SpectralLeader: Online Spectral Learning for Single Topic Models

Abstract

We study the problem of learning a latent variable model from a stream of data. Latent variable models are popular in practice because they can explain observed data in terms of unobserved concepts. These models have been traditionally studied in the offline setting. The online EM is arguably the most popular algorithm for learning latent variable models online. Although it is computationally efficient, it typically converges to a local optimum. In this work, we develop a new online learning algorithm for latent variable models, which we call SpectralLeader. SpectralLeader always converges to the global optimum, and we derive a $O(\sqrt{n})$ upper bound up to log factors on its n-step regret in the bag-of-words model. We show that SpectralLeader performs similarly to or better than the online EM with tuned hyper-parameters, in both synthetic and real-world experiments.

1 Introduction

Latent variable models are classical approaches to explain observed data via unobserved concepts. They have been successfully applied in a wide variety of fields, such as speech recognition, natural language processing, and computer vision [Rabiner, 1989; Wallach, 2006; Nowozin and Lampert, 2011; Bishop, 2006]. Despite their successes, latent variable models are typically studied in the offline setting. However, in many practical problems, a learning agent needs to learn a latent variable model online while interacting with real-time streaming data with unobserved concepts. For instance, a recommender system may want to learn to cluster its users online based on their real-time behavior. The goal of this paper is to develop algorithms for such online learning problems.

Previous works proposed several algorithms to learn latent variable models online by extending the expectation maximization (EM) algorithm. Those algorithms are known as online EM algorithms, and include the stepwise EM [Cappé and Moulines, 2009; Liang and Klein, 2009] and the incremental EM [Neal and Hinton, 1998]. Similar to the offline EM, each iteration of an online EM algorithm includes an E-step to fill in the values of latent variables based on their estimated distribution, and an M-step to update the model parameters. The main difference is that each step of online EM algorithms only uses data received in the current iteration, rather than the whole dataset. This ensures that online EM algorithms are computationally efficient and can be used to learn latent variable models online. However, similar to the offline EM, online EM algorithms have one major drawback: they may converge to a local optimum and hence suffer from a non-diminishing performance loss.

To overcome these limitations, we develop an online learning algorithm that performs almost as well as the globally optimal latent variable model, which we call SpectralLeader. Specifically, we propose an online learning variant of the spectral method [Anandkumar *et al.*, 2014], which can learn the parameters of latent variable models offline with guarantees of convergence to a global optimum. Our online learning setting is defined as follows. A sequence of latent topics is fixed in advance, one topic at each time. This sequence does not have to be stochastic. The words in the document at each time are generated i.i.d. conditioned on the topic at that time. The sampling distribution does not change over time. At each time, the learning agent observes a document and updates its model parameters. The goal of the agent is to predict the sequence of parameters with low cumulative regret with respect to the best solution in hindsight, which knows all topics and the sampling distribution of the words in advance.

This paper makes multiple contributions. First, it is the first paper to formulate online learning with the spectral method as a regret minimization problem. This includes the definition of the loss function, which is non-trivial because the loss function of the spectral method is non-decomposable. Second, we propose SpectralLeader, an online learning variant of the spectral method for our problem. Third, we prove a $O(\sqrt{n})$ upper bound up to log factors on the n-step regret of SpectralLeader. Finally, we compare SpectralLeader to the stepwise EM in extensive synthetic and real-world experiments. We observe that the stepwise EM is sensitive to its hyper-parameters. In all experiments, SpectralLeader performs similarly to or better than the stepwise EM with tuned hyper-parameters.

2 Related Work

The spectral method by tensor decomposition has been widely applied in different latent variable models such as mixtures of tree graphical models [Anandkumar *et al.*, 2014], mixtures of linear regressions [Chaganty and Liang, 2013], hidden Markov models (HMM) [Anandkumar *et al.*, 2012b], latent Dirichlet allocation (LDA) [Anandkumar *et al.*, 2012a], Indian buffet process [Tung and Smola, 2014], and hierarchical Dirichlet process [Tung *et al.*, 2017]. One major advantage of the spectral method is that it learns globally optimal solutions [Anandkumar *et al.*, 2014]. The spectral method first empirically estimates low-order moments of observations, and then applies decomposition methods with a unique solution, to recover the model parameters.

Traditional online learning methods for latent variable models usually extend traditional iterative methods for learning latent variable models in the offline setting. Offline EM calculates the sufficient statistics based on all the data, while in online EM the sufficient statistics are updated with the recent data in each iteration [Cappé and Moulines, 2009; Neal and Hinton, 1998; Liang and Klein, 2009]. Online variational inference is used to learn LDA efficiently [Hoffman *et al.*, 2010]. These online iterative algorithms converge to local minima, while we aim to develop an algorithm with a theoretical guarantee of convergence to a global optimum.

An online spectral learning method has also been developed [Huang *et al.*, 2015], with a focus on improving computational efficiency, by conducting optimization of multilinear operations in SGD and avoiding directly forming the tensors. Online stochastic gradient for tensor decomposition has been analyzed [Ge *et al.*, 2015] with a different online setting: they do not look at the online problem as regret minimization and the analysis focuses on convergence to a local minimum. In contrast, we develop an online spectral method with a theoretical guarantee of convergence to a global optimum. Besides, our online spectral method is robust in the non-stochastic setting where the topics of documents are correlated over time. This non-stochastic setting has not been previously studied in the context of online spectral learning [Huang *et al.*, 2015].

3 Spectral Method for Topic Model

This section introduces the spectral method in latent variable models. Specifically, we describe how the method works in the simple bag-of-words model [Anandkumar *et al.*, 2014].

In the bag-of-words model, the goal is to understand the latent topic of documents based on the observed words in each document. Without loss of generality, we describe the spectral method and `SpectralLeader` (Section 5) in the setting where the number of words in each document is 3. The extension to more than 3 words is straightforward [Anandkumar *et al.*, 2014]. Let the number of distinct topics be K and the size of the vocabulary be d. Then our model can be viewed as a mixture model, where the observed words $\mathbf{x}^{(1)}$, $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$ are conditionally i.i.d. given topic C, which is also a random variable. Each word is one-hot encoded, $\mathbf{x}^{(l)} = e_i$ if and only if $\mathbf{x}^{(l)}$ represents word i, where $e_1, e_2, \ldots, e_d$ is the standard coordinate basis in $\mathbb{R}^d$. The model is parameterized by the probability of each topic j, $\omega_j = P(C = j)$ for $j \in [K]$, and the conditional probability of all words $u_j \in [0,1]^d$ given topic j. The ith entry of u_j is $u_j(i) = P(\mathbf{x}^{(l)} = e_i | C = j)$ for $i \in [d]$. With 3 observed words, it suffices to construct a third order tensor $\bar{M}_3$ as

$$\mathbb{E}[\mathbf{x}^{(1)} \otimes \mathbf{x}^{(2)} \otimes \mathbf{x}^{(3)}] = \sum_{1 \le i,j,k \le d} P(\mathbf{x}^{(1)} = e_i, \mathbf{x}^{(2)} = e_j, \mathbf{x}^{(3)} = e_k) e_i \otimes e_j \otimes e_k. \quad (1)$$

To recover the parameters of the topic model, we want to decompose $\bar{M}_3$ as $\bar{M}_3 = \sum_{i=1}^K \omega_i u_i \otimes u_i \otimes u_i$.

Unfortunately, such a decomposition is generally NP-hard [Anandkumar *et al.*, 2014]. Instead, we can efficiently obtain the decomposition for orthogonal decomposable tensors. One way to make $\bar{M}_3$ orthogonal decomposable is by whitening. We can define a whitening matrix as $\bar{W} = UA^{-1/2}$, where $A \in \mathbb{R}^{K \times K}$ is the diagonal matrix of positive eigenvalues of $\bar{M}_2 = \mathbb{E}[\mathbf{x}^{(1)} \otimes \mathbf{x}^{(2)}] = \sum_{i=1}^K \omega_i u_i \otimes u_i$, and $U \in \mathbb{R}^{d \times K}$ is the matrix of the eigenvectors associated with those eigenvalues. After whitening, instead of decomposing $\bar{M}_3$, we can decompose $\bar{T} = \bar{W}^\top \mathbb{E}[\mathbf{x}^{(1)}] \otimes \bar{W}^\top \mathbb{E}[\mathbf{x}^{(2)}] \otimes \bar{W}^\top \mathbb{E}[\mathbf{x}^{(3)}]$ as $\bar{T} = \sum_{i=1}^K \lambda_i v_i \otimes v_i \otimes v_i$ by the *power iteration method* [Anandkumar *et al.*, 2014]. Finally, the model parameters are recovered by $\omega_i = \frac{1}{\lambda_i^2}$ and $u_i = \lambda_i (\bar{W}^\top)^+ v_i$, where $(\bar{W}^\top)^+$ is the pseudoinverse of $\bar{W}^\top$.

Only a noisy realization of $\bar{T}$, T, is typically available in practice, and it is constructed from empirical counts. Such tensors can be decomposed approximately, and the error of such a decomposition is analyzed in Theorem 5.1 of Anandkumar *et al.* [2014].

To simplify the exposition of tensor decompositions, we introduce the following notation. Let $T \in \mathbb{R}^{K \times K \times K}$ be any tensor and $\theta = ((\lambda_i)_{i=1}^K, (v_i)_{i=1}^K)$ be its decomposition by the power iteration method. Then $f(T) = \theta$, function f decomposes tensor T, and $\mathcal{T}(\theta) = \sum_{i=1}^K \lambda_i v_i \otimes v_i \otimes v_i$, function $\mathcal{T}$ builds the tensor from θ. If tensor T is perfectly decomposable, then $T = \mathcal{T}(f(T))$. The expected tensor $\bar{T}$ is one such tensor which is perfectly decomposable.

4 Online Learning for Topic Models

We study the following online learning problem in a single topic model. Fix a sequence of n latent topics $(C_t)_{t=1}^n$, one at each time t. We denote by $\mathbf{x}_t = (\mathbf{x}_t^{(l)})_{l=1}^3$ a tuple of one-hot encoded words in the document at time t. At time t, each word $\mathbf{x}_t^{(l)}$ is generated i.i.d. conditioned on C_t. The sampling distribution of the words does not change over time.

The goal of the learning agent is to predict a sequence of model parameters with low cumulative regret, with respect to the best solution in hindsight of knowing $(C_t)_{t=1}^n$ and the sampling distribution of the words. Formally, let $\ell_t(\theta)$ be the loss of solution θ at time t, $\hat{\theta}_{t-1}$ be the *solution of the learning agent* at time t, and $\theta^* = \arg\min_\theta \sum_{t=1}^n \ell_t(\theta)$ be the *best solution in hindsight*. Then our goal is to minimize

$$R(n) = \sum_{t=1}^n \ell_t(\hat{\theta}_{t-1}) - \sum_{t=1}^n \ell_t(\theta^*). \quad (2)$$

To define the notion of the loss, we introduce the following notation. Let $T_t = \frac{1}{t} \sum_{z=1}^t Y_{t,z}$ be the *tensor from the first t observations*, where $Y_{t,z} = \frac{1}{|\Pi_3(3)|} \sum_{\pi \in \Pi_3(3)} W_t^\top \mathbf{x}_z^{(\pi(1))} \otimes W_t^\top \mathbf{x}_z^{(\pi(2))} \otimes W_t^\top \mathbf{x}_z^{(\pi(3))}$, W_t is the whitening matrix constructed from these observations, and $\Pi_3(3)$ is the set of all 3-permutations of [3]. Similarly, let $\bar{T}_t = \frac{1}{t} \sum_{z=1}^t \bar{Y}_{t,z}$ be the *expected tensor from the first t observations*, where $\bar{Y}_{t,z} = \bar{W}_t^\top \mathbb{E}[\mathbf{x}_z^{(1)}] \otimes \bar{W}_t^\top \mathbb{E}[\mathbf{x}_z^{(2)}] \otimes \bar{W}_t^\top \mathbb{E}[\mathbf{x}_z^{(3)}]$, $\bar{W}_t$ is the whitening matrix from the first t expected observations, and $\mathbb{E}[\mathbf{x}_z^{(l)}]$ is conditioned on a fixed topic C_z at time z.

The spectral method [Anandkumar *et al.*, 2014] can be viewed as follows. When an empirical tensor is close to its

expectation, then its decomposition is close to that of the expected tensor in the tensor operator norm. Therefore, if θ is the decomposition of the empirical tensor from t observations, the spectral method minimizes the *cumulative loss*

$$\mathcal{L}_t(\theta) = t\|\bar{T}_t - \mathcal{T}(\theta)\|^p \,, \tag{3}$$

where $\|\cdot\|$ is the *tensor operator norm* in Section 5.2 of Anandkumar *et al.* [2014] and p can be any positive integer. This loss is *non-decomposable* [Kar *et al.*, 2014]. Motivated by Kar *et al.* [2014], we define the *loss* at time t as

$$\begin{aligned}\ell_t(\theta) &= \mathcal{L}_t(\theta) - \mathcal{L}_{t-1}(\theta) \\ &= t\|\bar{T}_t - \mathcal{T}(\theta)\|^p - (t-1)\|\bar{T}_{t-1} - \mathcal{T}(\theta)\|^p\end{aligned} \tag{4}$$

and assume that $\mathcal{L}_0(\theta) = 0$. The main reason for the above formulation is that the sum of the first t losses, $\sum_{z=1}^{t} \ell_z(\theta) = t\|\bar{T}_t - \mathcal{T}(\theta)\|^p$, is identical to (3). Roughly speaking, $\ell_t(\theta)$ measures the additional loss due to modeling the outcome at time t without knowing it. Another reason is that the cumulative loss of the best solution in hindsight θ^* is zero. In particular, note that $\sum_{t=1}^{n} \ell_t(\theta^*) = 0$ when $\theta^* = f(\bar{T}_n)$, since $\bar{T}_n$ can be perfectly decomposed and reconstructed as discussed in Section 3. This is the lowest achievable loss.

Unlike traditional online learning algorithms that minimize the negative log-likelihood in latent variable models, our online algorithm minimizes the recovery loss in (3). In the offline setting, the spectral method minimizes the recovery loss in a wide range of latent variable models [Anandkumar *et al.*, 2014; Chaganty and Liang, 2013; Shaban *et al.*, 2015].

5 Algorithm `SpectralLeader`

We propose `SpectralLeader`, an online learning algorithm for minimizing the regret in (2). Its pseudocode is in Algorithm 1. At time t, the inputs of `SpectralLeader` are one-hot encoded words $((\mathbf{x}_z^{(l)})_{l=1}^3)_{z=1}^{t-1}$ from the first $t-1$ time steps. The algorithm operates as follows. First, we construct the second-order moment from the input words, where $\Pi_2(3)$ is the set of all 2-permutations of $[3]$. Then we estimate A_{t-1} and U_{t-1} by eigendecomposition, and construct the whitening matrix W_{t-1}. After whitening, we build the third-order tensor T_{t-1} from whitened words $((\mathbf{y}_z^{(l)})_{l=1}^3)_{z=1}^{t-1}$, where $\Pi_3(3)$ is the set of all 3-permutations of $[3]$. Then we decompose T_{t-1} with the power iteration method and get $\hat{\theta}_{t-1} = ((\lambda_{t-1,i})_{i=1}^K, (v_{t-1,i})_{i=1}^K) = f(T_{t-1})$. Finally, we recover the parameters of the model, $u_{t-1,i}$ and $\omega_{t-1,i}$.

5.1 Discussion

`SpectralLeader` is relatively easy to analyze (Section 6). However, `SpectralLeader` is not computationally efficient, because its time complexity at time t is linear in t. In particular, the operations in lines 1 and 4 of Algorithm 1 depend on t because all past observations are used to construct $M_{2,t-1}$ and T_{t-1}. Besides, the whitening operation in line 3 depends on t because all past observations are whitened by a matrix W_{t-1} that changes with t.

To resolve this issue, we design a more space and computationally efficient algorithm using *reservoir sampling*. The key idea is to maintain a pool of R observations $\mathbf{x}_z$, $z \in [t-1]$.

Algorithm 1: `SpectralLeader` at time t

Input: $((\mathbf{x}_z^{(l)})_{l=1}^3)_{z=1}^{t-1}$

1 $M_{2,t-1} = \frac{1}{(t-1)|\Pi_2(3)|} \sum_{z=1}^{t-1} \sum_{\pi \in \Pi_2(3)} \mathbf{x}_z^{(\pi(1))} \otimes \mathbf{x}_z^{(\pi(2))}$

2 $W_{t-1} = U_{t-1} A_{t-1}^{-1/2}$

// $A_{t-1} \in \mathbb{R}^{K \times K}$ is the diagonal matrix of K positive eigenvalues of $M_{2,t-1}$

// $U_{t-1} \in \mathbb{R}^{d \times K}$ is the matrix of eigenvectors associated with these positive eigenvalues

3 $\forall z \in [t-1], l \in [3] : \mathbf{y}_z^{(l)} = W_{t-1}^\top \mathbf{x}_z^{(l)}$

4 $T_{t-1} = \frac{1}{(t-1)|\Pi_3(3)|} \sum_{z=1}^{t-1} \sum_{\pi \in \Pi_3(3)} \mathbf{y}_z^{(\pi(1))} \otimes \mathbf{y}_z^{(\pi(2))} \otimes \mathbf{y}_z^{(\pi(3))}$

5 $\hat{\theta}_{t-1} = ((\lambda_{t-1,i})_{i=1}^K, (v_{t-1,i})_{i=1}^K) = f(T_{t-1})$

// f is defined in Section 3

6 $\omega_{t-1,i} = \frac{1}{\lambda_{t-1,i}^2}, \quad u_{t-1,i} = \lambda_{t-1,i} (W_{t-1}^\top)^+ v_{t-1,i}$

Output: Model parameters $\omega_{t-1,i}$ and $u_{t-1,i}$

When $t \le R$, a new observation is added to the pool. When $t > R$, a new observation replaces a random observation in the pool with probability $R/(t-1)$. Then we can approximate T_{t-1} using only the observations in the pool. With reservoir sampling, `SpectralLeader` has per-step space and computational cost independent of t. We focus on analyzing `SpectralLeader` without reservoir sampling, and leave the analysis with reservoir sampling for future work.

6 Analysis

In this section, we bound the regret of `SpectralLeader`. We analyze two settings. In the first setting, `SpectralLeader` is assumed to act based on the expected tensor at time t, $\hat{\theta}_{t-1} = f(\bar{T}_{t-1})$. In the second setting, `SpectralLeader` acts based on the empirical tensor at time t, $\hat{\theta}_{t-1} = f(T_{t-1})$, which is a noisy realization of $\bar{T}_{t-1}$. The noise is due to the random realizations of words. We make the following assumptions in our analysis. We discuss how to satisfy them in Section 6.4.

Assumption 1. *The values of $Y_{t,z}$ are bounded,*

$$\forall t, z \in [t], i, j, k : |Y_{t,z}(i,j,k)| \le r_1 \,,$$

where $Y_{t,z}$ is defined in Section 4 and $r_1 > 0$ is a constant.

Assumption 2. *At any time t,*

$$\forall t, z \in [t-1], i, j, k : |\bar{Y}_{t,z}(i,j,k) - \bar{Y}_{t-1,z}(i,j,k)| \le r_2 t^{-\frac{1}{2}} \,,$$

where $\bar{Y}_{t,z}$ is defined in Section 4 and $r_2 > 0$ is a constant.

As discussed in Section 4, the cumulative loss of θ^* is 0. Therefore, the problem of bounding the cumulative regret of a learning agent reduces to bounding its cumulative loss. For any tensor $T \in \mathbb{R}^{K \times K \times K}$, we define its Frobenius norm as $\|T\|_F = \sqrt{\sum_{i,j,k=1}^K T(i,j,k)^2}$. All constants from our analysis are listed in Table 1.

6.1 Noise-Free Setting

We first consider the noise-free setting, which is less realistic. At time t, `SpectralLeader` knows the expected tensor $\bar{T}_{t-1}$.

Theorem 1. *Let $\hat{\theta}_{t-1} = f(\bar{T}_{t-1})$ at all times t and $p \geq 3$. Then for any $(\bar{T}_t)_{t=1}^n$,*

$$R(n) \leq 2c_2^3\sqrt{n}\,.$$

Proof. Since $\bar{T}_{t-1}$ is decomposable, $\mathcal{T}(f(\bar{T}_{t-1})) = \bar{T}_{t-1}$ at any time t. Therefore, the loss at time t can be bounded from above as

$$\ell_t(f(\bar{T}_{t-1})) = t\|\bar{T}_t - \bar{T}_{t-1}\|^p \leq c_2^3 t^{-\frac{1}{2}}\,,$$

where the last inequality follows from Lemma 1 and $p \geq 3$. Finally, we sum up the losses at all t and get our claim. ■

6.2 Noisy Setting

Now we consider the noisy setting, which is realistic. At time t, `SpectralLeader` knows the noisy tensor T_{t-1}. The challenge is that T_{t-1} is only an approximation of $\bar{T}_{t-1}$. To characterize how this approximation affects the decomposition of T_{t-1}, we introduce a problem-specific constant

$$\lambda_{\min} = \min\{\lambda_{t,i} : t \in [n], i \in [K]\}\,, \tag{5}$$

where $\bar{T}_t = \sum_{i=1}^K \lambda_{t,i} v_{t,i} \otimes v_{t,i} \otimes v_{t,i}$ is an orthogonal decomposition of $\bar{T}_t$. Our main claim is below.

Theorem 2. *Let $\hat{\theta}_{t-1} = f(T_{t-1})$ at all times t and $p \geq 3$. Then in expectation over the random realizations of words,*

$$\mathbb{E}\left[R(n)\right] \leq 2c_2c_4K^3\sqrt{n} + c_2c_4c_1^2K^2c_3^{-2}\lambda_{\min}^{-2}\sqrt{n} + 2(c_2 + 55c_1)^p\sqrt{n}\,.$$

Proof. The key step in this proof is to note that

$$\begin{aligned}\ell_t(f(T_{t-1})) &\leq t\|\bar{T}_t - \mathcal{T}(f(T_{t-1}))\|^p \\ &= t\|\bar{T}_t - \bar{T}_{t-1} + \bar{T}_{t-1} - \mathcal{T}(f(T_{t-1}))\|^p \\ &\leq t(\|\bar{T}_t - \bar{T}_{t-1}\| + \|\bar{T}_{t-1} - \mathcal{T}(f(T_{t-1}))\|)^p\end{aligned}$$

at any time t. If $\|\bar{T}_{t-1} - \mathcal{T}(f(T_{t-1}))\|$ was on the order of $t^{-\frac{1}{2}}$, then $\ell_t(f(\bar{T}_{t-1}))$ would be $O(t^{-\frac{1}{2}})$ for $p \geq 3$, and the proof could be completed as in Theorem 1. The decomposition of T_{t-1} is that close to $\bar{T}_{t-1}$ when event

$$\mathcal{E} = \left\{\forall t \in [n] : \|\bar{T}_t - T_t\| \leq c_1 t^{-\frac{1}{2}}\right\} \tag{6}$$

happens and t is larger that some time horizon t_0. Therefore, the expected regret can be bounded from above as

$$\begin{aligned}&\sum_{t=1}^n \mathbb{E}\left[\ell_t(f(T_{t-1}))\right] \\ &\leq P(\bar{\mathcal{E}})\sum_{t=1}^n \ell_{t,\max} + \sum_{t=1}^{t_0}\ell_{t,\max} + \sum_{t=t_0+1}^n \mathbb{E}\left[\ell_t(f(T_{t-1}))\,|\,\mathcal{E}\right] \\ &\leq 2c_2c_4K^3\sqrt{n} + c_2c_4c_1^2K^2c_3^{-2}\lambda_{\min}^{-2}\sqrt{n} + 2(c_2+55c_1)^p\sqrt{n}\end{aligned}$$

where $\ell_{t,\max}$ is the *maximum loss at time* t. In Lemma 2, we show that the complement of event $\mathcal{E}$, $\bar{\mathcal{E}}$, is unlikely. This gives us an upper bound on $P(\bar{\mathcal{E}})$. In Lemma 3, we determine the time horizon t_0 conditioned on event $\mathcal{E}$. In Lemma 4, we bound $\ell_{t,\max}$. The last step in the above inequality is from combining all these results. ■

Constant	Value
c_1	$K^{\frac{3}{2}}r_1\sqrt{\log n}$
c_2	$2\sqrt{K^3}r_1 + \sqrt{K^3}r_2$
c_3	Theorem 5.1 of Anandkumar *et al.* [2014]
c_4	$\max_\theta \sum_{i=0}^p \|\bar{T}_t - \mathcal{T}(\theta)\|^i \|\bar{T}_{t-1} - \mathcal{T}(\theta)\|^{p-i}$

Table 1: List of constants c_i used in the analysis.

6.3 Technical Lemmas

Lemma 1. *At any time t, $\|\bar{T}_t - \bar{T}_{t-1}\| \leq c_2 t^{-\frac{1}{2}}$.*

Proof. By the definition of $\bar{T}_t$ and the triangle inequality, $\|\bar{T}_t - \bar{T}_{t-1}\| = \frac{1}{t}\|\bar{Y}_{t,t} - \frac{\sum_{z=1}^{t-1}\bar{Y}_{t,z}}{t-1} + \frac{t\sum_{z=1}^{t-1}(\bar{Y}_{t,z} - \bar{Y}_{t-1,z})}{t-1}\| \leq \frac{1}{t}\left(\|\bar{Y}_{t,t}\| + \frac{\sum_{z=1}^{t-1}\|\bar{Y}_{t,z}\|}{t-1} + \frac{t\sum_{z=1}^{t-1}\|\bar{Y}_{t,z} - \bar{Y}_{t-1,z}\|}{t-1}\right) \leq \frac{1}{t}\left(\|\bar{Y}_{t,t}\|_F + \frac{\sum_{z=1}^{t-1}\|\bar{Y}_{t,z}\|_F}{t-1} + \frac{t\sum_{z=1}^{t-1}\|\bar{Y}_{t,z} - \bar{Y}_{t-1,z}\|_F}{t-1}\right) \leq \frac{1}{\sqrt{t}}(\sqrt{K^3}r_1 + \sqrt{K^3}r_1 + \sqrt{K^3}r_2) = \frac{1}{\sqrt{t}}c_2$, where the third inequality is from Assumptions 1 and 2. ■

Lemma 2. *The bad event $\bar{\mathcal{E}} = \left\{\exists t : \|\bar{T}_t - T_t\| > c_1 t^{-\frac{1}{2}}\right\}$ is unlikely, $P(\bar{\mathcal{E}}) \leq 2K^3 n^{-1}$.*

Proof. We have $P(\bar{\mathcal{E}}) \leq P\left(\exists t : \|\bar{T}_t - T_t\|_F > c_1 t^{-\frac{1}{2}}\right) \leq P\left(\exists t, i, j, k : |\bar{T}_t(i,j,k) - T_t(i,j,k)| > c_1 K^{-\frac{3}{2}} t^{-\frac{1}{2}}\right) \leq \sum_{t,i,j,k} P\left(|\bar{T}_t(i,j,k) - T_t(i,j,k)| > c_1 K^{-\frac{3}{2}} t^{-\frac{1}{2}}\right) \leq 2K^3 n^{-1}$, where the third inequality is from union bound and the last inequality is from Azuma's inequality. ■

Lemma 3. *Let the good event $\mathcal{E}$ in* (6) *happen. Then*

$$\|\bar{T}_t - \mathcal{T}(f(T_t))\| \leq 55\, c_1 t^{-\frac{1}{2}}$$

at any time $t \geq t_0 = c_1^2K^2c_3^{-2}\lambda_{\min}^{-2}$.

Proof. When event $\mathcal{E}$ in (6) happens, $\|\bar{T}_t - T_t\| \leq c_1 t^{-\frac{1}{2}}$. Let $\epsilon = c_1 t^{-\frac{1}{2}}$. We have $\epsilon \leq c_3\lambda_{\min}K^{-1}$ for any $t \geq t_0 = \frac{c_1^2K^2}{c_3^2\lambda_{\min}^2}$. Therefore $\|\bar{T}_t - \mathcal{T}(f(T_t))\| \leq 55\, c_1 t^{-\frac{1}{2}}$ for any $t \geq t_0$, from Theorem 5.1 of Anandkumar *et al.* [2014]. ■

Lemma 4. *Let $p \geq 2$. Then $\ell_t(\theta) \leq c_2c_4t^{\frac{1}{2}}$ at any time t.*

Proof. Let $p \geq 2$ and c_4 take the value in Table 1. By definition, $\ell_t(\theta) \leq t(\|\bar{T}_t - \mathcal{T}(\theta)\|^p - \|\bar{T}_{t-1} - \mathcal{T}(\theta)\|^p) \leq tc_4(\|\bar{T}_t - \mathcal{T}(\theta)\| - \|\bar{T}_{t-1} - \mathcal{T}(\theta)\|) \leq tc_4\|\bar{T}_t - \bar{T}_{t-1}\| \leq c_2c_4t^{\frac{1}{2}}$, where the third inequality is from the reverse triangle inequality and the last inequality is from Lemma 1. ■

6.4 Discussion

We derive two $O(\sqrt{n})$ upper bounds up to log factors on the n-step regret of `SpectralLeader`. In the noise-free setting, we bound the regret. In the noisy setting, we bound the expected regret. The choice of $p \geq 3$ in our analysis does not fundamentally alter our problem, in the sense that the regret of any fixed solution but θ^* grows linearly with n.

Assumptions 1 and 2 are satisfied when the whitening matrix in $Y_{t,z}$, W_t, is $\bar{W}_t$. More precisely, Assumption 1 holds

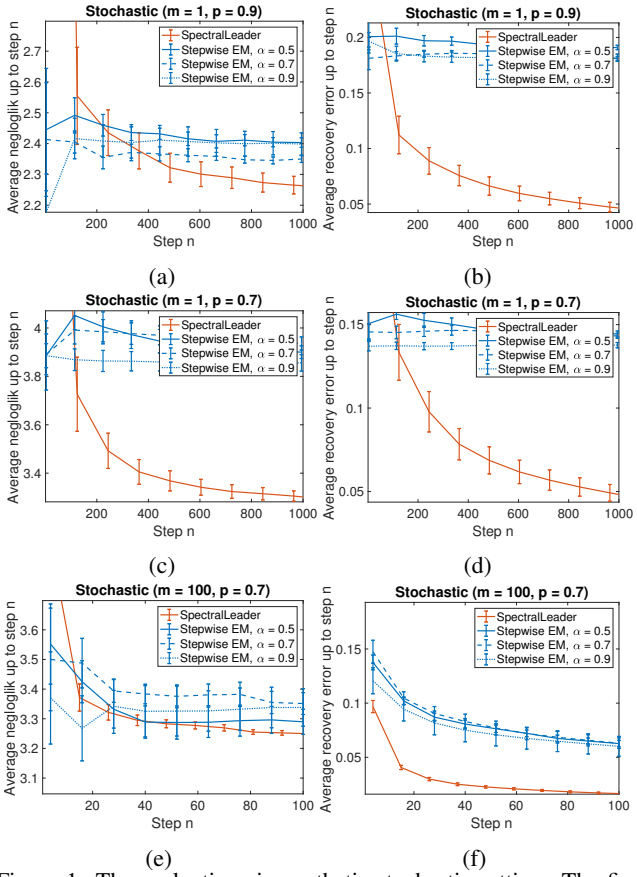

Figure 1: The evaluations in synthetic stochastic setting. The first column shows the results under the metric $\mathcal{L}_n^{(1)}$ and the second column shows the results under the metric $\mathcal{L}_n^{(2)}$.

when $\bar{W}_t$ is invertible, which holds under a mild regularity condition that the learning agent initially observes each topic once. Assumption 2 follows from the Sherman-Morrison formula for matrix inversion, since any $\bar{W}_t$ and $\bar{W}_{t-1}$ are computed by inverting matrices that differ by a rank 1 update.

7 Experiments

In this section, we evaluate `SpectralLeader` and compare it with stepwise EM [Cappé and Moulines, 2009]. We experiment with both stochastic and non-stochastic synthetic problems, as well as with two real-world problems.

Our chosen baseline is stepwise EM [Cappé and Moulines, 2009], an online EM algorithm. We choose this baseline as it outperforms other online EM algorithms, such as incremental EM [Liang and Klein, 2009]. Stepwise EM has two key tuning parameters: the step-size reduction power α and the mini-batch size m [Liang and Klein, 2009; Cappé and Moulines, 2009]. The smaller the α, the faster the old sufficient statistics are forgotten. The mini-batch size m is the number of documents to calculate the sufficient statistics for each update of stepwise EM. In the following experiments, we compared `SpectralLeader` to stepwise EM with varying α and m.

At each time t, we evaluate model θ_{t-1} learned from the first $t-1$ observations, by the stepwise EM or `SpectralLeader`. To have a fair comparison, we report experimental results under two metrics: (i) *average negative predictive log-likelihood up to step* n, $\mathcal{L}_n^{(1)} = \frac{1}{n}\sum_{t=2}^{n} \left(-\log \sum_{i=1}^{K} P_{\theta_{t-1}}(C=i) \prod_{l=1}^{L} P_{\theta_{t-1}}(\mathbf{x}=\mathbf{x}_t^{(l)}|C=i)\right)$, where L is the number of observed words in each document, and (ii) *average recovery error up to step* n, $\mathcal{L}_n^{(2)} = \frac{1}{n}\sum_{t=2}^{n} \|M_2(\theta^*) - M_2(\theta_{t-1})\|_F^2$. Here $M_2(\theta^*)$ and $M_2(\theta_{t-1})$ are the reconstructed second order moments from the optimal model θ^* and the model θ_{t-1}. In synthetic problems, we know θ^*. In real-world problems, we learn θ^* by the spectral method because we have all data in advance. We choose $\mathcal{L}_n^{(2)}$ as a metric instead of $\mathcal{L}_n$ in (3), as $\mathcal{L}_n^{(2)}$ can easily be computed for both `SpectralLeader` and stepwise EM. This metric measures parameter reconstruction error, and therefore is also closely related to our objective in (3). Note that EM in the offline setting minimizes the negative log-likelihood, while `SpectralLeader` in the offline setting minimizes the recovery error of tensors. All reported results are averaged over 10 runs.

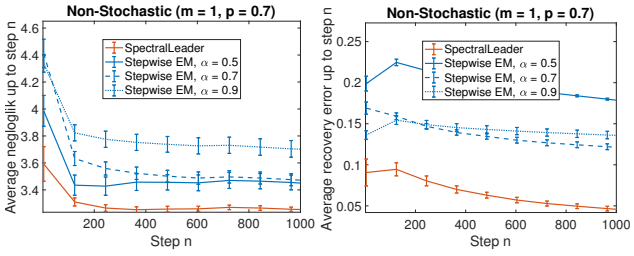

Figure 2: The evaluations in synthetic non-stochastic setting.

7.1 Synthetic Stochastic Setting

We compare `SpectralLeader` with stepwise EM on two synthetic problems in the stochastic setting. In this setting, the topic of the document at all times t is sampled i.i.d. from a fixed distribution. This setting represents a scenario where the sequence of topics is not correlated. The number of distinct topics is $K = 3$, the vocabulary size is $d = 3$, and each document has 3 observed words. In practice, some topics are more popular than others. Therefore, we sample topics as follows. At each time, the topic is randomly sampled from the distribution where $P(C=1) = 0.15$, $P(C=2) = 0.35$, and $P(C=3) = 0.5$. Given the topic, the conditional probability of words is $P(\mathbf{x} = e_i | C = j) = p$ when $i = j$, and $P(\mathbf{x} = e_i | C = j) = \frac{1-p}{2}$ when $i \neq j$. With smaller p, the conditional distribution of words given different topic becomes similar, and the difficulty of distinguishing different topics increases. In Section 7.1 and 7.2, we define the hard problem as the synthetic problem with $p = 0.7$ and the easy problem as the synthetic problem with $p = 0.9$. For $m = 1$, we evaluate on the easy problem and the hard problem. For $m = 100$, we further focus on the hard problem. We show the results before the different methods converge: for $m = 1$, we report results before $n = 1000$, and for $m = 100$ we report both results before $n = 100$.

Our results are reported in Figure 1. We observe three trends. First, under the metric $\mathcal{L}_n^{(1)}$, stepwise EM is very sensitive to its parameters α and m, while `SpectralLeader` is competitive or even better, compared to the stepwise EM with its best α and m. For example, the best α is 0.7 in Figure 1a,

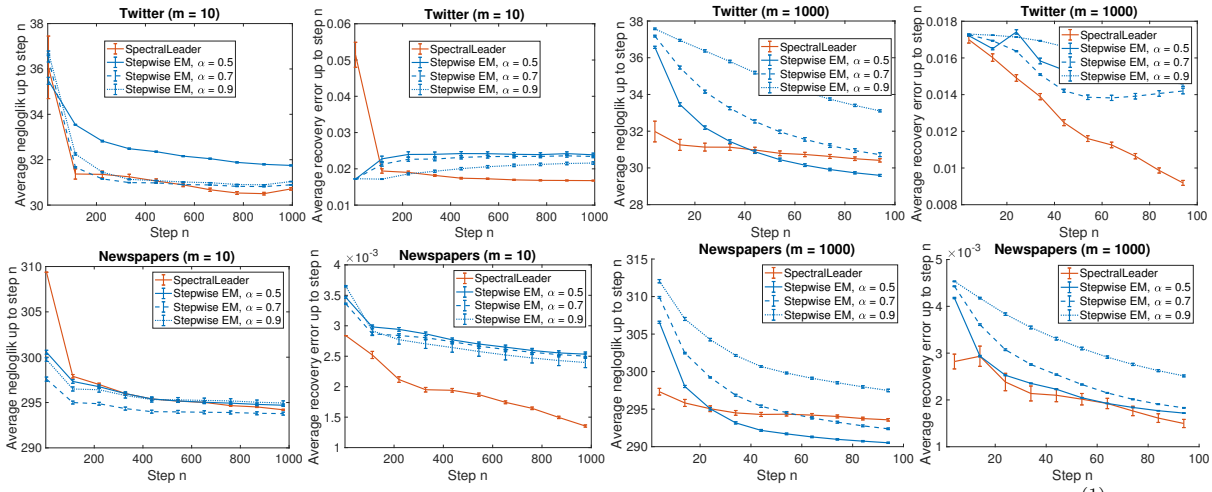

Figure 3: The evaluations on two real-world datasets. The first and third columns show the results under the metric $\mathcal{L}_n^{(1)}$, while the second and fourth columns show the results under the metric $\mathcal{L}_n^{(2)}$.

and the best α is 0.9 in Figure 1c. Even for the same problem with different m, the best α is different: the best α is 0.9 in Figure 1c, while the best α is 0.5 in Figure 1e. In all cases, `SpectralLeader` performs the best. Second, similar to [Liang and Klein, 2009], stepwise EM improves when the mini-batch size increases to $m = 100$. But `SpectralLeader` still performs better compared to stepwise EM with its best α. Third, `SpectralLeader` performs much better than stepwise EM under the metric $\mathcal{L}_n^{(2)}$. These results indicate that a careful grid search of α and m is usually needed to optimize stepwise EM. In contrast, `SpectralLeader` is very competitive without any parameter tuning. Note that grid search in the online setting is nearly impossible, since the future data are unknown in advance.

7.2 Synthetic Non-Stochastic Setting

Now we evaluate all algorithms on the hard problem in non-stochastic setting. This problem is the same as the hard problem in the stochastic setting, except that topics of the documents are strongly correlated over time. In each batch of 100 steps, sequentially we have 15 documents from topic 1, 35 documents from topic 2, and 50 documents from topic 3.

Our results are reported in Figure 2. we observe two major trends. First, for stepwise EM, the α leading to lowest negative log-likelihood is 0.5. This result matches well the fact that the smaller the α, the faster the old sufficient statistics are forgotten, and the faster stepwise EM adapts to the non-stochastic setting. Second, in terms of adaptation to correlated topics, `SpectralLeader` is even better than stepwise EM with $\alpha = 0.5$. Note that $\alpha = 0.5$ is the smallest valid value of α for stepwise EM [Liang and Klein, 2009].

7.3 Real World Problems

In this section, we compare `SpectralLeader` to stepwise EM on real world problems. We evaluate on Newspapers data collected over multiple years and Twitter data collected during the 2016 United States elections.[1] They provide sequences of documents with timestamps and the distributions of topics change over time. After preprocessing, we retain 500 most frequent words in the vocabulary. We set $K = 5$. We evaluate all algorithms on 100K documents. We compare `SpectralLeader` to stepwise EM with multiple α, and mini-batch sizes $m = 10$ and $m = 1000$. We show the results before the different methods converge: for $m = 10$, we report results before $n = 1000$, and for $m = 1000$ we report results before $n = 100$. To handle the large scale streaming data (*e.g.*, 5M words in Newspapers data), we introduce reservoir sampling, and set the window size of reservoir as 10,000.

Our results are reported in Figure 3. We observe four major trends. First, under the metric $\mathcal{L}_n^{(2)}$, `SpectralLeader` performs better than stepwise EM. Second, under the metric $\mathcal{L}_n^{(1)}$ on both datasets, the optimal α for stepwise EM are different, for $m = 10$ versus $m = 1000$. Third, when $m = 10$, under the metric $\mathcal{L}_n^{(1)}$, `SpectralLeader` performs competitive with or better than stepwise EM with its best α. Fourth, when $m = 1000$, under the metric $\mathcal{L}_n^{(1)}$, `SpectralLeader` is not as good as the stepwise EM with its best α. However, directly using `SpectralLeader` without the effort of tuning any parameters can still provide good performance. These results suggest that, even when the mini-batch size is large, `SpectralLeader` is still very useful under the log-likelihood metric: in practice we can quickly achieve reasonable results by `SpectralLeader` without any parameter tuning.

8 Conclusions

We propose `SpectralLeader`, a novel online learning algorithm for latent variable models. With an instance of bag-of-words model, we define a novel per-step loss function, prove that `SpectralLeader` converges to a global optimum, and derive a $O(\sqrt{n})$ cumulative regret bound up to log factors in n for `SpectralLeader`. Our experimental results show that, in most cases `SpectralLeader` performs similar to or better than an optimally-tuned online EM. In future work, we would extend our method to learn more complicated latent-variable models, such as HMMs and LDA [Anandkumar *et al.*, 2014].

[1] Please see `https://www.kaggle.com/snapcrack/all-the-news` and `https://www.kaggle.com/kinguistics/election-day-tweets`.

References

[Anandkumar *et al.*, 2012a] Anima Anandkumar, Dean P Foster, Daniel J Hsu, Sham M Kakade, and Yi-Kai Liu. A spectral algorithm for latent dirichlet allocation. In *Advances in Neural Information Processing Systems*, pages 917–925, 2012.

[Anandkumar *et al.*, 2012b] Animashree Anandkumar, Daniel Hsu, and Sham M Kakade. A method of moments for mixture models and hidden markov models. In *Conference on Learning Theory*, pages 33–1, 2012.

[Anandkumar *et al.*, 2014] Animashree Anandkumar, Rong Ge, Daniel J Hsu, Sham M Kakade, and Matus Telgarsky. Tensor decompositions for learning latent variable models. *Journal of Machine Learning Research*, 15(1):2773–2832, 2014.

[Bishop, 2006] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[Cappé and Moulines, 2009] Olivier Cappé and Eric Moulines. On-line expectation–maximization algorithm for latent data models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(3):593–613, 2009.

[Chaganty and Liang, 2013] Arun T Chaganty and Percy Liang. Spectral experts for estimating mixtures of linear regressions. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 1040–1048, 2013.

[Ge *et al.*, 2015] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on Learning Theory*, pages 797–842, 2015.

[Hoffman *et al.*, 2010] Matthew Hoffman, Francis R Bach, and David M Blei. Online learning for latent dirichlet allocation. In *advances in neural information processing systems*, pages 856–864, 2010.

[Huang *et al.*, 2015] Furong Huang, UN Niranjan, Mohammad Umar Hakeem, and Animashree Anandkumar. Online tensor methods for learning latent variable models. *Journal of Machine Learning Research*, 16:2797–2835, 2015.

[Kar *et al.*, 2014] Purushottam Kar, Harikrishna Narasimhan, and Prateek Jain. Online and stochastic gradient methods for non-decomposable loss functions. In *Advances in Neural Information Processing Systems*, pages 694–702, 2014.

[Liang and Klein, 2009] Percy Liang and Dan Klein. Online em for unsupervised models. In *Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics*, pages 611–619. Association for Computational Linguistics, 2009.

[Neal and Hinton, 1998] Radford M Neal and Geoffrey E Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. In *Learning in graphical models*, pages 355–368. Springer, 1998.

[Nowozin and Lampert, 2011] Sebastian Nowozin and Christoph H Lampert. Structured learning and prediction in computer vision. *Foundations and Trends® in Computer Graphics and Vision*, 6(3–4):185–365, 2011.

[Rabiner, 1989] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[Shaban *et al.*, 2015] Amirreza Shaban, Mehrdad Farajtabar, Bo Xie, Le Song, and Byron Boots. Learning latent variable models by improving spectral solutions with exterior point method. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pages 792–801. AUAI Press, 2015.

[Tung and Smola, 2014] Hsiao-Yu Tung and Alexander J Smola. Spectral methods for indian buffet process inference. In *Advances in Neural Information Processing Systems*, pages 1484–1492, 2014.

[Tung *et al.*, 2017] Hsiao-Yu Fish Tung, Chao-Yuan Wu, Manzil Zaheer, and Alexander J Smola. Spectral methods for nonparametric models. *arXiv preprint arXiv:1704.00003*, 2017.

[Wallach, 2006] Hanna M Wallach. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pages 977–984. ACM, 2006.