Provable Algorithms for Inference in Topic Models

Sanjeev Arora ARORA@CS.PRINCETON.EDU
Department of Computer Science, Princeton University

Rong Ge RONGGE@CS.DUKE.EDU
Computer Science Department, Duke Unversity

Frederic Koehler FKOEHLER@PRINCETON.EDU
Department of Mathematics, Princeton University

Tengyu Ma TENGYU@CS.PRINCETON.EDU
Department of Computer Science, Princeton University

Ankur Moitra MOITRA@MIT.EDU
Department of Mathematics and CSAIL, Massachusetts Institute of Technology

Abstract

Recently, there has been considerable progress on designing algorithms with provable guarantees — typically using linear algebraic methods — for parameter learning in latent variable models. But designing provable algorithms for inference has proven to be more challenging. Here we take a first step towards provable inference in topic models. We leverage a property of topic models that enables us to construct simple linear estimators for the unknown topic proportions that have small variance, and consequently can work with short documents. Our estimators also correspond to finding an estimate around which the posterior is well-concentrated. We show lower bounds that for shorter documents it can be information theoretically impossible to find the hidden topics. Finally, we give empirical results that demonstrate that our algorithm works on realistic topic models. It yields good solutions on synthetic data and runs in time comparable to a *single* iteration of Gibbs sampling.

1. Introduction

Generative models of data are ubiquitous in unsupervised learning, and lead to two types of computational problems: In *parameter learning*, the goal is find the parameters of the model that best fits a given collection of data. In *inference*, the goal is to learn the values of latent variables for a specific datapoint. A wide range of approaches are empirically effective for both tasks, including *Gibbs sampling* and *variational inference*. However, for the most part we lack strong provable guarantees — on running time, or quality of solution — for these approaches.

Recently, there has been considerable progress on designing new algorithms for parameter learning with such provable guarantees. Since the usual maximum likelihood estimator is often NP-hard to compute even in simple models, these new algorithms use alternative estimators based on the method of moments and linear algebra. Their analysis usually involves making a structural assumption about the parameters of the problem, which can often be justified in applications. Some highlights include algorithms for *topic modeling* (Arora et al., 2013b; Anandkumar et al., 2012), *learning mixture models* (Moitra & Valiant, 2010; Hsu & Kakade, 2013; Ge et al., 2015), community detection (Anandkumar et al., 2014) and (special cases of) *deep learning* (Arora et al., 2014; Janzamin et al., 2015).

But there has been comparatively much less progress on designing algorithms with provable guarantees for inference. The current paper takes a first step in this direction, in context of topic models. Our algorithms leverage a property of topic models (Definition 3.1) that turns out to hold in many datasets — the existence of a good approximate inverse matrix. We also give empirical results that demonstrate that our algorithm works on realistic topic models. On synthetic data, its error is competitive with state-of-the-art approaches (which have no such provable guarantees).

Proceedings of the 33^{rd} International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48.

It obtains somewhat weaker results on real data.

1.1. Setup and Overview

Here we describe topic modeling, and why inference appears more difficult than parameter learning. In topic modeling, each document is represented as a *bag of words* where we ignore the order in which words occur. The model assumes there is a fixed set of k *topics*, each of which is a distribution on words. Thus the ith topic is a vector $A_i \in \mathbb{R}^D$ (where D is the number of words in the language) whose coordinates are nonnegative and sum to 1. Each document is generated by first picking its topic proportions from some distribution; say x_i is the proportion of topic i, so that $\sum_i x_i = 1$. The model assumes a distribution on x that favors sparse or approximately sparse vectors; a popular choice is the Dirichlet distribution (Blei et al., 2003). Then the document $\{w_1, w_2, \ldots, w_n\}$ is generated by drawing n words independently from the distribution $A \cdot x$ where A is the matrix whose columns are the topics. It is important to note that the document size n can be quite small (e.g., n may be 400, and D may be $50,000$) so the empirical distribution of words in a document is in general a very *inaccurate* approximation to Ax. With some abuse of notation we also think of y as a vector in $\mathbb{R}^D$, whose jth coordinate is the number of occurences of word j in the document.

Parameter learning involves recovering the best A for a corpus of documents; this can be seen as a the latent structure in the corpus. Recent (provable) algorithms for this problem (Anandkumar et al., 2012; Arora et al., 2013b) use the *method of moments*, leveraging the fact that some form of averaging over the corpus yields a linear algebraic problem for recovering A. For example the *word-word* cooccurence matrix (whose i, j entry is the probability that words i, j co-occur in a document) is given by

$$E_x[Axx^TA^T] = AZA^T$$

where Z is the 2nd moment matrix of the prior distribution on x. It is possible to recover A from this expression, under natural conditions like *separability* (Arora et al., 2013b). Alternatively, one can use a co-occurrence tensor and recover A under weaker assumptions (Anandkumar et al., 2012).

In the inference problem, we know the topic matrix A and are given a single document y generated using this matrix. The goal is to find the posterior distribution $x|y$. This can be seen as labeling or categorizing this document, which is important in applications. Inference is reminiscent of classical regression problems where the goal is to find x given $y = Ax +$ noise vector. The key difference here is the nature of noise —for each word coordinate j is 1 with probability $(Ax)_j$, and 0 otherwise— which means that the noise on a coordinate-by-coordinate basis can be much larger than the signal. In particular the vector $y \in \mathbb{R}^D$ is very sparse even though Ax is dense. This problem can be seen as an analog of sparse linear regression when the target (regression) vector x has nonnegative coordinate and $\sum_i x_i = 1$. (This is distinct from usual ℓ_1-regression where regression vector is in ℓ_2 even though the loss function is ℓ_1.) The difficulty here, in addition to the issue of high coordinate-wise error already mentioned, is that the usual sparsity-enforcing ℓ_1-regularization buys nothing since the solution needs to exactly satisfy $\|x\|_1 = 1$.

Inference seems more difficult than parameter learning because averaging over many documents is no longer an option. Furthermore, the solution x is not unique in general, and in some cases the posterior distribution on x is not well concentrated around any particular value. (In practice Gibbs Sampling can be used to sample from the posterior (Griffiths & Steyvers, 2004; Yao et al., 2009), but as mentioned, a rigorous analysis has proved difficult. The inference is actually NP-hard.) We will view inference as a problem of recovering some *ground truth* x^* that was used to generate the document, and we show that with probability close to 1 our estimate $\hat{x}$ is close to x^* in ℓ_1 norm.

Bayesian vs Frequentist Views. So far we have not differentiated between Bayesian and frequentist approaches to frame the inference problem, and now we show that the two are closely related here. The above description is frequentist, assuming an unknown "ground truth"vector x^* of topic proportions (which is r-sparse for some small r) was used to generate a document y, using a distribution $y|x^*$. Let $\mathcal{E}_{x^*}$ be the event that our algorithm recovers a vector $\hat{x}$ such that $\|\hat{x} - x^*\|_1 \leq \epsilon$. For our algorithm $\Pr_{y|x^*}[\mathcal{E}_{x^*}] \geq 1 - \delta^2$ for some $\delta > 0$. By contrast, in the Bayesian view, one assumes a prior distribution on x^* and seeks to output a sample from the conditional distribution $x^*|y$. Now we show that the success of our frequentist algorithm implies that the posterior $x^*|y$ must also be concentrated, and place most probability mass on set of x such that $\|x - \hat{x}\|_1 \leq \epsilon$. By law of total expectation, we have $\Pr_{x^*,y}[\mathcal{E}_{x^*}] = \Pr_{x^*}\left[\Pr_{y|x^*}[\mathcal{E}_{x^*}|x^*]\right] \geq 1-\delta^2$. Switching the order of expectation, we obtain

$$\Pr_y\left[\Pr_{x^*|y}[\mathcal{E}_{x^*} \mid y]\right] \geq 1 - \delta^2\,.$$

Then it follows by Markov argument that

$$\Pr_y\left[\Pr_{x^*|y}[\mathcal{E}_{x^*} \mid y] \geq 1 - \delta\right] \geq 1 - \delta\,.$$

Note that the inner probability is over the posterior distribution $p_{x^*|y}$. But the event $\mathcal{E}_{x^*}$ only depends on the output $\hat{x}$ of the algorithm given y. Thus the probability is at least $1 - \delta$ over choice of y, that $1 - \delta$ of the probability mass of $x^*|y$ is concentrated in the ℓ_1 ball of radius ϵ around the algorithm's answer $\hat{x}$.

From now on the goal of our algorithm is to *recover* x^* *given* y, and we identify conditions under which the event has probability close to 1.

Minimum Variance Estimators (with Bias). Having set up the problem as above, next we consider how to *recover* an approximation to x^* given a document y generated with topic proportions x^*.

Since A has orders of magnitude more rows than columns, it has many left inverses to choose from. If we find any matrix B where BA is equal to the identity matrix, then By is an unbiased estimate for x^*. However this estimate has high variance if B has large entries, necessitating working with only very large documents. Motivated by applications to collaborative filtering, Kleinberg & Sandler (2008) introduce the notion of the ℓ_1 condition number (see Definition 2.1) of A, which allows them to construct a left inverse B with a much smaller maximum entry. We introduce a weaker notion of condition number called the ℓ_∞-*to*-ℓ_1 *condition number*, which leverages the observation that even if BA is *close* to the identity matrix it still yields a good linear estimator for x^*. We call B an approximate inverse of A. Moreover it has the benefit that the condition number as well as the approximate left inverse B with minimum variance can also be computed in polynomial time using a linear program (Proposition 3.2)!

In our experiments, we compute the exact condition number of word-topic matrices that were found using standard topic modeling algorithms on real-life corpora. (By contrast, we do not know the ℓ_1 condition number of these matrices.) In all of the examples, we found that the condition number is at most a small constant, which allows us to compute good approximate left inverses to the topic matrix A to enable us to estimate x^* even with relatively short documents.

Main results. Our main result (Theorem 4.1) shows that when the condition number is small, it is possible to estimate x^* using a combination of thresholding and a left inverse B of minimum variance. Our overall algorithm runs efficiently and requires time $O(nk)$ and $\widetilde{O}(r^2)$ samples to achieve $o(1)$ error in ℓ_1 norm and $o(1/r)$ error in ℓ_∞ norm, where r is the number of topics represented in the document. Note that we do not need to assume a particular model (e.g. uniform random) for the r topics, the algorithm works even when the topics may be correlated with each other.

As an intermediate step, we are able to recover the support of x^* when each of its non-zero coordinates is suitably bounded away from zero. We complement this result by showing that maximizing the log-likelihood function over the recovered support can further reduce the estimation error (measured in the ℓ_1-norm) to $\widetilde{O}(\sqrt{r/n})$ (see Section 5). The experiments show that it indeed yields estimates for x^* with smaller error (see Section 7).

Finally we show that in order to recover support of x^*, it is necessary to observe $\Omega(r^2)$ words, even if A is perfectly conditioned and x^* is promised to have all non-zero coordinates larger than $\Omega(1/r)$ (see Lemma 6.2 for a family of such perfectly conditioned but hard instances of A, and Lemma 6.3 for hard instance of x^*).

Thus to sum up, our overall approach involves simple linear algebraic primitives followed by convex programming. For a topic model with k topics, the sample complexity of our algorithms depend on $\log k$ instead of k. This is important in practice as k is often at least 100. The accuracy on synthetic data is good for sparse x, though not quite as good as Gibbs sampling. However, if we forgo the convex programming step we can compute a reasonable estimate for x from a single matrix vector multiplication plus thresholding, which is an order of magnitude faster than finding an estimate of the same quality via Gibbs sampling. And of course, our approach comes with a performance guarantee.

2. Notations and Setup

In addition to the description of topic model in Section 1.1, we introduce the following notations. We use $\mathcal{S}_k = \{z \in \mathbb{R}^k_{\geq 0} : |z|_1 = 1\}$ to denote the k-dimensional probability simplex. We assume that the true topic proportion vector $x^* \in \mathcal{S}_k$ is r-sparse throughout the paper. Sometimes we also abuse notations and use y as a D dimensional vector instead of a set, in this case y_i is the number of times word i appears in the document. We will use $a_i^\top$ to denote the i-th row of A. We will use $\text{cat}(p)$ to denoted the categorical distribution defined by probability vector p. Euclidean norm, ℓ_1, ℓ_∞ norm of a vector is denoted by $\|\cdot\|$, $\|\cdot\|_1$ and $\|\cdot\|_\infty$ respectively.

Condition Numbers of Matrices Condition number of a matrix usually represents the ratio of the largest and smallest singular values. However, this concept is tied to ℓ_2 norm, and for probability distributions the most natural norms are ℓ_1 and ℓ_∞.

Next we define various matrix norms that we will utilize. Let $|A|_\infty = \max_{i,j} |A_{ij}|$ denotes the maximum absolute value of the entries of the matrix A, and $|A|_1 = \sum_{i,j} |A_{ij}|$ denotes the sum of the absolute value of the entries of the matrix A. Let Id_k denotes the identity matrix of dimension k. For a matrix, let $\|\cdot\|$ denote the spectral norm, and $\|\cdot\|_Q$ denote the norm defined by $\|x\|_Q = \sqrt{x^\top Q x}$ where Q is a positive semidefinite matrix. We will use this norm particularly with Q being fisher information matrix.

We will also work with various notions of condition num-

ber, that we will use in our guarantees.

Definition 2.1 (ℓ_1-condition number)**.** For a nonnegative matrix A, define its ℓ_1-condition number $\kappa(A)$ to be the minimum κ such that for any $x \in \mathbb{R}^k$,

$$\|Ax\|_1 \geq \|x\|_1/\kappa \tag{1}$$

This condition number was introduced by Kleinberg & Sandler (2008) in analyzing various algorithms for collaborative filtering. We will use a weaker (i.e. smaller) notion of condition number. Empirically, it seems that most of the word-topic matrices that we have encountered have a reasonably small ℓ_1-condition number, and have an even smaller $\ell_\infty \to \ell_1$-condition number.

Definition 2.2 ($\ell_\infty \to \ell_1$-condition number)**.** Let $\lambda(A)$ be the minimum number λ such that for any $x \in \mathbb{R}^k$,

$$\|Ax\|_1 \geq \|x\|_\infty/\lambda \tag{2}$$

Remark 1. Based on the relationship between ℓ_1 and ℓ_∞ norm, we have that $\lambda(A) \leq \kappa(A) \leq k\lambda(A)$. In Section 6 we give an example where the $\ell_1 \to \ell_1$ condition number is significantly worse: $\kappa(A) \geq \Omega(\sqrt{k})\lambda(A)$.

3. δ-Biased Minimum Variance Estimators

Let $y \in \mathbb{R}^D$ be the document vector whose i-th entry y_i is the number of times word i appears. We try to estimate the true topic vector x^* by left multiplying y with some matrix B. Intuitively, $\mathbb{E}[By] = BAx^*$, so we want BA to be close to the identity matrix. On the other hand, when we apply B to the document vector, each word will select a column of B, and its variance on *any* entry is bounded by the maximum entry in B. Therefore we would like to optimize over two things: first, we want BA to be close to identity; second, we want the matrix B to have small $|B|_\infty$. This inspires the following linear program:

Definition 3.1. For $A \in \mathbb{R}^{D\times k}$ and $\delta \geq 0$, define $\lambda_\delta(A)$ to be the solution of the following convex program:

$$\begin{aligned} \lambda_\delta(A) = \min \quad & |B|_\infty \\ \text{s.t.} \quad & |BA - \mathrm{Id}_k|_\infty \leq \delta \\ & B \in \mathbb{R}^{k\times D} \end{aligned} \tag{3}$$

We will refer to the minimizer B of the above convex program as the δ-biased minimum variance inverse for A. The solution to the above convex program will help minimize our sample complexity both theoretically and empirically.

Allowing a nonzero δ can potentially reduce the variance of the estimator while introducing a small bias. Such bias-variance trade-off has been studied in other settings (Moitra & Saks, 2013; Javanmard & Montanari, 2014).

What is the optimal $|B|_\infty$? To answer this question we get the dual of the LP 3 (with variable $Q \in \mathbb{R}^{k\times k}$),

$$\begin{aligned} \text{maximize} \quad & \mathrm{tr}(Q) - \delta|Q|_1 \\ \text{s.t.} \quad & |AQ|_1 \leq 1 \end{aligned} \tag{4}$$

We can further show that (4) is equivalent to the following (non-convex) program with vector variables $x \in \mathbb{R}^k$ (see Appendix B for the proof):

$$\begin{aligned} \text{maximize} \quad & \|x\|_\infty - \delta\|x\|_1 \\ \text{s.t.} \quad & \|Ax\|_1 \leq 1 \end{aligned}$$

Note that this is very closely related to the condition number λ in Definition 2.2. In particular, the optimal value is exactly $\lambda(A)$ when $\delta = 0$! When $\delta > 0$ this can be viewed as a relaxation of the $\ell_\infty \to \ell_1$ condition number. This is summarized in the following Proposition whose proof is deferred to appendix.

Proposition 3.2. *For any $\delta \geq 0$, we have that $\lambda_\delta(A) \leq \lambda_0(A) = \lambda(A) \leq \kappa(A)$.*

4. Recovery Guarantees in the ℓ_1-Norm

In this section we show how to estimate the topic proportion vector using a δ-biased minimum variance inverse B of word-topic matrix A (Definition 3.1). For a small δ (that is $\ll 1/r$), given a solution B of program (3) with entries of absolute value at most $\lambda_\delta(A)$, the following Thresholded Linear Inverse estimator (Algorithm 1) is guaranteed to be close to the true x^* in both ℓ_1 and ℓ_∞ norm.

Algorithm 1 Thresholded Linear Inverse Algorithm (TLI)

Input: Document y with n words, and δ-biased inverse matrix B of matrix A.
Output: Topic vector estimator x.

1. Compute $\hat{x} = \frac{1}{n}By$.
2. Let $\tau = 2\lambda_\delta(A)\sqrt{\log k/n} + \delta$. For all $i \in [k]$, , if $\hat{x}_i < \tau$, set $x_i = 0$, otherwise set $x_i = \hat{x}_i$.

Theorem 4.1. *Suppose document y is generated from r-sparse topic vector x^*. For any $\epsilon > 4\delta r$, given $n = \Omega(\lambda_\delta(A)^2 r^2 \log k/\epsilon^2)$ samples, with high probability Algorithm 1 returns a vector that has ℓ_1-distance at most ϵ with x^*.*

Our first step is to bound the variance of the partial estimator $\hat{x}$ before thresholding. Our bound will utilize the maximum entry in B, which is why we tried to find B that minimizes this quantity in the first place. In particular we can show:

Lemma 4.2. *With probability at least $1 - 1/k^2$, for every i we have $|\hat{x}_i - x_i^*| \leq \delta + 2\lambda_\delta(A)\sqrt{(\log k)/n}$.*

Proof of Lemma 4.2. By definition, $\hat{x}_i = \frac{1}{n}\sum_{j=1}^{n}(B\mathbf{1}_{w_j})_i$ where $\mathbf{1}_{w_j}$ is the indicator vector for the jth word in the document. By summing over words in the document as opposed to words in the vocabulary, we have written $\hat{x}_j$ as a sum of independent random variables, and we will use Bernstein's inequality to show that it is concentrated around its mean. This is straightforward, but the key is the way we have chosen B ensures that the estimator is at most δ-based. To elaborate, we can compute

$$\mathbb{E}[\hat{x}_i] = (BAx^*)_i = \sum_{j=1}^{k}(BA)_{i,j}x^*_j$$
$$= x^*_j + \sum_{j=1}^{k}((BA)_{i,j} - \mathbf{1}_{i=j})x^*_j\,,$$

where $\mathbf{1}_{i=j} = 1$ if $i = j$ and $\mathbf{1}_{i=j} = 0$ otherwise. Now by construction we have that for all i and j, $|(BA)_{i,j} - \mathbf{1}_{i=j}| \leq \delta$. Hence, $|\sum_{j=1}^{k}((BA)_{i,j} - \mathbf{1}_{i=j})x^*_j| \leq \delta\sum_{j=1}^{k}x^*_j = \delta$. Therefore we conclude that $|\mathbb{E}[\hat{x}_i] - x^*_i| \leq \delta$ which shows that our estimator has bias at most δ.

Now we can appeal to standard concentration arguments. Recall that $\hat{x}_i$ is a sum of independent random variables $\hat{x}_i = \frac{1}{n}\sum_{j=1}^{n}(B\mathbf{1}_{w_j})_i$, and each summand here is bounded by $\max(B\mathbf{1}_{w_j})_i \leq \lambda_\delta(A)$. We apply Hoeffding's inequality and obtain that with probability at least $1-1/k^2$, $|\hat{x}_i - \mathbb{E}[\hat{x}_i]| \leq 2\lambda_\delta(A)\sqrt{(\log k)/n}$ and this completes the proof of the lemma. □

Lemma above shows that the vector $\hat{x}$ is close to the true x^* in infinity norm. As a corollary, we know the algorithm finds the correct support if x^* does not have very small entries

Corollary 4.3. *With high probability, x output by Algorithm 1 satisfies that for every $i \in [k]$, if $x^*_i = 0$ then $x_i = 0$, and if $x^*_i \geq 4\lambda_\delta(A)\sqrt{(\log k)/n} + 2\delta$ then $x_i > 0$. In particular, if all the nonzero entries of x^* are at least ϵ/r for some $\epsilon > 4\delta r$, the algorithm finds the correct support with $O(\lambda_\delta(A)^2 r^2 \log k/\epsilon^2)$ samples.*

Using the corollary above we can then prove Theorem 4.1 (see supplementary material). The key intuition is x can only incur error on non-zero coordinates of x^*, and a fixed amount of error on non-zero coordinates of x^*.

5. Rate of MLE estimator

In this section, we show that given the correct support R of x^*, we can optimize the log-likelihood function over the variables in R and obtain a finer solution with smaller ℓ_1 error. We make the following two assumptions: first, that the non-zero coordinates of x^* are bounded away from zero; second, that the word-topic matrix has small restricted $\ell_1 \to \ell_1$ condition number.

Assumption 5.1. *We assume that $x^* \in \mathcal{S}_k$ satisfies that $R = \mathrm{supp}(x^*)$ is of size at most r and $x^*_i \geq \tau/r$ for any $i \in R$.*

Assumption 5.2 (restricted $\ell_1 \to \ell_1$ condition number)**.** *We assume that word-topic matrix A satisfies that for any r-sparse vector $v \in \mathbb{R}^d$,*

$$\|Av\|_1 \geq \|v\|_1/\bar{\kappa}\,.$$

We note that by definition $\bar{\kappa} \leq \kappa(A)$. Moreover, the restricted $\ell_1 \to \ell_1$ condition number can be viewed as ℓ_1 analog of the restricted isometry property (Candes & Tao, 2005) or restricted eigenvalue conditions (Bickel et al., 2009; Meinshausen & Yu, 2009) associated to ℓ_2 norm. This type of assumption is particularly useful (and somewhat necessary) for the estimation problem.

We will restricted our attention to support R throughout this section. Let $\hat{a}_w \in \mathbb{R}^r$ be the restriction of a_w to the support R, and $\hat{A}$ be the word-topic matrix restricted to columns indexed by R. Let $f(x)$ be the log-likelihood function restricted to the support R. That is, for $x \in \mathbb{R}^r$,

$$f(x) = \log \Pr[y \mid x] = \sum_{w \in y}\log(\langle \hat{a}_w, x\rangle),\,. \qquad (5)$$

The main theorem in this section below shows that when $n = \Omega(r^2)$, the maximum likelihood estimator (MLE) restricted to the support R has ℓ_1 rate $\widetilde{O}(\bar{\kappa}\sqrt{r/n})$. Moreover, the error on predicting Ax^* is $\widetilde{O}(\sqrt{r/n})$ which doesn't depend on the condition number.

Theorem 5.3. *Under assumption 5.1 and 5.2, suppose $n \geq c\bar{\kappa}^2 r^2 \log k/\tau^2$ for a sufficiently large constant c. Let x_{MLE} be the maximizer of the log-likelihood function $f(x)$ restricted to support R. Then with high probability x_{MLE} satisfies that $\|Ax_{\mathrm{MLE}} - Ax^*\|_1 \leq \widetilde{O}\left(\sqrt{\frac{r}{n}}\right)$, and $\|x_{\mathrm{MLE}} - x^*\|_1 \leq \widetilde{O}\left(\bar{\kappa}\sqrt{\frac{r}{n}}\right)$.*

Asymptotically, we know that the error vector $x_{\mathrm{MLE}} - x^*$ converges to standard normal with covariance matrix Q being the Fisher information matrix (see Equation (7)). This means that $Q^{-1/2}(x_{\mathrm{MLE}} - x^*)$ is bounded in ℓ_2 norm. Therefore the keys towards proving Theorem 5.3 consists of a) converting the above to a non-asymptotic bound with careful concentration inequality b) understanding how $Q^{-1/2}$ converts ℓ_1 space to ℓ_2 space so that an ℓ_1 norm of the error can be obtained.

We give intuitions for the proofs here, which mostly follows from the classical asymptotic normality of Maximum Likelihood Estimator, and our main contribution here is to give a finite sample bound using concentration inequalities.

First we consider the gradients and Hessians of the likeli-

hood function.

$$\nabla f(x) = \sum_{w\in y} \frac{\hat{a}_w}{\langle \hat{a}_w, x\rangle},\ \nabla^2 f(x) = -\sum_{w\in y} \frac{\hat{a}_w \hat{a}_w^\top}{\langle \hat{a}_w, x\rangle^2}\,. \quad (6)$$

Let Q be the Fisher information matrix as defined below,

$$Q = \mathbb{E}\left[\frac{\hat{a}_w \hat{a}_w^\top}{\langle \hat{a}_w, x\rangle^2}\right] = \sum_{i\in[D]} \langle \hat{a}_i, x^*\rangle \frac{\hat{a}_i \hat{a}_i^\top}{\langle \hat{a}_i, x^*\rangle^2}\,. \quad (7)$$

Note that we have $\mathbb{E}[\nabla^2 f(x^*)] = -nQ$. When n is sufficiently large and x_{MLE} is sufficiently close to x^*, we have,

$$-\nabla f(x^*) = \nabla f(x_{\text{MLE}}) - \nabla f(x^*) \approx \nabla^2 f(x^*)(x_{\text{MLE}} - x^*)\,.$$

Therefore, it follows that

$$x^* - x_{\text{MLE}} \approx \nabla^2 f(x^*)^{-1} \nabla f(x^*)\,.$$

It can be shown that the covariance of the gradient is $\mathbb{E}[\nabla f(x^*)\nabla f(x^*)^\top] = nQ$. Therefore when $\nabla^2 f(x^*)^{-1}$ is sufficiently close to its expectation nQ, the covariance of the error $x^* - x_{\text{MLE}}$ is approximately equal to $\frac{1}{n}Q^{-1}$.

Towards establishing a non-asymptotic result, we bound from above $\nabla f(x^*)$ and lower from below $\nabla^2 f(x)$ in a proper Euclidean norm – the norm defined by the Fisher information matrix Q in the following three lemmas. Lemma 5.4 below controls the ℓ_2 and ℓ_∞ norm of $Q^{-1/2}\nabla f(x^*)$ (which is supposed to be spherical Gaussian with covariance matrix $\sqrt{n}\mathrm{Id}_r$ asymptotically).

Lemma 5.4. *Under assumption 5.1 and 5.2, suppose $n \geq c\bar{\kappa}^2 r^2/\tau^2 \cdot \log k$ for a sufficiently large constant c. Then, with high probability we have $\|Q^{-1/2}(\nabla f(x^*) - n\mathbf{1}_r)\| \leq \widetilde{O}(\sqrt{nr})$, and $\|Q^{-1/2}(\nabla f(x^*) - n\mathbf{1}_r)\|_\infty \leq \widetilde{O}(\sqrt{n})$.*

Lemma 5.5 relates $-\nabla^2 f(x)$ with the Fisher information matrix Q spectrally around a neighborhood of x^* which MLE will be proved to fall in. We essentially show that $-\nabla f(x^*)$ concentrates around its expectation nQ, and moreover we can effectively approximate the Hessian $-\nabla^2 f(x)$ around a neighborhood of x^* by nQ as well.

Lemma 5.5. *Under assumption 5.1 and 5.2, suppose $n = c_0\bar{\kappa}^2 r^2 \log k/\tau^2$ for sufficiently large constant c_0. Then, with high probability over the randomness of y, it holds that for all x such that $x \leq Cx^*$, $-\nabla^2 f(x) \succeq \frac{n}{2C^2}\cdot Q$.*

Finally, as alluded before, Lemma 5.6 characterizes the distortion caused by $Q^{-1/2}$ transforming ℓ_2 to ℓ_1 space. We note that the square-root of Fisher information matrix $Q^{1/2}$ converts naturally ℓ_1 to ℓ_2. Therefore we can get the desired ℓ_1 error bound in Theorem 5.3.

Lemma 5.6. *Under assumption 5.1 and 5.2, Fisher information matrix Q satisfies that $\|\hat{A}Q^{-1/2}\|_{2\to 1} \leq 1$, and $\|Q^{-1/2}\|_{2\to 1} \leq \bar{\kappa}$. As a corollary, $Q \succeq \frac{1}{\bar{\kappa}^2}\cdot \mathrm{Id}_r$.*

6. Sample Complexity Lower Bounds

In this section we construct a natural (distribution of) word-topic matrix A with low $\Lambda_\delta(A)$ value for very small δ for which given document with $o(r^2)$ words, it is impossible to determine the support x^* even if all the nonzero coordinates of x^* are roughly $1/r$. This shows that for the task of support recovery, our algorithm in Section 4 achieves optimal sample complexity up to logarithmic factor.

Theorem 6.1. *There exists a (distribution of) word-topic matrix A with $\Lambda_\delta(A) = 1$ for $\delta = O(\sqrt{\log k/D})$ such that any algorithm $\mathcal{A}$ that takes document of $o(r^2)$ words as input cannot recover the support of the topic vector x^* that is used to generate y with probability $3/4$. This is still true when x^* is promised to have only non-zero entries that are larger than $1/r$.*

The hard instance that we constructed is pretty simple: we consider a word-topic matrix where every topic contains roughly half of words in the vocabulary, and gives probability roughly $2/D$ to each of the words. The words in the topic are uniformly randomly selected. To make this more precise, let $S_1, S_2, ..., S_k \subset [D]$ be k independent subsets that are uniformly chosen among all subsets of $[D]$. Let matrix $A_{i,j} = 1/|S_j|$ if $i \in S_j$ and $A_{i,j} = 0$ otherwise.

We first show that indeed $\Lambda_\delta(A)$ is very small, and therefore it has a good δ-biased minimum variance estimator and our algorithm works well on this matrix.

Lemma 6.2. *With high probability over the randomness of A, we have $1 - \delta \leq \Lambda_\delta(A) \leq 1$ for any $\delta \geq c\sqrt{(\log k)/D}$ where c is a sufficiently large constant.*

The proof is deferred to supplementary material. Lemma 6.2 in particular implies that if nonzero entries in the true topic vector x^* are at least $1/r$, our algorithm can detect the support with $O(r^2 \log k)$ samples. We show below that no algorithm can do much better by constructing the following hard instance:

Lemma 6.3. *With high probability over the choice of A, there exists vector r-sparse vectors x, x^- with non-zero entries bounded below from $1/r$, such that no algorithm that only takes document of $o(r^2)$ words can distinguish distributions $\mathrm{cat}(Ax)$ and $\mathrm{cat}(Ax^-)$ with probability better than $1/2 + o(1)$.*

The full proof is deferred to Section C. In fact, using the same matrix we can bound the difference between $\ell_1 \to \ell_1$ condition number $\kappa(A)$ and $\ell_1 \to \ell_\infty$ condition number $\lambda(A)$.

Lemma 6.4. *When $D \gg k \log k$, with high probability, the $\ell_1 \to \ell_1$ condition number $\kappa(A) \geq \Omega(\sqrt{k})$. When $D \gg k^2 \log k$, with high probability $\lambda(A) \leq 2$.*

We give the proof in Appendix C. Intuitively, for $\kappa(A)$ we

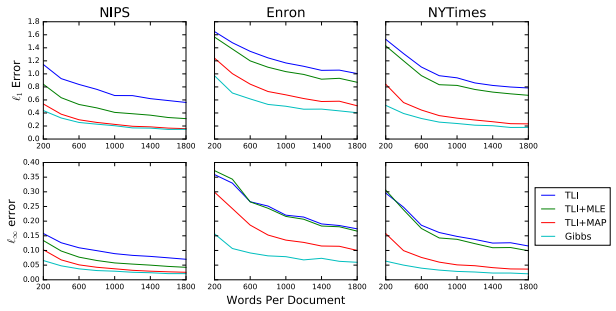

Figure 1. Estimation error on semi-synthetic data in ℓ_1 and ℓ_∞ norms: Dirichlet prior, $\alpha = r/k = 5/k$

show that the uniform mixture of first half of topics is very similar to the uniform mixture of the last half of topics. For $\lambda(A)$, we use the construction for the δ-biased linear inverse, and show that "fixing" the bias does not change the condition number by too much.

7. Experiments

The corpora we use consist of New York Times articles (295,000 documents, average document length 298), Enron emails (39,861 documents, average document length 136), and NIPS papers (1500 documents, average length 1042). We compute the word-topic matrices using the algorithm in (Arora et al., 2013a), using 100 topics for NIPS, 100 topics for Enron, and 400 topics for NYTimes.

Condition Numbers of Matrices First we empirically verify the assumption that the word-topic matrix have small $\ell_\infty \to \ell_1$ condition number (see Table 1). Solving LP (3) on 16 processors using the Mosek LP solver takes 1 minute and a half for the NIPS dataset, 4 minutes for the Enron dataset, and roughly 4 hours for the NYTimes dataset (this is partly the result of using more topics for this dataset).

Note that we only need to compute the inverse matrix *once* and then it can be used to do inference on all the documents, so the running time for computing the inverse is not a major concern. The procedure can also be easily parallelized because the LP for different rows of B are independent.

For comparison, we also list lower bounds on the $\ell_1 \to \ell_1$ condition number $\kappa(\cdot)$, the condition used by (Kleinberg & Sandler, 2008). We see that it's at least 2 times larger than $\ell_\infty \to \ell_1$ condition number. There is no efficient algorithm known for computing $\ell_1 \to \ell_1$ condition number, and therefore we only compute provable lower bounds for various dataset (see Section D for the approach).

Synthetic Experiments We first verify the recovery guarantee of our algorithm on synthetic documents. For each document, we sample $r = 5$ topics uniformly at random, and choose weights for these topics uniformly from the r-dimensional probability simplex. The results[1] are listed in Figure 2.

	$\lambda(A)$	$\lambda_{0.1}(A)$	$\kappa(A)$
NIPS	1.547	1.245	≥ 3.334
Enron	5.032	3.877	≥ 12.46
NYTimes	2.990	2.349	≥ 6.755

Table 1. Condition numbers $\lambda_\delta(A)$ and $\kappa(A)$ of word-topic matrices trained from datasets (smaller is better, always ≥ 1)

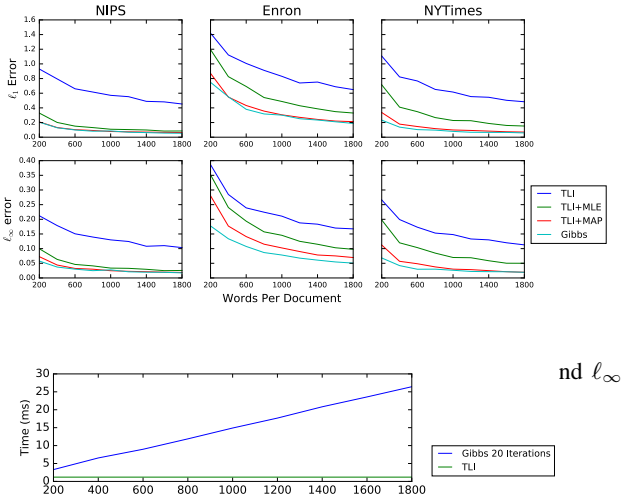

Figure 3. Speed of TLI vs Gibbs Inference (20 iterations)

We compare our TLI algorithm (Algorithm 1)[2] with the collapsed Gibbs sampling algorithm implemented in MALLET (McCallum, 2002) and its anchor word compatible extension[3]; we use 200 iteration of burn-in and 1000 further iterations of sampling. Note that when applying Gibbs sampling for the topic vectors, we treat the word-topic matrix as a fixed constant. It is not easy to do Gibbs Sampling for the prior we specified, so we compare against a Dirichlet prior with $\alpha = \frac{r}{k}\mathbf{1}$ which encourages r-sparse vectors. Note that TLI-Unnormalized is the output of the TLI algorithm as described, whereas TLI is the result after normalizing the entries to give a probability distribution. We can also improve the quality of the TLI estimate by using gradient ascent on the likelihood (see Section 5), restricted to the top r entries of our initial estimate; this is

[1] Code to reproduce the results is available at: https://github.com/frytvm/topic-inference

[2] For documents of the length found in the corpuses, the thresholding value τ used in the theoretical section is too conservative (large). As a more practical alternative we replace τ as given in the theoretical section by $\tau/4.5$. We use unbiased pseudoinverses, taking $\delta = 0$.

[3] https://github.com/mimno/anchor

denoted TLI+MLE in the figure. Finally, we give the result of gradient ascent on the posterior (treating the prior as Dirichlet, similarly to Gibbs sampling) starting from the TLI estimate, and denote this TLI+MAP.

We can also replace the uniform sparse prior with a Dirichlet prior with $\alpha = \frac{r}{k}\mathbf{1}$ and get similar results; see Figure 1.

The performance of the TLI algorithm is 3 to 5 times worse than Gibbs sampling in terms of ℓ_1 or ℓ_∞ error with same number of words. This is mostly because a simple linear estimator cannot capture the correlations between weights of different topics. The post processing using maximum likelihood improves the performance significantly. The performance of the algorithms seems to be related to the $\ell_\infty \to \ell_1$ condition number (with NIPS being best and Enron being worst).

A virtue of our algorithm is its speed. On the NYTimes dataset, computing the TLI estimate for a single document, which is just a matrix multiplication and a single thresholding step, takes approximately 0.8 milliseconds. In contrast, on a document of length 1600, a single iteration of Gibbs sampling takes approximately 1.0 ms (and to get the result in the plot we used 1000 iterations). On the Enron semi-synthetic data with uniform sparse prior and documents of length 1600, we find it takes about 20 total iterations of Gibbs sampling (15 of them as burn in) to return a result of similar accuracy to TLI. The speed of these methods for different length documents is illustrated in Figure 3.

Inference on Real Documents We run both our TLI and Gibbs on a subsample of real documents and test the similarity of results(See Table 2). Since we don't have the ground truth in this setting, we focus on recovering a small number of high-weight topics: we take the top-3 scoring topics from each estimated topic vector, and then the overlap is the cardinality of the intersection divided by 3. If we treat the Gibbs sampling result as the "ground truth", this gives the fraction of the high-probability topics our algorithm correctly finds. We note that by taking 5 instead of 3 topics from TLI, we improve recall (fraction of Gibbs topics found by TLI) at the expense of precision (fraction of TLI topics that are also from Gibbs) [4].

Corpus	NYTimes	Enron	NIPS
Overlap: 3 vs. 3	34%	26%	60%
Recall: 5 vs. 3	42%	40%	75%

Table 2. Results for the top-3 topic recovery experiment on real data, averaged over sample of size 200. In 5 vs. 3 experiment we take the top 5 topics from TLI and compare to the top 3 from Gibbs, which improves recall (cardinality of intersection over cardinality of Gibbs topics).

8. Conclusion

This work takes a step towards designing algorithms with provable guarantees for inference in topic modeling, building upon earlier work of Kleinberg and Sandler (Kleinberg & Sandler, 2008) in collaborative filtering. We use a notion of the approximate inverse of a topic matrix (as opposed to its exact inverse) and characterize the mathematical condition — namely, the $\ell_\infty \to \ell_1$ condition number — that determines how well it behaves as an estimator. Furthermore, we showed that this algorithm approximately solves inference in a document with as few as $O(r^2 \log k)$ words where k is the number of topics and r is the sparsity of the topic vector generating the document. We have showed that such guarantees are optimal in the sense that there are word-topic matrices for which it is information theoretically impossible to make meaningful conclusions about the topic vector with fewer than r^2 samples. We also show that our linear estimator identifies a reasonable set of topics, which allows us to solve (via convex programming) the maximum likelihood problem restricted to this set of topics and get better estimations in theory and practice. We also find that in practice, the standard pseudoinverse of the topic matrix is a good choice for B, though we do not have theory to support it.

The experiments show that topic model matrices associated with real-life corpora have good $\ell_\infty \to \ell_1$ condition number, and that the above method works well with synthetic documents generated using these topic matrices. Moreover the running time is comparable to a single iteration of Gibbs sampling. Topic recovery on real-life documents seems to be slightly weaker, and seems to require further modifications to be more robust to model-misspecification.

Acknowledgments. The work was supported by NSF grants CCF-1527371, DMS-1317308, Simons Investigator Award, Simons Collaboration Grant, and ONR- N00014-16-1-2329 (for Arora), Simons Award in Theoretical Computer Science and IBM PhD Fellowship (for Ma) and NSF CAREER Award CCF-1453261, MIT NEC Grant and Google Faculty Research Award (for Moitra).

References

Anandkumar, A., Foster, D., Hsu, D., Kakade, S., and Liu, Y. Two svds suffice: Spectral decompositions for probabilistic topic modeling and latent dirichlet allocation. In *NIPS*, 2012.

Anandkumar, Animashree, Ge, Rong, Hsu, Daniel, and Kakade, Sham M. A tensor approach to learning mixed membership community models. *Jour-*

[4] We only list recall values because in this setting precision $= 3/5 \cdot$ recall.

nal of Machine Learning Research, 15(1):2239–2312, 2014. URL http://dl.acm.org/citation.cfm?id=2670323.

Arora, Sanjeev, Ge, Rong, Halpern, Yonatan, Mimno, David M., Moitra, Ankur, Sontag, David, Wu, Yichen, and Zhu, Michael. A practical algorithm for topic modeling with provable guarantees. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Proceedings*, pp. 280–288. JMLR.org, 2013a. URL http://jmlr.org/proceedings/papers/v28/arora13.html.

Arora, Sanjeev, Ge, Rong, and Moitra, Ankur. New algorithms for learning incoherent and overcomplete dictionaries. *ArXiv*, 1308.6273, 2013b.

Arora, Sanjeev, Bhaskara, Aditya, Ge, Rong, and Ma, Tengyu. Provable bounds for learning some deep representations. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pp. 584–592, 2014. URL http://jmlr.org/proceedings/papers/v32/arora14.html.

Bickel, Peter J., Ritov, Yaacov, and Tsybakov, Alexandre B. Simultaneous analysis of lasso and dantzig selector. *Ann. Statist.*, 37(4):1705–1732, 08 2009. doi: 10.1214/08-AOS620. URL http://dx.doi.org/10.1214/08-AOS620.

Blei, D., Ng, A., and Jordan, M. Latent dirichlet allocation. *Journal of Machine Learning Research*, pp. 993–1022, 2003. Preliminary version in *NIPS* 2001.

Candes, E.J. and Tao, T. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51 (12):4203–4215, Dec 2005. ISSN 0018-9448. doi: 10.1109/TIT.2005.858979.

Ge, Rong, Huang, Qingqing, and Kakade, Sham M. Learning mixtures of gaussians in high dimensions. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pp. 761–770, 2015. doi: 10.1145/2746539.2746616. URL http://doi.acm.org/10.1145/2746539.2746616.

Griffiths, Thomas L and Steyvers, Mark. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235, 2004.

Hsu, Daniel and Kakade, Sham M. Learning mixtures of spherical gaussians: moment methods and spectral decompositions. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, pp. 11–20, 2013.

Janzamin, M., Sedghi, H., and Anandkumar, A. Beating the Perils of Non-Convexity: Guaranteed Training of Neural Networks using Tensor Methods. *ArXiv e-prints*, June 2015.

Javanmard, Adel and Montanari, Andrea. Confidence intervals and hypothesis testing for high-dimensional regression. *Journal of Machine Learning Research*, 15 (1):2869–2909, 2014. URL http://dl.acm.org/citation.cfm?id=2697057.

Kleinberg, Jon M. and Sandler, Mark. Using mixture models for collaborative filtering. *J. Comput. Syst. Sci.*, 74(1):49–69, 2008. doi: 10.1016/j.jcss.2007.04.013. URL http://dx.doi.org/10.1016/j.jcss.2007.04.013.

McCallum, A.K. Mallet: A machine learning for language toolkit, 2002. http://mallet.cs.umass.edu.

Meinshausen, Nicolai and Yu, Bin. Lasso-type recovery of sparse representations for high-dimensional data. *Ann. Statist.*, 37(1):246–270, 02 2009. doi: 10.1214/07-AOS582. URL http://dx.doi.org/10.1214/07-AOS582.

Moitra, Ankur and Saks, Michael E. A polynomial time algorithm for lossy population recovery. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pp. 110–116, 2013. doi: 10.1109/FOCS.2013.20. URL http://dx.doi.org/10.1109/FOCS.2013.20.

Moitra, Ankur and Valiant, Gregory. Settling the polynomial learnability of mixtures of gaussians. In *the 51st Annual Symposium on the Foundations of Computer Science (FOCS)*, 2010.

Yao, Limin, Mimno, David, and McCallum, Andrew. Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pp. 937–946, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-495-9. doi: 10.1145/1557019.1557121. URL http://doi.acm.org/10.1145/1557019.1557121.