Sequential Logistic Principal Component Analysis (SLPCA): Dimensional Reduction in Streaming Multivariate Binary-State System

Zhaoyi Kang, Costas J. Spanos
Dept. of Electrical Engineering & Computer Sciences, UC Berkeley, Berkeley, CA 94709
{kangzy, spanos}@berkeley.edu,

***Abstract*—Sequential or online dimensional reduction[1] is of interests due to the explosion of streaming data based applications and the requirement of adaptive statistical modeling, in many emerging fields, such as the modeling of energy end-use profile. Principal Component Analysis (PCA), is the classical way of dimensional reduction. However, traditional PCA coincides with maximum likelihood interpretation only when data follows Gaussian distribution. The Bregman Divergence was introduced to extend PCA with maximum likelihood in exponential family distribution. In this work, we study this generalized form PCA for Bernoulli variables, which is called Logistic PCA (LPCA). We extend the batch-mode LPCA [1] to a sequential version (SLPCA). The convergence property of this algorithm is discussed compared to the batch version (BLPCA), as well as its performance in reducing the dimension for multivariate binary-state systems. Its application in building energy end-use profile modeling is also investigated.**

I. Introduction

Sequential data mining has received considerable attention recently as the development in wireless-sensor information technology facilitates the collection of huge amount of streaming data – This brings about several challenges on the efficiency in computation, storage and the performance of statistical learning algorithms [2]. Dimensional reduction in the streaming environment is one of the techniques that can help to overcome those issues [3].

Among the dimensional reduction techniques, Principal Component Analysis (PCA) is most widely-known. PCA finds the linear projection of the original data matrix which explains the largest portion of the variance. From probability perspective, PCA coincides with maximum likelihood reconstruction only when the data are consistently[2] Gaussian distributed. Therefore, it is natural to consider alternatives of traditional PCA when data largely deviates from Gaussian distribution [4]. Recently, Bregman Divergence is introduced to achieve a generalized PCA framework for a family of exponential distributed data (i.e. *e*PCA) [4]. As a generalization over the Frobenious norm, KL-divergence, Mahalanobis distance etc., Bregman Divergence is believed to better quantify the distance of variables coming from non-Gaussian distributions [5] [6]. In the case of Bernoulli random variables, which we are interested in, the generalized PCA can be viewed as Logistic PCA (LPCA).

In this work, we extend the LPCA to the sequential version, based on the sequential convex optimization theory [7] [8]. The convergence property of this algorithm is discussed with respect to the batch optimization algorithm. An application in building energy end-use profile modeling is investigated as an experiment of this method, which demonstrates its capability in reducing dimension in multivariate binary-state systems.

This paper is organized as follows: In Section II, the background and the detail of the algorithm is given, including PCA, exponential family, the Bregman Divergence and eventually the sequential LPCA (i.e. SLPCA) which we propose. In Section III, the convergence property of the algorithm is discussed, followed by the simulation results as well as the application in energy end-use modeling in Section IV. In Section V, conclusion is drawn.

II. Algorithm Framework

A. Principal Component Analysis

PCA is a well-known technique for dimensional reduction for high dimension data. It is of importance in high dimensional model, and in a variety of applications, ranging from face recognition to generalized machine learning [2].

There are two perspectives of PCA [9]. The first is the matrix factorization perspective. For a matrix $\mathbf{X} \in \mathbb{R}^{N\times P}$, we find a lower rank matrix $\mathbf{\Theta}$ to minimize the error:

$$\min_{\mathbf{\Theta}} \|\mathbf{X} - \mathbf{\Theta}\|_F^2 \tag{1}$$

in which $\|\cdot\|_F$ is the Frobenious norm. This problem can be solved by Singular Vector Decomposition (SVD).

However, there is another perspective of PCA that is less widely-known, which is called the probabilistic interpretation. Here, the columns of $\mathbf{X} \in \mathbb{R}^{N\times P}$ can be viewed as N samples drawn from a Gaussian distribution with dimension lower than P. This idea can be used in larger family of distributions, for example, the exponential family distributions.

B. Exponential Family

Definition 1 (Exponential Family)**.** *In the exponential family of distributions the conditional probability of a value X given*

[1]Dimesionality reduction, dimension reduction, dimensional reduction refer to the same thing, in this work.

[2]By consistency the streaming data are following same distribution.

parameter value Θ takes the following form:

$$\log P(X|\Theta) = \log P_0(X) + X\Theta - G(\Theta) \tag{2}$$

In which, Θ is called the natural parameter of the distribution. Then we have $\mathrm{E}[X] = \nabla G(\Theta) = g(\Theta)$ is the inverse canonical link function, and $\mathrm{Var}[X] = \nabla\nabla^T G(\Theta)$.

C. Exponential Family PCA

The *squared distance* in Equation (1) is inappropriate when the data is not Gaussian, which happens a lot in real world. The Bregman Divergence is introduced to generalize the distance between X and Θ in Equation (1) [6] [5] [4].

Definition 2 (Bregman Divergence)**.** *The* Bregman divergence *between two variable $p, q \in \mathbb{R}^d$ w.r.t. F is defined as:*

$$B_F(p\|q) = F(p) - F(q) - \nabla F(q)^T(p - q) \tag{3}$$

For an exponential family distribution in (2)*, we are interested to define the distance between the data X and its expectation $\mathrm{E}[X] = g(\Theta)$, i.e. $B_F(X\|g(\Theta))$.*

We choose $F(g(\theta)) + G(\theta) = g(\theta)\theta$, in which $g(x) = \nabla G(x)$ [4]. Then the canonical Bregman divergence[3] is:

$$B(X\|g(\Theta)) = F(X) + \log P_0(X) - \log P(X|\Theta) \tag{4}$$

In this case, in terms of Θ, Bregman divergence is negative log likelihood function. Minimize Bregman divergence is equivalent to maximizing log likelihood function, which is statistically well-defined. Note that if $\mathbf{X}, \mathbf{\Theta} \in \mathbb{R}^{N\times P}$ form, the Bregman divergence is decomposed into item-wise as:

$$B(\mathbf{X}\|\mathbf{\Theta}) = \sum_{i,j} B(\mathbf{X}_{ij}\|\mathbf{\Theta}_{ij}) \tag{5}$$

Example 1. *In the case of Gaussian distribution, the Bregman Divergence equals to squared loss $B(x\|g(\theta)) = \frac{1}{2}(x-\theta)^2$.*

Example 2. *In the case of Bernoulli distribution, Bregman Divergence is the logit function $B(x\|g(\theta)) = \log(1 + \exp(-x^*\theta))$, where $x^* = 2x - 1 \in \{-1, 1\}$. In this case, Bregman Divergence is a convex function of θ.*

Therefore, similar to Equation (1), we can construct an optimization problem based on the Bregman Divergence. For data matrix $\mathbf{X} = \{x_{ij}\rangle$ and $\mathbf{\Theta} = \{\theta_{ij}\rangle$. Then we have:

$$\min_{\mathbf{\Theta}} B(\mathbf{X}\|g(\mathbf{\Theta})) \tag{6}$$

D. Batch Logistic PCA (BLPCA)

In this work, we will only work on Bernoulli variable, as in Example 2, we replace Bregman divergence in (6) by the logit function. Moreover, to achieve a rank-r $\mathbf{\Theta}$, we write $\mathbf{\Theta} = \mathbf{A}\mathbf{V}^T$ where $\mathbf{A} \in \mathbb{R}^{N\times r}$ and $\mathbf{V} \in \mathbb{R}^{P\times r}$, both rank-$r$. Thus the Bregmand divergence becomes:

$$B(\mathbf{X}\|g(\mathbf{A}\mathbf{V}^T)) = \sum_{i,j} \log(1 + e^{-x^*_{ij}(\mathbf{A}\mathbf{V}^T)_{ij}}) \tag{7}$$

[3]We simply use $B(X\|g(\Theta))$ to refer to this canonical form Bregman divergence in the rest of this paper

The optimization problem in (7) in not jointly convex because of the $\mathbf{A}\mathbf{V}^T$ term, but is marginally convex for $\mathbf{A}$ and $\mathbf{V}$ when the other matrix is fixed. Empirically, we can solve it in an alternating minimization algorithm to each locally best solution [4] [10] by solving $\mathbf{A}$ and $\mathbf{V}$ iteratively. However, even marginally (7) is not strongly convex, so that we can put a regularizer to avoid an infinity. Hence, we solve (7) for each $t = 1, \cdots, N$, and we call this the Batch Logistic PCA (BLPCA) algorithm:

$$\begin{cases} \mathbf{A}^t &= arg\min_{\mathbf{A}\in\mathbb{R}^{N\times r}} B(\mathbf{X}\|g(\mathbf{A}(\mathbf{V}^{t-1})^T)) + \frac{\gamma}{2}\|\mathbf{A}\|_F^2 \\ \mathbf{V}^t &= arg\min_{\mathbf{V}\in\mathbb{R}^{P\times r}} B(\mathbf{X}\|g(\mathbf{A}^t\mathbf{V}^T)) + \frac{\lambda}{2}\|\mathbf{V}\|_F^2 \end{cases} \tag{8}$$

The resulted solution of BLPCA locates in local minimum of (6). However, interestingly, from [11] [12] [13], all the local minimum are global minimum, which is partially because of the interchangeability between $\mathbf{A}$ and $\mathbf{V}$. Without loss of generality, we mark all the local minimum obtained from BLPCA as $\mathbf{A}^*$ and $\mathbf{V}^*$.

E. Sequential Logistic PCA (SLPCA)

For a sequential LPCA, we want to solve (6) with streaming data, which means $\mathbf{A} \in \mathbb{R}^{N\times r}$ changes in size as N increases, though luckily the dimension of $\mathbf{A}$ is still fixed. For simplicity, in sequential case we only consider rank-1 approximation, which means $\mathbf{A} \in \mathbb{R}^N$ and $\mathbf{V} \in \mathbb{R}^P$. Let $\mathbf{a}_t$ as the t^{th} element of $\mathbf{A}$, $t = 1, \cdots, N$. As discussed in Equation (5), we can decompose the full Bregman divergence w.r.t. each $\mathbf{a}_t$, and the x-th data $\mathbf{x}_t$:

$$B(\mathbf{x}_t\|g(\mathbf{A}\mathbf{V}^T)) = \sum_t B(\mathbf{x}_t\|g(\mathbf{a}_t\mathbf{V}^T)) = \sum_t h_t(\mathbf{a}_t, \mathbf{V})$$

in which we note:

$$h_t(\mathbf{a}_t, \mathbf{V}) = B(\mathbf{x}_t\|g(\mathbf{a}_t\mathbf{V}^T)) = \sum_j \log(1 + e^{-x^*_{tj}(\mathbf{A}\mathbf{V}^T)_{tj}}) \tag{9}$$

Then, similar to [14], at each time t, instead of working on the full $\mathbf{A}$ up to step t, we only solve for the current element $\mathbf{a}_t$ (i.e. solution is $\widetilde{\mathbf{a}}_t$); whereas update $\mathbf{V}$ with the $\widetilde{\mathbf{a}}_t$'s up to t (i.e. solution is $\widetilde{\mathbf{V}}^t$). We call this Sequential LPCA (SLPCA) algorithm. For $t = 1, \cdots, N$:

$$\begin{cases} \widetilde{\mathbf{a}}_t &= arg\min_{\mathbf{a}\in\mathbb{R}} h_t(\mathbf{a}, \widetilde{\mathbf{V}}^{t-1}) + \frac{\gamma}{2}\|\mathbf{a}\|_F^2 \\ \widetilde{\mathbf{V}}^t &= arg\min_{\mathbf{V}\in\mathbb{R}^P} \sum_{s=1}^t h_s(\widetilde{\mathbf{a}}_s, \mathbf{V}) + \frac{\lambda}{2}\|\mathbf{V}\|_F^2 \end{cases} \tag{10}$$

The one for $\widetilde{\mathbf{a}}_t$ in (10) is easy to solve with a Newton method. The one for $\widetilde{\mathbf{V}}^t$ in (10) deal with a target function increasing in size. [14] suggested to solve it sequentially based on the past value $\widetilde{\mathbf{V}}^{t-1}$ based on a surrogate function. Another benefit of this iterative method is that it is essentially a stochastic gradient descent method [15], so we don't need a strong regularization term. To see how this works, we define the surrogate function $\widetilde{h}_t(\mathbf{a}_t, \mathbf{V})$ to approximate $h_t(\mathbf{a}_t, \mathbf{V})$:

$$\begin{aligned} \widetilde{h}_t(\widetilde{\mathbf{a}}_t, \mathbf{V}) = h_t(\widetilde{\mathbf{a}}_t, \widetilde{\mathbf{V}}^{t-1}) + \nabla_{\mathbf{V}} h_t(\widetilde{\mathbf{a}}_t, \widetilde{\mathbf{V}}^{t-1})^T(\mathbf{V} - \widetilde{\mathbf{V}}^{t-1}) \\ + \frac{\alpha_t}{2}\|\mathbf{V} - \widetilde{\mathbf{V}}^{t-1}\|_F^2, \quad \alpha_t \geq \|\nabla^2_{\mathbf{V}} h_t\|_{opt} \end{aligned} \tag{11}$$

where $\|\cdot\|_{opt}$ is the operator norm.

From the above it follows that $\widetilde{h}_t(\widetilde{\mathbf{a}}_t, \mathbf{V}) \geq h_t(\widetilde{\mathbf{a}}_t, \widetilde{\mathbf{V}}^{t-1})$, and moreover, as we solve Equation (10) under $\widetilde{h}_t$ instead of h_t, we get:

$$\widetilde{\mathbf{V}}^t = \widetilde{\mathbf{V}}^{t-1} - \eta_t \nabla_{\mathbf{V}} h_t(\widetilde{\mathbf{a}}_t, \widetilde{\mathbf{V}}^{t-1}) \tag{12}$$

where $\eta_t \propto (\sum_{\tau=1}^{t} \alpha_\tau)^{-1}$ is the step size. The choice of step size η_t deserves some discussions. We will investigate in Section III on the convergence of this algorithm. The full SLPCA algorithms is shown below.

begin
  Input: data $\mathbf{X} \in \mathbb{R}^{N\times P}$, $\mathbf{X}^* = 2\mathbf{X} - 1 \in \{-1, 1\}$;
  Initialize: $\widetilde{\mathbf{V}}^t \approx 0, C, \gamma, \epsilon, \beta \in (0,1), \alpha$;
  for $t = 1, \ldots, N$, $l_t(\widetilde{\mathbf{a}}_t) \doteq h_t(\widetilde{\mathbf{a}}_t, \widetilde{\mathbf{V}}^{t-1}) + \lambda \frac{\|\widetilde{\mathbf{a}}_t\|_F^2}{2}$ **do**
    Set $\widetilde{\mathbf{a}}_t = 0$, $\Delta = \nabla l_t(\widetilde{\mathbf{a}}_t) \left(\nabla^2 l_t(\widetilde{\mathbf{a}}_t)\right)^{-1} \nabla l_t(\widetilde{\mathbf{a}}_t)$;
    while $\lambda > \epsilon$ **do**
      Let $\Delta = -\left(\nabla^2 l_t(\widetilde{\mathbf{a}}_t)\right)^{-1} \nabla l_t(\widetilde{\mathbf{a}}_t)$, $d = d_0$;
      while $l_t(\widetilde{\mathbf{a}}_t + d\Delta) > l_t(\widetilde{\mathbf{a}}_t) + \alpha d \nabla l_t^T \Delta$ **do**
        Update $d = \beta d$;
      end
      Update $\widetilde{\mathbf{a}}_t = \widetilde{\mathbf{a}}_t + d\Delta$;
      Update $\Delta = \nabla l_t(\widetilde{\mathbf{a}}_t) \left(\nabla^2 l_t(\widetilde{\mathbf{a}}_t)\right)^{-1} \nabla l_t(\widetilde{\mathbf{a}}_t)$;
    end
    Set η_t;
    Update $\widetilde{\mathbf{V}}^t = \widetilde{\mathbf{V}}^{t-1} - \eta_t \nabla_{\mathbf{V}} h_t(\widetilde{\mathbf{a}}_t, \widetilde{\mathbf{V}}^{t-1})$
  end
end

Algorithm 1: Sequential LPCA (SLPCA) Pseudo-Code

III. Convergence Analysis

In this section, we will study the convergence of SLPCA with respect to BLPCA algorithm in terms of some widely-used settings from online statistical learning society.

A. Evaluation Settings

- *Batch Bregman Divergence* (BBD), use $\{\mathbf{A}^*\}$ $\{\mathbf{V}^*\}$:

$$C_N(\mathbf{V}^*) = \frac{1}{N} \sum_{t=1}^{N} h_t(\mathbf{a}_t^*, \mathbf{V}^*) \tag{13}$$

- *Online Bregman Divergence* (OBD), use $\{\widetilde{\mathbf{a}}_t\}$ $\{\widetilde{\mathbf{V}}^N\}$:

$$\widehat{C}_N(\widetilde{\mathbf{V}}^N) = \frac{1}{N} \sum_{t=1}^{N} h_t(\widetilde{\mathbf{a}}_t, \widetilde{\mathbf{V}}^N) \tag{14}$$

- *Regret Bregman Divergence* (RBD), use $\{\widetilde{\mathbf{a}}_t\}$ $\{\widetilde{\mathbf{V}}^t\}$:

$$\widehat{Re}_N = \frac{1}{N} \sum_{t=1}^{N} h_t(\widetilde{\mathbf{a}}_t, \widetilde{\mathbf{V}}^t) \tag{15}$$

It is important to note that, the three settings coincide with the BLPCA and SLPCA problem in Equation (8) (10), except the regularization term. However, because of the term $\frac{1}{N}$, the regularization term will be diminishing as N increases. Therefore, the three settings can be used as the evaluation of the LPCA algorithm.

Moreover, RBD is of more interests since it can *sequentially* accumulate the Bregman divergence without waiting til we calculate the last update $\mathbf{V}^N$.

B. Convergence Analysis

Lemma 1. *For $t = 1, \cdots, N$ and $h_t(\cdot)$ defined in* (9)*, $\|\nabla_{\mathbf{V}} h_t\|_F \leq \|\mathbf{a}\|_F$, and $\|\nabla_{\mathbf{V}}^2 h_t\|_{opt} \leq \frac{1}{4}\|\mathbf{a}\|_F^2$.*

Proof. W.l.o.g., let $\text{rank}(\mathbf{\Theta}) = 1$, we have:

$$[\nabla_{\mathbf{V}} h_t]_j = -\frac{x_{tj}^* \mathbf{a}_t}{1 + \exp(x_{tj}^* \mathbf{a}_t \mathbf{v}_j^T)}$$

$$\left[\nabla_{\mathbf{V}}^2 h_t\right]_{ij} = \left(\frac{x_{tj}^* \mathbf{a}_t \delta_{ij}}{2\cosh(\frac{1}{2} x_{tj}^* \mathbf{a}_t \mathbf{v}_j^T)}\right)^2$$

where $\delta_{ij} = 1$ only when $i = j$ means matrix $\nabla_{\mathbf{V}}^2 h_t$ is diagonal. Since $\cosh(x) \geq 1$, hence the norms satisfy $\|\nabla_{\mathbf{V}} h_t\|_F \leq \|\mathbf{a}\|_F$, and $\|\nabla_{\mathbf{V}}^2 h_t\|_{opt} \leq \frac{1}{4}\|\mathbf{a}\|_F^2$. □

Lemma 2. *Let $\widetilde{\mathbf{a}}_t$ be bounded by Ω, for $\forall t = 1, \cdots, N$. Based on* (13) *we have $\|\widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1}\|_F \leq \eta_t \Omega$.*

Proof. From Equation (12), we have $\|\widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1}\|_F = \eta_t \|\nabla_{\mathbf{V}} h_t\|_F$. Since $\widetilde{\mathbf{a}}_t$ result from a regularized problem in (10), so $\widetilde{\mathbf{a}}_t$ is bounded by Ω. Thus we have $\|\widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1}\|_F \leq \eta_t \|\widetilde{\mathbf{a}}_t\|_F \leq \eta_t \Omega$. □

Lemma 3. *For $h_t(\cdot)$ in* (9)*, $\langle \mathbf{a}, \nabla_{\mathbf{a}} h_t \rangle = \langle \mathbf{V}, \nabla_{\mathbf{V}} h_t \rangle$. Hence, for $t = 1, \cdots, N$, $\eta_t \gamma \|\widetilde{\mathbf{a}}_t\|_F^2 = \langle \widetilde{\mathbf{V}}^{t-1}, -\eta_t \nabla_{\mathbf{V}} h_t \rangle = \langle \widetilde{\mathbf{V}}^{t-1}, \widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1} \rangle$.*

This follows directly from (10) and (12).

Lemma 4. *$h_t(\cdot)$ and surrogate function $\widetilde{h}_t(\cdot)$, as well as their first derivative $\nabla h_t(\cdot)$ and $\nabla \widetilde{h}_t(\cdot)$ are all Lipschitz continuous.*

This is indicated directly from Lemma 1 & Lemma 2 and the definition of Lipschitz continuous [16].

Theorem 1 (Proposition 2, [14])**.** *Under the regularity condition of Lemma 4, and $h_t(\cdot)$ a marginally convex function, $\widehat{C}_N(\widetilde{\mathbf{V}}^N)$ converges a.s. to $C_N(\mathbf{V}^*)$.*

The Proof has been implemented in [17] and [14], following a quasi-martingale theory, and use the Bregman divergence under surrogate function as a bridge $\widetilde{h}_t(\cdot)$.

Theorem 2. *Given step size as $\eta_t = C \times t^{-1/2}$ or $\eta_t = C$, the Regret $\widehat{Re}_N$ converges to within a constant of $\widehat{C}_N(\widetilde{\mathbf{V}}^N)$, and thus converges to within a constant of $C_N(\mathbf{V}^*)$.*

A sketch of proof is given in *Appendix A*. The results basically show that $\lim_{N\to\infty} |\widehat{Re}_N - \widehat{C}_N(\widetilde{\mathbf{V}}^N)| \leq \frac{\gamma\Omega^2}{2}$ if $\eta_t = Ct^{-1/2}$ and $\lim_{N\to\infty} |\widehat{Re}_N - \widehat{C}_N(\widetilde{\mathbf{V}}^N)| \leq \gamma\Omega^2 + C\Omega^2$ if $\eta_t = C$, Ω as a constant. From Theorem 1 & Theorem 2, we recognize that both the average sequential function and Regret function converge to within a constant from the average batch optimum. However, it should be noted here that a better convergence result could be possible, probably by re-design the algorithms, which is one of our future tasks.

IV. Experimental Results

A. Simulated Binary-State System

Firstly, we use simulated binary data to test the performance of our SLPCA algorithm in binary-state system. The generation of correlated Bernoulli sequences is illustrated in [18]. In this work, we focus on the case where $\text{rank}(\mathbf{\Theta}) = 1$ since this usually demonstrates the best dimension reduction capability. It should be noted here that the extension to multiple Principal Components is straight-forward following the iterative updating rules in [4].

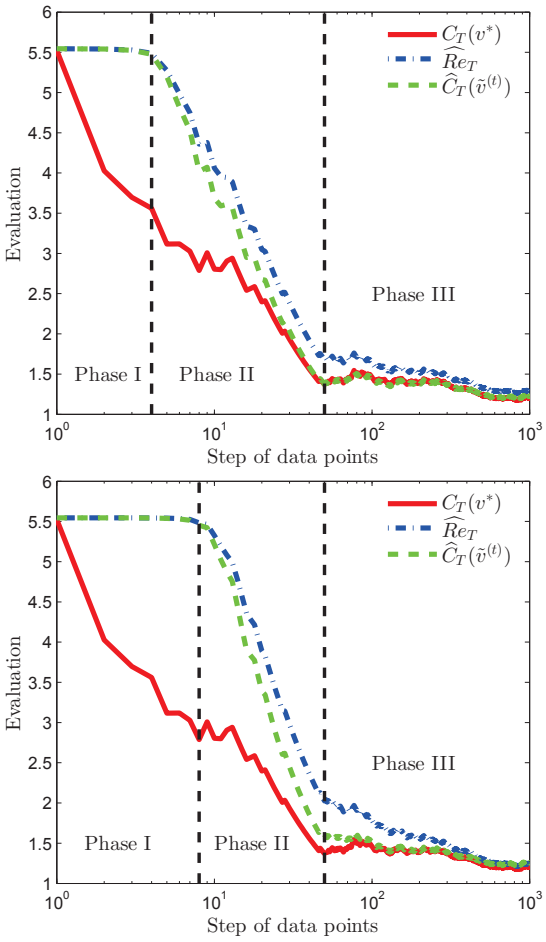

Fig. 1: The three functions $C_t(\mathbf{V}^*)$, $\widehat{C}_t(\widetilde{\mathbf{V}}^t)$ and $\widehat{Re}_t$ as function of t. Top: $\eta_t = Ct^{-1/2}$, with $C = 0.2$, $\gamma = 0.1$. Bottom: $\eta_t = C$, with $C = 0.05$, $\gamma = 0.1$.

We tried the above on data with $P = 8$ dimension and length of $N = 1000$ data points. We initialize $\widetilde{\mathbf{V}}^0$ such that its norm is close but not equal to zero, for computation and convergence purposes. Fig 1 shows the three functions defined in (15); whereas Fig 2 shows the key parameters in the sequential steps. There are some interesting findings.

Firstly, though both $\widehat{C}_N(\widetilde{\mathbf{V}}^N)$ and $\widehat{Re}_N$ converges at least within a constant to $C_N(\mathbf{V}^*)$, the stochastic learning can be clearly divided into three Phases, as shown in Fig 1. Phase I stands for the period when the norm of $\widetilde{\mathbf{V}}^0$ is close to

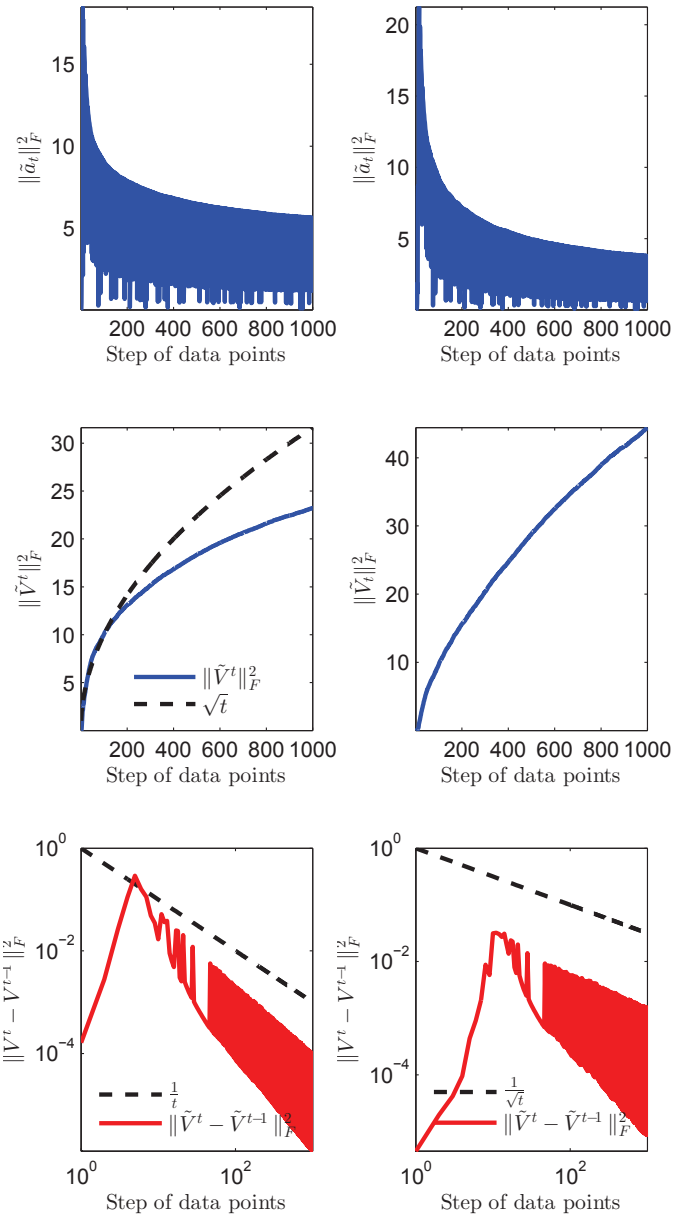

Fig. 2: The convergence property of $\widetilde{\mathbf{a}}_t$, $\widetilde{\mathbf{V}}^t$ and $\|\widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1}\|_F$. Top: $\eta_t = Ct^{-1/2}$, with $C = 0.2$, $\gamma = 0.1$. Bottom: $\eta_t = C$, with $C = 0.05$, $\gamma = 0.1$.

zero right after the initialization, when $h_t(\mathbf{a}_t, \mathbf{V})$ approaches $P \log 2$ as in Equation (14). Phase II characterizes the decay of error versus N, whereas Phase III stands for when the error converges to within a constant independent of N.

Secondly, $\|\widetilde{\mathbf{V}}^t\|_F^2$ increases versus t, which means that $\|\widetilde{\mathbf{V}}^t\|_F^2$ behaves differently from the coefficient in sequential learning of linear model [17] [14]. Matrix factorization places no constraints for $\widetilde{\mathbf{V}}^t$, hence cannot guarantee the bound of $\widetilde{\mathbf{V}}^t$. From another perspective, $\widetilde{\mathbf{a}}_t$ is bounded since Equation (10) has fixed in size, while $\widetilde{\mathbf{V}}_t$ not since there is a summation of loss functions. It should be noted that, in Fig 2, $\widetilde{\mathbf{a}}_t$ decreases

versus t, which could result from (9) and is an interesting topic in the future.

Thirdly, due to the unbounded $\widetilde{\mathbf{V}}^t$, the term $\|\widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1}\|_F$ is not $\propto t^{-1}$ as in [17] and [14]. It should be noted that the theoretical bound for $\|\widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1}\|_F$ under constant step size could be as low as $t^{-1/2}$, which could be a result of the convergence behavior of $\widetilde{\mathbf{a}}_t$ under constant step size.

Last but not least, it is important to mention that the bounds obtained in Theorem 2 assume N large enough. However, in many cases the decay of N is not that fast. Therefore, the effect of N cannot be completely ignored in the analysis.

B. Building End-Use Energy Modeling

Here, we introduce an application of SLPCA in Building Energy End-Use Modeling. Building End-Uses corresponds to the energy sectors that are occupant-driven. This subject has attracted significant interest in recent years because building energy shows strong dependence on end-user behavior, e.g. plug-in loads, user-controlled lighting, user-adjusted HVAC, etc. [19] [20].

Energy end-use modeling has been attempted from either a top-down or a bottom-up approach. In this work, since we are more interested in modeling occupant behavior, we adopt the bottom-up approach. This approach is usually based on stochastic simulations of the energy usage pattern for each individual appliance. Dimensional reduction can help to generate one or more *Principal Appliances*, and can more efficiently characterize the whole space energy consumption.

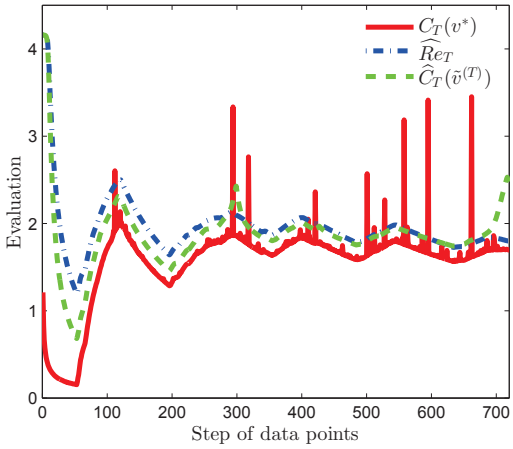

Fig. 3: The three functions $C_t(\mathbf{V}^*)$, $\widehat{C}_t(\widetilde{\mathbf{V}}^t)$ and $\widehat{Re}_t$ as function of t for energy end-use simulation with constant step size $\eta_t = C$ as $C = 0.05$, $\gamma = 0.1$.

Here, we want to study the modeling of all the computer monitors in a small, shared work space. We collect the data of 6 monitors in 10 minutes interval, and use BLPCA and SLPCA to obtain the *Principal Monitor* profile of the building. Considering that the pattern could be non-stationary, we choose the constant step size that is short enough to track the changes as they appear[4]. We also only consider the first *Principal Monitor* to achieve the best dimensional reduction. The convergence of the algorithm is shown in Fig 3. We observe a good convergence for both $\widehat{C}_N(\widetilde{\mathbf{V}}^N)$ and $\widehat{Re}_N$. Periodic fluctuation is observed, due to the transition between day and night energy consumption, which results in periodical changing of the data model. Moreover, the online algorithm demonstrate less fluctuations because they adaptively update the model of the data.

The reconstruct of the original data is done by three sets of variables: the BBD setting $\mathbf{A}^*$, $\mathbf{V}^*$; the OBD setting $\{\widetilde{\mathbf{a}}_t\}$, $\widetilde{\mathbf{V}}^N$; and the RBD setting $\{\widetilde{\mathbf{a}}_t\}$, $\{\widetilde{\mathbf{V}}^t\}$. The results are compared with the original data in Fig 4 (sum of states of all appliances, 1 as ON and 0 as OFF). Interestingly, OBD setting gives better approximation to BBD setting since it is more adaptive in terms of $\widetilde{\mathbf{V}}^t$ and can better catch the periodic pattern of the original data. On the other hand, BBD setting uses the $\widetilde{\mathbf{V}}^N$, which could give unpromising result if data is non-stationary.

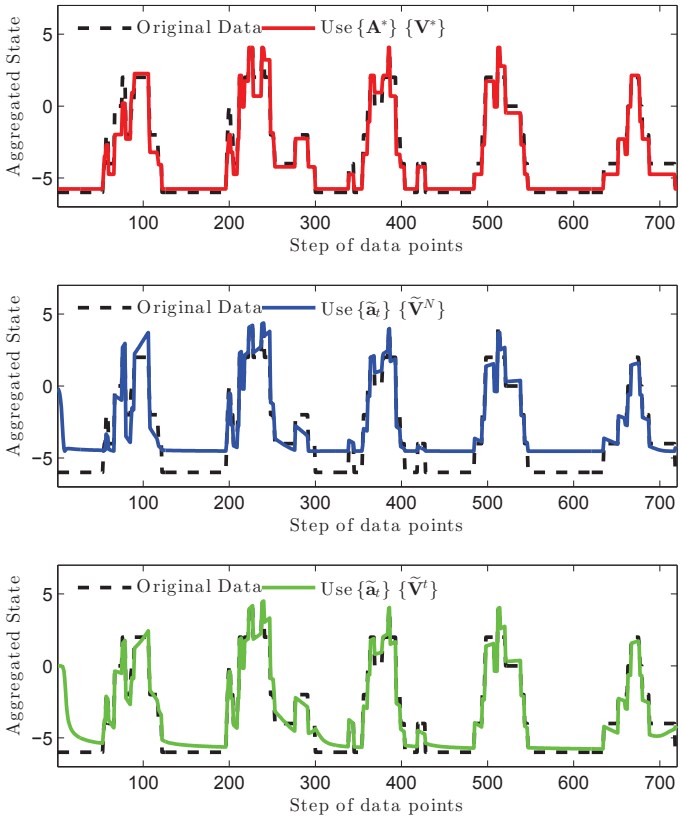

Fig. 4: Reconstruction of the aggregated state (sum of states of 6 monitors) under the three sets of variables.

V. Conclusion

Sequential or online dimension reduction addresses more and more attentions due to the explosion of streaming data based application and the requirement of adaptive statistical modeling in many emerging fields. In this work, we extend the theory of *e*PCA or LPCA to sequential version based on online

[4]one could presumably also leverage the likely periodic behavior of the data by appropriate aggregation

convex optimization theory, which can maintain the capability to model large families of distributions, at the same time achieve the computation and storage efficiency. In our work, we define two functions to evaluate the SLPCA algorithm, the average sequential target function $\widehat{C}_N(\widetilde{\mathbf{V}}^N)$ and the Regret function $\widehat{Re}_N$, and show that both of them converge at least within a constant to BLPCA results. We also demonstrate an application of this algorithm in building energy end-use modeling.

APPENDIX

Lemma 5. *For $t = 1, \cdots, N$, if Ω is the upper bound of $\|\mathbf{a}\|^2_{opt}$ as in Lemma 2, $\|\widetilde{\mathbf{V}}^t\|^2_F \leq \Omega^2 \sum_{s=1}^t \eta_s^2 + 2\gamma\Omega^2 \sum_{s=1}^t \eta_s$.*

Proof. We start from the relationship:

$$\begin{aligned}\|\widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1}\|^2_F &= \|\widetilde{\mathbf{V}}^t\|^2_F - \|\widetilde{\mathbf{V}}^{t-1}\|^2_F - 2\langle \widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^{t-1}, \widetilde{\mathbf{V}}^{t-1}\rangle \\ &= \|\widetilde{\mathbf{V}}^t\|^2_F - \|\widetilde{\mathbf{V}}^{t-1}\|^2_F - 2\eta_t\gamma\|\widetilde{\mathbf{a}}^t\|^2_F\end{aligned}$$

We sum over the LHS and RHS and get:

$$\sum_{s=1}^t \|\widetilde{\mathbf{V}}^s - \widetilde{\mathbf{V}}^{s-1}\|^2_F + 2\gamma\sum_{s=1}^t \eta_s\|\widetilde{\mathbf{a}}_s\|^2_F = \|\widetilde{\mathbf{V}}^t\|^2_F - \|\widetilde{\mathbf{V}}^0\|^2_F$$

For simplicity, assume $\|\widetilde{\mathbf{V}}^0\|^2_F \approx 0$, we prove the lemma. □

Now turn to proof of Theorem 2. Based on (13) we have:

$$\begin{aligned}\|\widetilde{\mathbf{V}}^t - \widetilde{\mathbf{V}}^N\|^2_F = \|\widetilde{\mathbf{V}}^{t-1} - \widetilde{\mathbf{V}}^N\|^2_F + \eta_t^2\|\nabla_{\mathbf{V}} h_t\|^2_F \\ - 2\eta_t\langle \nabla_{\mathbf{V}} h_t, \widetilde{\mathbf{V}}^{t-1} - \widetilde{\mathbf{V}}^N\rangle\end{aligned}$$

From Lemma 1, Lemma 5, and $\|\nabla_{\mathbf{V}} h_t\|^2_F \leq \Omega^2$, thus:

$$\begin{aligned}&N\{\widehat{Re}_N - \widehat{C}_N(\widetilde{\mathbf{V}}^N)\} \leq \sum_{t=1}^N \langle \nabla_{\mathbf{V}} h_t, \widetilde{\mathbf{V}}^{t-1} - \widetilde{\mathbf{V}}^N\rangle \\ &\leq \frac{\|\widetilde{\mathbf{V}}^N\|^2_F}{2\eta_0} + \sum_{t=1}^N \left(\frac{1}{2\eta_t} - \frac{1}{2\eta_{t-1}}\right)\|\widetilde{\mathbf{V}}^N - \widetilde{\mathbf{V}}^{t-1}\|^2_F + \frac{\Omega^2}{2}\eta_t \\ &\leq \frac{\|\widetilde{\mathbf{V}}^N\|^2_F}{2\eta_0} + \sum_{t=1}^N \left(\frac{1}{2\eta_t} - \frac{1}{2\eta_{t-1}}\right)\|\widetilde{\mathbf{V}}^N\|^2_F + \frac{\Omega^2}{2}\eta_t\end{aligned}$$

- diminishing step size $\eta_t = Ct^{-1/2}$. From Lemma 5, we have:

$$\begin{aligned}|\widehat{Re}_N - \widehat{C}_N(\widetilde{\mathbf{V}}^N)| \leq \frac{\Omega^2 C}{2}\frac{\log N}{N} + \frac{\Omega^2 C}{4}\frac{\log N}{\sqrt{N}} \\ + \frac{\Omega^2(2\gamma + C)}{2\sqrt{N}} + \frac{\gamma\Omega^2}{2}\end{aligned}$$

  Then $\lim_{N\to\infty}|\widehat{Re}_N - \widehat{C}_N(\widetilde{\mathbf{V}}^N)| \leq \frac{\gamma\Omega^2}{2}$. But with reasonable N, the term $\frac{\Omega^2 C\log N}{\sqrt{N}}$ will also be significant. Usually, small C and γ can force a lower error bound. However, small γ can result in more steps in optimizing for $\widetilde{\mathbf{a}}_t$, whereas small C would make the step size too small, which may not be a good choice if we want a fast decaying of the error bound.

- constant step size $\eta_t = C$: For constant step, we have:

$$|\widehat{Re}_N - \widehat{C}_N(\widetilde{\mathbf{V}}^N)| \leq \gamma\Omega^2 + \Omega^2 C$$

  Similarly, we prefer small small C and γ. The challenge of using small C and γ have already been discussed.

ACKNOWLEDGMENT

This research is funded by the Republic of SingaporeâĂŹs National Research Foundation through a grant to the Berkeley Education Alliance for Research in Singapore (BEARS) for the Singapore-Berkeley Building Efficiency and Sustainability in the Tropics (SinBerBEST) Program. BEARS has been established by the University of California, Berkeley as a center for intellectual excellence in research and education in Singapore.

REFERENCES

[1] A. I. Schein, L. K. Saul, and L. H. Ungar, "A generalized linear model for principal component analysis of binary data," in *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, vol. 38, 2003, p. 46.

[2] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, and R. Tibshirani, *The elements of statistical learning*. Springer, 2009, vol. 2, no. 1.

[3] S. Papadimitriou, J. Sun, and C. Faloutsos, "Streaming pattern discovery in multiple time-series," in *Proceedings of the 31st international conference on Very large data bases*. VLDB Endowment, 2005, pp. 697–708.

[4] M. Collins, S. Dasgupta, and R. E. Schapire, "A generalization of principal components analysis to the exponential family," in *Advances in neural information processing systems*, 2001, pp. 617–624.

[5] A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh, "Clustering with bregman divergences," *The Journal of Machine Learning Research*, vol. 6, pp. 1705–1749, 2005.

[6] L. M. Bregman, "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR computational mathematics and mathematical physics*, vol. 7, no. 3, pp. 200–217, 1967.

[7] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," 2003.

[8] S. Shalev-Shwartz, "Online learning and online convex optimization," *Foundations and Trends in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2011.

[9] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 3, pp. 611–622, 1999.

[10] I. Csisz, G. Tusnády *et al.*, "Information geometry and alternating minimization procedures," *Statistics and decisions*, 1984.

[11] M. Mardani, G. Mateos, and G. Giannakis, "Decentralized sparsity-regularized rank minimization: Algorithms and applications," *Signal Processing, IEEE Transactions on*, vol. 61, no. 21, pp. 5374–5388, 2013.

[12] J. Abernethy, F. Bach, T. Evgeniou, and J.-P. Vert, "A new approach to collaborative filtering: Operator estimation with spectral regularization," *The Journal of Machine Learning Research*, vol. 10, pp. 803–826, 2009.

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

[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] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.

[16] D. P. Bertsekas, "Nonlinear programming," 1999.

[17] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *The Journal of Machine Learning Research*, vol. 11, pp. 19–60, 2010.

[18] A. D. Lunn and S. J. Davies, "A note on generating correlated binary variables," *Biometrika*, vol. 85, no. 2, pp. 487–490, 1998.
[19] L. G. Swan and V. I. Ugursal, "Modeling of end-use energy consumption in the residential sector: A review of modeling techniques," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 8, pp. 1819–1835, 2009.
[20] Z. Kang, M. Jin, and C. J. Spanos, "Modeling of end-use energy profile: An appliance-data-driven stochastic approach," *arXiv preprint arXiv:1406.6133*, 2014.