf{D}}_t)| + |f_{t+1}(\hat{\mathbf{D}}_{t+1}) - f_t(\hat{\mathbf{D}}_t)|, \quad (52)$$

and via (40),

$$\begin{aligned}&|f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_t(\hat{\mathbf{D}}_t)|\\ &\le |f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t)| + \left|\frac{\ell(\mathbf{y}_{t+1},\hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t)}{t+1}\right|.\end{aligned}$$

By Lemmas 3 and 5, f'_t is Lipschitz and $\|\hat{\mathbf{D}}_{t+1}-\hat{\mathbf{D}}_t\|_F \le \frac{c}{t}$; moreover, ℓ and f'_t are bounded on compact domains $\mathcal{C}_1 \times \mathcal{C}_2$ and $\mathcal{C}_2$ respectively, we have

$$\begin{aligned}&|f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_t(\hat{\mathbf{D}}_t)|\\ &\le |f'_{t+1}(\hat{\mathbf{D}}_{t+1}) - f'_{t+1}(\hat{\mathbf{D}}_t)| + \left|\frac{\ell(\mathbf{y}_{t+1},\hat{\mathbf{D}}_t) - f'_t(\hat{\mathbf{D}}_t)}{t+1}\right|\\ &\le c_1\|\hat{\mathbf{D}}_{t+1} - \hat{\mathbf{D}}_t\|_F + \frac{|\ell(\mathbf{y}_{t+1},\hat{\mathbf{D}}_t)| + |f'_t(\hat{\mathbf{D}}_t)|}{t+1}\\ &\le \frac{c_1 c}{t} + \frac{c_3}{t+1} \le \frac{M'}{t},\end{aligned}$$

where M', c_1, c_2 and c_3 are constants. Similarly, we can also show

$$|f_{t+1}(\hat{\mathbf{D}}_{t+1}) - f_t(\hat{\mathbf{D}}_t)| \le \frac{M''}{t} \tag{53}$$

for some constant M''. Therefore, we have shown that $f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t) \to 0$ a.s.. Hence $\{f_t(\hat{\mathbf{D}}_t)\}$, $\{f'_t(\hat{\mathbf{D}}_t)\}$ and $\{f(\hat{\mathbf{D}}_t)\}$ converge a.s. to the same limit. □

Proof of Theorem 2. The proof is established in the following steps. We first utilize the compactness to ensure the existence of convergent subsequences and show the uniform convergence of $f'_t \to f'_\infty$. We then define a function $h_t := f'_t - f_t$. By taking limits on both side of this equation, we derive that $f = f'_\infty - h_\infty$. In order to show $\hat{\mathbf{D}}_\infty$ is a local minimum of f, we establish that $\langle \nabla f'_\infty(\hat{\mathbf{D}}_\infty), \mathbf{D} - \hat{\mathbf{D}}_\infty\rangle \ge 0$ and $\nabla h_\infty(\hat{\mathbf{D}}_\infty) = 0$ for any $\mathbf{D} \in \mathcal{C}_2$. Based on these results, we show that $\langle \nabla f(\hat{\mathbf{D}}_\infty), \mathbf{D} - \hat{\mathbf{D}}_\infty\rangle \ge 0$ for any $\mathbf{D} \in \mathcal{C}_2$, which concludes the proof.

By Theorem 1, we can almost surely find a realization $\{\mathbf{y}_t\}$ such that $f'_t(\hat{\mathbf{D}}_t) \to f_t(\hat{\mathbf{D}}_t)$. Note that now $\{\hat{\mathbf{D}}_t\}$ is a deterministic sequence and it is enough to show that any convergent subsequence of $\{\hat{\mathbf{D}}_t\}$ converges to a local minimum of f. The existence of convergent subsequence of $\{\hat{\mathbf{D}}_t\}$ is guaranteed by the compactness assumption on $\mathbf{D}$. In order to ease the notation, we assume that $\{\hat{\mathbf{D}}_t\}$ converges without loss of generality and denote the limit as $\hat{\mathbf{D}}_\infty$.

We first show that $\{f'_t\}$ converges uniformly to a differentiable function f'_∞. Since $f'_t(\mathbf{D}) := \frac{1}{t}\sum_{n=1}^t \ell'(\mathbf{y}_n, \mathbf{D}, \hat{\mathbf{a}}_n)$, and we have

$$\begin{aligned}&\frac{1}{t}|\ell'(\mathbf{y}_n,\mathbf{D},\hat{\mathbf{a}}_n)| = \frac{1}{t}\left|\log \mathrm{Pois}(\mathbf{y}_n;\mathbf{D}\hat{\mathbf{a}}_n) + \lambda\|\mathbf{D}\|_F^2 + \mu\|\hat{\mathbf{a}}_n\|_2\right|\\ &\le \frac{1}{t}\sum_{i=1}^N \left[|\mathbf{d}_i^T\hat{\mathbf{a}}_n| + |y_{n,i}\log(\mathbf{d}_i^T\hat{\mathbf{a}}_n)| + |\log y_{n,i}!|\right] + \frac{\lambda}{t}\|\mathbf{D}\|_F^2\\ &\qquad + \frac{\mu}{t}\|\hat{\mathbf{a}}_n\|_2 \qquad (54)\\ &\le \frac{NU + \lambda U' + \mu U''}{t}, \qquad (55)\end{aligned}$$

where U, U' and U'' are positive constants and the last inequality follows from the fact that all variables are supported on compact domains. Since $\lim_{t\to\infty}\sum_{n=1}^t \frac{NU+\lambda U'}{t} = NU + \lambda U' + \mu U'' < \infty$, by the Weierstrass M-test [33], $f'_t(\mathbf{D})$ converges uniformly.

Define $h_t = f'_t - f_t$ and note that $h_t \to h_\infty := f'_\infty - f$ and $\nabla f = \nabla f'_\infty - \nabla h_\infty$. We aim to show that $\hat{\mathbf{D}}_\infty$ is a local minimum of f, and it is equivalent to show that $\langle \nabla f_\infty(\hat{\mathbf{D}}_\infty), \mathbf{D} - \hat{\mathbf{D}}_\infty\rangle \ge 0$ for any $\mathbf{D} \in \mathcal{C}_2$, *i.e.,* directional derivative is non-negative. It is enough to show that $\langle \nabla f'_\infty(\hat{\mathbf{D}}_\infty), \mathbf{D} - \hat{\mathbf{D}}_\infty\rangle \ge 0$ and $\nabla h_\infty(\hat{\mathbf{D}}_\infty) = 0$ for any $\mathbf{D} \in \mathcal{C}_2$. We have

$$\begin{aligned}&|f'_t(\hat{\mathbf{D}}_t) - f'_\infty(\hat{\mathbf{D}}_\infty)|\\ &= |f'_t(\hat{\mathbf{D}}_t) - f'_\infty(\hat{\mathbf{D}}_t) + f'_\infty(\hat{\mathbf{D}}_t) - f'_\infty(\hat{\mathbf{D}}_\infty)|\\ &\le |f'_t(\hat{\mathbf{D}}_t) - f'_\infty(\hat{\mathbf{D}}_t)| + |f'_\infty(\hat{\mathbf{D}}_t) - f'_\infty(\hat{\mathbf{D}}_\infty)|. \qquad (56)\end{aligned}$$

Since $f'_t \to f'_\infty$ uniformly and f'_∞ is continuous, we have $|f'_t(\hat{\mathbf{D}}_t) - f'_\infty(\hat{\mathbf{D}}_t)| \to 0$ and $|f'_\infty(\hat{\mathbf{D}}_t) - f'_\infty(\hat{\mathbf{D}}_\infty)| \to 0$. Hence we show $f'_t(\hat{\mathbf{D}}_t) \to f'_\infty(\hat{\mathbf{D}}_\infty)$. By definition, $f'_t(\hat{\mathbf{D}}_t) \le f'_\infty(\mathbf{D})$ for any $\mathbf{D} \in \mathcal{C}_2$ and take the limit on both side, we show that

$$f'_\infty(\hat{\mathbf{D}}_\infty) \le f'_\infty(\mathbf{D}) \tag{57}$$

and this implies $\langle \nabla f'_\infty(\hat{\mathbf{D}}_\infty), \mathbf{D} - \hat{\mathbf{D}}_\infty\rangle \ge 0$. Similarly, we have

$$\begin{aligned}&|f_t(\hat{\mathbf{D}}_t) - f(\hat{\mathbf{D}}_\infty)|\\ &= |f_t(\hat{\mathbf{D}}_t) - f(\hat{\mathbf{D}}_t) + f(\hat{\mathbf{D}}_t) - f(\hat{\mathbf{D}}_\infty)| \qquad (58)\\ &\le |f_t(\hat{\mathbf{D}}_t) - f(\hat{\mathbf{D}}_t)| + |f(\hat{\mathbf{D}}_t) - f(\hat{\mathbf{D}}_\infty)|. \qquad (59)\end{aligned}$$

Since $f_t \to f$ uniformly and f is continuous, we derive $f_t(\hat{\mathbf{D}}_t) \to f(\hat{\mathbf{D}}_\infty)$.

By Lemma 3, f'_t and f_t are Lipschitz functions. Denote the Lipschitz constant as L and h_t is a $2L$-Lipschitz function, we have

$$\frac{1}{2L}\|\nabla h_t(\hat{\mathbf{D}}_t)\|_2 \le 1. \tag{60}$$

By definition, $h_t(\mathbf{D}) \ge 0$ for any $\mathbf{D} \in \mathcal{C}_2$. Multiplying both sides of (60) by $h_t(\hat{\mathbf{D}}_t)$, we have

$$\begin{aligned}\frac{h_t(\hat{\mathbf{D}}_t)}{2L}\|\nabla h_t(\hat{\mathbf{D}}_t)\|_2 &\le h_t(\hat{\mathbf{D}}_t)\\ &= f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t). \qquad (61)\end{aligned}$$

Since $f'_t(\hat{\mathbf{D}}_t) - f_t(\hat{\mathbf{D}}_t) \to 0$, we conclude that at least either $h_t(\hat{\mathbf{D}}_t) \to 0$ or $\nabla h_t(\hat{\mathbf{D}}_t) \to 0$. If $h_t(\hat{\mathbf{D}}_t) \to 0$, then $h_\infty(\hat{\mathbf{D}}_\infty) \to 0$ and $\hat{\mathbf{D}}_\infty$ is a minimum of h_∞. Hence $\nabla h_\infty(\hat{\mathbf{D}}_\infty) = 0$ and $\langle \nabla h_\infty(\hat{\mathbf{D}}_\infty), \mathbf{D} - \hat{\mathbf{D}}_\infty\rangle \ge 0$ for any $\mathbf{D} \in \mathcal{C}_2$. The proof is concluded.

Otherwise, if $\nabla h_t(\hat{\mathbf{D}}_t) \to 0$, consider the Taylor expansion of h_t at $\hat{\mathbf{D}}_t$

$$h_t(\mathbf{D}) = h_t(\hat{\mathbf{D}}_t) + \langle \nabla h_t(\hat{\mathbf{D}}_t), \mathbf{D} - \hat{\mathbf{D}}_t\rangle + o(\|\mathbf{D} - \hat{\mathbf{D}}_t\|_2).$$

Take limits on both sides of above equation and together with $\nabla h_t(\hat{\mathbf{D}}_t) \to 0$, we end up with

$$h_\infty(\mathbf{D}) = h_\infty(\hat{\mathbf{D}}_\infty) + o(\|\mathbf{D} - \hat{\mathbf{D}}_\infty\|_2).$$

If we compare this expansion to the Taylor expansion of

$h_\infty(\mathbf{D})$, we conclude

$$\nabla h_\infty(\hat{\mathbf{D}}_\infty) = 0.$$

Thus for both cases, we prove that $\langle \nabla h_\infty(\hat{\mathbf{D}}_\infty), \mathbf{D} - \hat{\mathbf{D}}_\infty \rangle = 0$ for any $\mathbf{D} \in \mathcal{C}_2$ and the proof is concluded. □

References

[1] T. Zheng, M. J. Salganik, and A. Gelman, "How many people do you know in prison? using overdispersion in count data to estimate social structure in networks," *Journal of the American Statistical Association*, vol. 101, no. 474, pp. 409–423, 2006.

[2] M. Seeger, H. Nickisch, R. Pohmann, and B. Schölkopf, "Optimization of k-space trajectories for compressed sensing by bayesian experimental design," *Magnetic resonance in medicine*, vol. 63, no. 1, pp. 116–126, 2010.

[3] L. Wang, J. Huang, X. Yuan, K. Krishnamurthy, J. Greenberg, V. Cevher, M. R. Rodrigues, D. Brady, R. Calderbank, and L. Carin, "Signal recovery and system calibration from multiple compressive poisson measurements," *SIAM Journal on Imaging Sciences*, vol. 8, no. 3, pp. 1923–1954, 2015.

[4] D. J. Brady, *Optical imaging and spectroscopy*. John Wiley & Sons, 2009.

[5] E. C. Hall and R. Willett, "Dynamical models and tracking regret in online convex programming." in *ICML*, 2013, pp. 579–587.

[6] E. Oja, "Simplified neuron model as a principal component analyzer," *Journal of mathematical biology*, vol. 15, no. 3, pp. 267–273, 1982.

[7] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 689–696.

[8] S. Tsuge, M. Shishibori, S. Kuroiwa, and K. Kita, "Dimensionality reduction using non-negative matrix factorization for information retrieval," in *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, vol. 2. IEEE, 2001, pp. 960–965.

[9] E. J. Candes and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, no. 6, pp. 717–772, April 2009.

[10] L. Balzano, R. Nowak, and B. Recht, "Online identification and tracking of subspaces from highly incomplete information," in *Communication, Control, and Computing (Allerton), 2010 48th Annual Allerton Conference on*. IEEE, 2010, pp. 704–711.

[11] Y. Chi, Y. C. Eldar, and R. Calderbank, "Petrels: Parallel estimation and tracking of subspace by recursive least squares from partial observations," *IEEE Trans. on Signal Processing*, 2013.

[12] J. Feng, H. Xu, and S. Yan, "Online robust pca via stochastic optimization," in *Advances in Neural Information Processing Systems*, 2013, pp. 404–412.

[13] H. Guo, C. Qiu, and N. Vaswani, "An online algorithm for separating sparse and low-dimensional signal sequences from their sum," *IEEE Transactions on Signal Processing*, vol. 62, no. 16, pp. 4284–4297, 2014.

[14] M. Mardani, G. Mateos, and G. B. Giannakis, "Rank minimization for subspace tracking from incomplete data," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 5681–5685.

[15] Z. Kang and C. J. Spanos, "Sequential logistic principal component analysis (slpca): Dimensional reduction in streaming multivariate binary-state system," in *Machine Learning and Applications (ICMLA), 2014 13th International Conference on*. IEEE, 2014, pp. 171–177.

[16] Y. Shen, M. Mardani, and G. B. Giannakis, "Online categorical subspace learning for sketching big data with misses," *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 4004–4018, 2016.

[17] Z. T. Harmany, R. F. Marcia, and R. M. Willett, "This is SPIRAL-TAP: Sparse poisson intensity reconstruction algorithms-theory and practice," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 1084–1096, 2012.

[18] M. Raginsky, S. Jafarpour, Z. T. Harmany, R. F. Marcia, R. Willett, and R. Calderbank, "Performance bounds for expander-based compressed sensing in Poisson noise," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4139–4153, 2011.

[19] M. Raginsky, R. Willett, Z. Harmany, and R. Marcia, "Compressed sensing performance bounds under Poisson noise," *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 3990–4002, 2010.

[20] L. Wang, D. E. Carlson, M. Rodrigues, D. Wilcox, R. Calderbank, and L. Carin, "Designed measurements for vector count data," in *Advances in neural information processing systems*, 2013, pp. 1142–1150.

[21] Y. Xie, Y. Chi, and R. Calderbank, "Low-rank matrix recovery with poisson noise," in *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*. IEEE, 2013, pp. 622–622.

[22] Y. Cao and Y. Xie, "Poisson matrix completion," in *Information Theory (ISIT), 2015 IEEE International Symposium on*. IEEE, 2015, pp. 1841–1845.

[23] L. Bottou, "Online learning and stochastic approximations," *On-line learning in neural networks*, vol. 17, no. 9, p. 142, 1998.

[24] A. Acharya, J. Ghosh, and M. Zhou, "Nonparametric bayesian factor analysis for dynamic count matrices," in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, 2015, pp. 1–9.

[25] D. L. Pimentel-Alarcón, N. Boston, and R. D. Nowak, "A characterization of deterministic sampling patterns for low-rank matrix completion," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 4, pp. 623–636, 2016.

[26] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA journal of numerical analysis*, vol. 8, no. 1, pp. 141–148, 1988.

[27] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research*, vol. 11, no. Jan, pp. 19–60, 2010.

[28] J. Mairal, "Stochastic majorization-minimization algorithms for large-scale optimization," in *Advances in Neural Information Processing Systems*, 2013, pp. 2283–2291.

[29] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, no. Jan, pp. 993–1022, 2003.

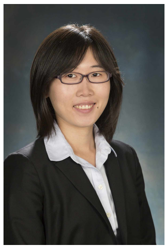
[30] S. Simic, "On an upper bound for Jensen's inequality," *Journal of Inequalities in Pure and Applied Mathematics*, 2009.

[31] M. Métivier, *Semimartingales: a course on stochastic processes*. Walter de Gruyter, 1982, vol. 2.

[32] A. W. Van der Vaart, *Asymptotic Statistics*. Cambridge university press, 2000, vol. 3.

[33] W. Rudin, *Principles of mathematical analysis*. McGraw-Hill New York, 1964, vol. 3.

Liming Wang (S'08-M'11) received the B.E. degree in Electronics and Information Engineering from the Huazhong University of Science and Technology, China, in 2006, the M.S. degree in Mathematics and the Ph.D. degree in Electrical and Computer Engineering from the University of Illinois at Chicago in 2011. From 2011 to 2017, he held postdoctoral positions at Columbia University, Duke University and The Ohio State University. He was also a Visiting Scholar at University College London, UK. Since October 2017, he is a Senior Research Engineer at HERE Technologies. His research interests are in high-dimensional signal processing, machine learning, information theory, genomic signal processing and bioinformatics.



Yuejie Chi (S'09-M'12-SM'17) received the Ph.D. degree in Electrical Engineering from Princeton University in 2012, and the B.E. (Hon.) degree in Electrical Engineering from Tsinghua University, Beijing, China, in 2007. Since September 2012, she has been with the department of Electrical and Computer Engineering and the department of Biomedical Informatics at The Ohio State University, where she is now an Associate Professor.

She is the recipient of the IEEE Signal Processing Society Young Author Best Paper Award in 2013 and the Best Paper Award at the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) in 2012. She received the Young Investigator Program Awards from AFOSR and ONR respectively in 2015, the Ralph E. Powe Junior Faculty Enhancement Award from Oak Ridge Associated Universities in 2014, a Google Faculty Research Award in 2013, the Roberto Padovani scholarship from Qualcomm Inc. in 2010, and an Engineering Fellowship from Princeton University in 2007. She is an Elected Member of the MLSP and SPTM Technical Committees of the IEEE Signal Processing Society since January 2016. She has held visiting positions at Colorado State University, Stanford University and Duke University, and interned at Qualcomm Inc. and Mitsubishi Electric Research Lab. Her research interests include statistical signal processing, information theory, machine learning and their applications in high-dimensional data analysis, network inference, active sensing and bioinformatics.