# Tensor Decomposition for Topic Models: An Overview and Implementation

Chicheng Zhang, E.D. Gutiérrez, Alexander Asplund, Linda Pescatore

December 10, 2012

## 1 Introduction

The goal of a topic model is to characterize observed data in terms of a much smaller set of unobserved *topics*. Topic models have proven especially popular for information retrieval. Latent Dirichlet Allocation (LDA) is the most popular generative model used for topic modeling.

Learning the optimal parameters of the LDA model efficiently, however, is an open question. As [2] point out, the traditional techniques for learning latent variables have major disadvantages when it comes to topic modeling. Straightforward maximum likelihood estimation does not produce a closed-form solution for LDA, and its approximations are NP-hard. Approaches relying on Expectation-maximization (EM) have been the most popular way of learning LDA [4]. Unfortunately, such approaches suffer from a lack of guarantees about the quality of the locally optimal solutions they produce. They also exhibit slow convergence. Markov chain Monte Carlo (MCMC) approaches [5] are prone to failure (due to non-ergodicity, for example), and can also exhibit slow mixing. For these reasons, the tensor decomposition approach on high-order moments of the data seems like a promising option for recovering the topic vectors.[1] This approach has been applied successfuly to other latent variable models, such as Hidden Markov Models and Gaussian Mixture Models [2] [3].

### 1.1 Outline

The remainder of this paper, broadly baseed on [1] and [2] is structured as follows: in §2 we will present a description of the LDA generative model, as well as a formulation of this model in terms of observed binary vectors, a latent topic matrix, and latent topic mixture vectors. In §3 we will describe how tensor decomposition methods can be used to find latent topic vectors. Then in §4 we will outline several approaches for implementing the tensor decomposition and estimation of the empirical moments of the data, and in §5 we sketch out some sample complexity bounds resulting from these approaches. In §6 we will present some experiments that evaluate the performance of the tensor decomposition methods. Finally we conclude with a discussion of our results and future work to evaluate the performance of tensor decomposition methods.

## 2 Latent Dirichlet Allocation

### 2.1 General Description

Suppose that we want to model multiple documents each composed of multiple words, in terms of a smaller number of latent topics. Our observed data are merely the set of words occurring in each document. In the LDA model, we make the simplifying assumption that each document is a "bag-of-words" where the precise ordering of the words does not affect the semantic content of the document. We attempt to model each document $\ell$ as containing a mixture of $K$ unobserved topics denoted by the topic mixture vector $\mathbf{h}^\ell \in \Delta^{K-1}$ (where the $\Delta^{K-1}$ denotes the $K$-dimensional simplex). $\mathbf{h}^\ell$ for each document is drawn independently according to a Dirichlet distribution with concentration parameter vector $\alpha = [\alpha_1, \alpha_2, ..., \alpha_K]$. We will let $\alpha_0 = \sum_{k=1}^{K} \alpha_k$ represent the sum of this parameter vector, which is sometimes referred to as the *precision* of the Dirichlet distribution. Then we assume that the topic $k$ pertaining to a word $t$ in document $\ell$ is drawn independently from a multinomial with parameters $\mathbf{h}^\ell$. Finally, the word type for the word is drawn independently from a multinomial with parameters determined by the topic distribution vector $\phi_k \in \Delta^{|Voc|-1}$ that corresponds to topic $k$, where $|Voc|$ is the size of the lexicon (i.e., the number of word types) in the entire data set.

### 2.2 Formulation of the Data

To make the tensor decomposition approach clear, we will need to represent our data as a set of binary vectors. Denote the $|Voc|$-dimensional basis vector by $\mathbf{e}_i$. Then let $\mathbf{x}_t{}^\ell = \mathbf{e}_i$ if the $t^{th}$ term in document $\ell$ belongs to word class $i$. We also collect the $\phi_k$ vectors into a $|Voc|$-by-$K$ latent topic matrix $\Phi = (\phi_1, \dots \phi_K)$.

# 3 Tensor Decomposition Approach: Intuition

## 3.1 Why cross-moments?

With the above formulation in place, we can begin to make note of some properties of the hidden moments of the data. By the assumption that the topic mixture vector $\mathbf{h}$ is drawn according to the Dirichlet distribution, we know that the expected value of the $k^{th}$ element of this vector is

$$\mathbf{E}[\mathbf{h}_k] = \frac{\alpha_k}{\alpha_0}.$$

Due to our formulation above, for a word $\mathbf{x_1}$ chosen from the set $\{\mathbf{x}_t{}^\ell\}$, the expectation of $x_1$ conditional on the topic mixture vector $\mathbf{h}$ is

$$\mathbf{E}[\mathbf{x_1}|\mathbf{h}] = \Phi\mathbf{h} = \sum_{k=1}^{K} \phi_k \mathbf{h}_k.$$

This equation exhibits clearly the relationship between the observed $\mathbf{x}_1$ and the hidden $\phi_k$, but it assumes knowledge of $\mathbf{h}$, which is hidden. However, we can observe our marginal expectation $\mathbf{E}[\mathbf{x}_1]$, and recalling the relationship between marginal and conditional expectations we can see that

$$\mathbf{E}[\mathbf{x}_1] = \mathbf{E}[\mathbf{E}[\mathbf{x}_1|\mathbf{h}]] = \Phi\mathbf{E}[\mathbf{h}] = \sum_{k=1}^{K} \frac{\alpha_k}{\alpha_0}\phi_k.$$

This exposition hints at how the moments of the data could help us recover the latent vectors $\phi_k$. However, note that under our topic model, the higher moments of a single word $\mathbf{x}_1$ are trivial. Recall that a crucial part of the structure of the LDA model is all the words in a document share a topic mixture vector $\mathbf{h}$, but within-document information does not figure in the higher moments of single words. For this reason, we derive the *cross-moments* of pairs and triples of distinct words $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$ in $\{\mathbf{x}_t^\ell\}$. Note that such pairs and triples of words are conditionally independent given $\mathbf{h}$. This allows us to write the second-order cross-moment in terms of $\Phi$ and $\mathbf{h}$ as

$$\mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2] = \mathbf{E}[\Phi\mathbf{h} \otimes \Phi\mathbf{h}] = \mathbf{E}[\mathbf{h} \otimes \mathbf{h}](\Phi^T, \Phi^T)$$

and similarly we can write the third-order cross-moment as

$$\mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2 \otimes \mathbf{x}_3] = \mathbf{E}[\Phi\mathbf{h} \otimes \Phi\mathbf{h} \otimes \Phi\mathbf{h}] = \mathbf{E}[\mathbf{h} \otimes \mathbf{h} \otimes \mathbf{h}](\Phi^T, \Phi^T, \Phi^T).$$

We have now written our observed cross-moments in terms of $\Phi$ and the cross-moments of $\mathbf{h}$. Fortunately, due to our assumption that the topic assignment vector $\mathbf{h}$ is drawn from a Dirichlet distribution, we can derive closed-form expressions for the cross-moments of $\mathbf{h}$ in terms of the Dirichlet parameters (see Appendix A). And using these expressions, we can explicitly write our observed cross-moments solely in terms of $\alpha$ and $\Phi$:

$$\mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2] = \frac{1}{\alpha_0(\alpha_0+1)}(\Phi\alpha \otimes \Phi\alpha + \sum_{k=1}^{K} \alpha_k(\phi_\mathbf{k} \otimes \phi_\mathbf{k}))$$

$$\mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2 \otimes \mathbf{x}_3] = \frac{1}{\alpha_0(\alpha_0+1)(\alpha_0+2)}(\Phi\alpha \otimes \Phi\alpha \otimes \Phi\alpha)$$

$$+ \sum_{k=1}^{K} \alpha_k(\phi_\mathbf{k} \otimes \phi_\mathbf{k} \otimes \mathbf{\Phi}\alpha + \phi_\mathbf{k} \otimes \mathbf{\Phi}\alpha \otimes \phi_\mathbf{k} + \mathbf{\Phi}\alpha \otimes \phi_\mathbf{k} \otimes \phi_\mathbf{k}) + \sum_{k=1}^{K} 2\alpha_k(\phi_\mathbf{k} \otimes \phi_\mathbf{k} \otimes \phi_\mathbf{k}))$$

The last terms in the two expressions above are especially promising candidates for recovering the latent vectors $\phi_k$. By the following algebraic manipulations, we can isolate these terms as noncentral moments of $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{x}_3$:

$$M_1 := \mathbf{E}[\mathbf{x}_1] = \sum_{k=1}^{K} \frac{\alpha_k}{\alpha_0}\phi_k$$

$$M_2 := \mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2] - \frac{\alpha_0}{\alpha_0+1}(M_1 \otimes M_1)$$

$$M_3 := \mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2 \otimes \mathbf{x}_3]M_1 - \frac{\alpha_0}{\alpha_0+2}(\mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2 \otimes M_1] + \mathbf{E}[\mathbf{x}_1 \otimes M_1 \otimes \mathbf{x}_2] + \mathbf{E}[M_1 \otimes \mathbf{x}_1 \otimes \mathbf{x}_2])$$

$$+ \frac{2\alpha_0^2}{(\alpha_0+2)(\alpha_0+1)}(M_1 \otimes M_1 \otimes M_1)$$

where $M_i$ denotes the $i^{th}$ non-central moment. Of special note is that these manipulations are in terms only of the observed moments themselves and of the parameter $\alpha_0$, not the entire $\alpha$ vector, which is a necessary input for EM methods for LDA. Our newly defined non-central moments can now be written as linear combinations of tensor powers of the $\phi_k$ vectors:

$$M_1 = \sum_{k=1}^{K} \frac{\alpha_k}{\alpha_0} \phi_k \tag{1}$$

$$M_2 = \sum_{k=1}^{K} \frac{\alpha_k}{(\alpha_0 + 1)\alpha_0} (\phi_k \otimes \phi_k) \tag{2}$$

$$M_3 = \sum_{k=1}^{K} \frac{2\alpha_k}{(\alpha_0 + 2)(\alpha_0 + 1)\alpha_0} (\phi_k \otimes \phi_k \otimes \phi_k). \tag{3}$$

### 3.2 Whitening Matrix

We now have moments defined in terms of the observed data and $\alpha_0$ that can be expressed as linear combinations of tensor powers the variables of interest. If we can symmetrize our moments in some way and express them in terms of orthogonal matrices, techniques for decomposing into such matrices can in principle be used to recover the latent variables. Suppose we found any matrix $W$ such that it whitens the second moment: $M_2(W, W) := W^T M_2 W = I$. Then this can be written using (2) as

$$M_2(W, W) = \sum_{k=1}^{K} \frac{\alpha_k}{\alpha_0 + 1} W^T \phi_k \otimes W^T \phi_k$$

$$= \sum_{k=1}^{K} W^T \sqrt{\frac{\alpha_k}{\alpha_0 + 1}} \phi_k \otimes W^T \sqrt{\frac{\alpha_k}{\alpha_0 + 1}} \phi_k,$$

and defining $\beta_k = W^T \sqrt{\frac{\alpha_k}{\alpha_0 + 1}} \phi_k$, we see that $M_2(W, W) = \sum_{k=1}^{K} \beta_k \otimes \beta_k = I$.

In other words, the $\beta_k$ are orthonormal vectors, and the whitened moment matrix is amenable to an orthogonal matrix decomposition. From these $\beta_k$ it would be possible to recover the $\phi_k$ vectors, as the $\beta_k$'s are merely linear combinations of the $\phi_k$'s and $W$. However, note that the solution produced by such a decomposition would not be unique in the general case, only in the case where no two $\alpha_k$'s are equal. Fortunately, application of the same whitening matrix $W$ yields the following results on the third moment:

$$M_3(W, W, W) = \sum_{k=1}^{K} \frac{2\alpha_k}{(\alpha_0 + 2)(\alpha_0 + 1)\alpha_0} W^T \phi_k \otimes W^T \phi_k \otimes W^T \phi_k$$

$$= \sum_{k=1}^{K} \frac{2\sqrt{(\alpha_0 + 1)\alpha_0}}{(\alpha_0 + 2)\sqrt{\alpha_k}} (\beta_k \otimes \beta_k \otimes \beta_k) := \sum_{k=1}^{K} \gamma_k (\beta_k \otimes \beta_k \otimes \beta_k).$$

Thus, the observed third moment can also be decomposed in terms of orthogonal vectors that are linear combinations of the $\phi_k$'s and a whitening matrix $W$ that depends on the observed data. The details of this tensor decomposition are covered in the Implementation section below.

## 4 Implementation

### 4.1 Empirical Estimation of Moments and Whitening Matrix

While the formulation of the data in terms of binary basis vectors $\mathbf{x}_\ell^t$ is helpful to develop intuition for our technique, it is quite cumbersome from an implementation point of view. The storage complexity of such an implementation grows linearly in the number of word tokens. Since the order of words within documents does not matter for LDA, a much more compact representation in terms of word-type count vectors is possible. Such a representation grows in the number of word types. We have derived estimates of our empirical moments and their products in terms of such count vectors, and these estimates are in Appendix B. Another matter of practical concern is estimating the whitening matrix $W$. As [1] point out, if we take our empirical second moment and find its singular value decomposition $\hat{M}_2 = A\Sigma A^T$, then the matrix $\hat{W} = A\Sigma^{-\frac{1}{2}}$ fulfills the property of whitening $\hat{M}_2$. Thus, $W$ can be efficiently estimated from our empirical moment estimators.

## 4.2 Tensor Decomposition Approaches

Suppose the empirical versions of $M_1$, $M_2$, and $M_3$ are observed, recall that the goal of the tensor decomposition is to recover $\phi_k, k = 1, 2, \ldots, K$. Several approaches have been introduced in [1],[2], [3], and [6] and are reviewed below. Our experiments focus on variants of the first two approaches only.

### 4.2.1 Tensor Power Method[2]

First we find the whitening matrix $W$, defined as above, and define $T := M_3(W, W, W)$. A power-deflation approach can be used to recover the $\beta_k$ from this tensor, because if we start with a $u_0 = \sum_{k=1}^{K} c_k\beta_k + \beta^{\perp}$ randomly drawn from unit shpere, where $\beta^{\perp}$ is the component outside the span of $(\beta_1, \ldots, \beta_k)$. Then after several iterations of $u_{t+1} = T(I, u_t, u_t)$, the result will be $u_{t+1} = \sum_{k=1}^{K}(2^t - 1)\gamma_k 2^t c_k\beta_k$, so when initially $c_k\gamma_k$ dominates, then it will dominate in the whole run, and the convergence speed is with respect to the rate of the largest $c_k\gamma_k$ to the second largest component of $c_k\gamma_k$, in initialization. Also note After extracting the approximate $\beta_k$, $T(\beta_k, \beta_k, \beta_k) = \gamma_k$ can be used to recover $\alpha_k$. After a pair $(\gamma_k, \beta_k)$ is extracted, we can deal with the new tensor $T - \gamma_k\beta_k \otimes \beta_k \otimes \beta_k$, and do this recursively.

Once the $\beta_k$ are recovered, because $\beta_k = \frac{\alpha_k}{\alpha_0(\alpha_0+1)}W^T\phi_k$, then for $\phi_k$ is in the column space of $W$, we can see that $\phi_k = Wc_k$,$c_k = \sqrt{\frac{\alpha_0(\alpha_0+1)}{\alpha_k}}(W^TW)^{-1}\beta_k$.

### 4.2.2 SVD Method[1]

The first two steps are simliar to the tensor power approach. When $T = M_3(W, W, W)$ is found, we project T into a matrix, i.e.:

$$T(I, I, \theta) = \sum_{k=1}^{K} \gamma_k({\beta_k}^T\theta)\beta_k \otimes \beta_k$$

it can also be treated as a "thin" SVD form of $T(I, I, \theta)$:

$$T(I, I, \theta) = USU^T = (\beta_1, \ldots, \beta_K)diag(\gamma_1({\beta_1}^T\theta), \ldots, \gamma_K({\beta_K}^T\theta))(\beta_1, \ldots, \beta_K)^T$$

So if we do SVD of the $T(I, I, \theta)$ matrix, as long as $\gamma_1({\beta_1}^T\theta), \ldots, \gamma_K({\beta_K}^T\theta)$ are distinct (in the empirical version we require them to have a not-too-small gap, note that the tensor power approach does not have this problem), then we can recover $\beta_k$.

Following the tensor power approach, we can first recover $\alpha_k$, then $\phi_k$.

## 4.3 Simultaneous power method[1]

To improve computational efficiency, we do not have to explicitly calculate the $M_2$, $M_3$ (in our implementation we calculate $M_2$, which can be improved, but never calculate $M_3$). [1] suggests a generic power method to calculate approximate orthogonal decomposition for matrices and tensors:

**Matrix case:** Suppose input matrix is M, then start with random initialization of $(v_1^0, \ldots, v_K^0)$, calculate $(M(I, v_1^t), \ldots, M(I, v_K^t))$ and then orthonormalize to get $(v_1^{t+1}, \ldots, v_K^{t+1})$, repeat the procedure until convergence.

**Tensor case:** Suppose input tensor is T, then start with random initialization of $(v_1^0, \ldots, v_K^0)$, calculate $(T(I, v_1^t, v_1^t), \ldots, T(I, v_K^t, v_$ and then orthonormalize them to get $(v_1^{t+1}, \ldots, v_K^{t+1})$, repeat the procedure until convergence.

Our first implementation used this approach to get the approximate $\beta_k$s, which works empirically, but its theoretical guarantee are to be verified. So this approach is not the main concern of the report.

## 4.4 Other methods

Three other approaches for recovering the orthonormal vectors are provided in [**?**] and [3]. However, since we did not implement these, our review of these methods is in Appendix C.

# 5 Theoretical Guarantees: Sample Complexity

First note that $\hat{M}_2(\hat{M}_3)$ will converge to $M_2(M_3)$, as can be seen by considering their vector stacking and applying McDiarmid's Lemma. With probability $1 - \delta$,

$$\sum_{i=1}^{d}\sum_{j=1}^{d}(\hat{M}_{2i,j} - M_{2i,j})^2 < \frac{(1 + \sqrt{\ln 1/\delta})^2}{N}$$

$$\sum_{i=1}^{d}\sum_{j=1}^{d}\sum_{k=1}^{d}(\hat{M}_{3i,j,k} - M_{3i,j,k})^2 < \frac{(1 + \sqrt{\ln 1/\delta})^2}{N}$$

Then it is straightforward to see that $||M_2 - \hat{M}_2|| < E_P, ||M_3(I,I,\eta) - \hat{M}_3(I,I,\eta)|| < E_T||\eta||$, $E_P = E_T = \frac{(1+\sqrt{\ln 1/\delta})}{\sqrt{N}}$. We verified these convergence results on two datasets, as we explain in §6 and §7 below.

Let $W = \hat{W}(\hat{W}^T M_2 \hat{W})^{\dagger\frac{1}{2}}$, then $W$ whitens $M_2$, but its range may not equal $range(M_2) = range(\Phi)$. Matrix perturbation theory yields the unsurprising result that for $E_P$ small,

$$||W^T\tilde{\Phi} - \hat{W^T}\tilde{\Phi}|| \leq \frac{4}{\sigma_k(\tilde{\Phi})^2}E_P,$$

$$||\hat{W}^\dagger - W^\dagger|| \leq \frac{6\sigma_1(\tilde{\Phi})}{\sigma_k(\tilde{\Phi})^2}E_P$$

$$||\Pi - \Pi_W|| \leq \frac{4}{\sigma_k(\tilde{\Phi})^2}E_P$$

Next, we consider the matrix $\hat{M}_3(\hat{W},\hat{W},\hat{W})$ that we are going to decompose. Then:

$$||\hat{M}_3(\hat{W},\hat{W},\hat{W}) - M_3(W,W,W)|| < c(\frac{(\alpha_0+2)^{1/2}E_P}{p_{min}^{3/2}\sigma_k(\Phi)^2} + \frac{(\alpha_0+2)^{3/2}E_T}{p_{min}^{3/2}\sigma_k(\Phi)^3})$$

Denote this deviation by $E$. Because $W$ whitens $M_2$, $M_3(W,W,W)$ has an orthogonal decomposition. Then for the SVD/Tensor Power decompositions of $\hat{M}_3(\hat{W},\hat{W},\hat{W})$, we have the results:
(**For SVD Method**), w.p 3/4:

$$||\hat{\beta}_i - \beta_i|| < c_1 K^3\sqrt{\alpha_0+2}E$$

(**For Tensor Power Method**), w.p 3/4:

$$||\hat{\beta}_i - \beta_i|| < c_2\sqrt{\alpha_0+2}E$$

So if we look at reconstruction accuracy,

$$||\Phi_i - \frac{(\hat{W}^\dagger)^T}{\hat{\beta}_i}|| \leq ||\Pi - \Pi_W|| + \frac{W^\dagger}{Z_i}||\hat{\beta}_i - \beta_i|| + \frac{1}{Z_i}||W^\dagger - \hat{W}^\dagger|| + ||\hat{W}^\dagger||\frac{1}{Z_i} - \frac{1}{\hat{Z}_i}|$$

It turns out the second and fourth terms dominate, and w.p. $1-\delta$ over the random examples given, it is bounded by:
(**For SVD Method**), w.p. 3/4 over the randomness of choice of $\theta$:

$$c_1'\frac{K^3(\alpha_0+2)^2}{p_{min}^2\sigma_k(\Phi)^3}(\frac{1+\sqrt{\ln 1/\delta}}{\sqrt{N}})$$

(**For Tensor Power Method**), w.p. 3/4 over the randomness of choice of iteration startpoint:

$$c_2'\frac{(\alpha_0+2)^2}{p_{min}^2\sigma_k(\Phi)^3}(\frac{1+\sqrt{\ln 1/\delta}}{\sqrt{N}})$$

One remarkable aspect of these results is that the error bound of the SVD method depends polynomially on $K^3$, while the error bound of the tensor power method does not depend on $K$ at all. For both methods, these results guarantee that the error falls off inversely with the square root of the number of word tokens in the sample.

# 6 Experiments

## 6.1 Datasets

We tested the empirical algorithm on two real-world datasets. The first dataset is Classic3/Classic4. Classic4 is comprises four different collections of abstracts: CACM, CISI, CRAN, and MED. These collections roughly correspond to the topics of computer science, information science, aeronautics, and medicine, respectively. Classic3 is the same as Classic4, with the exclusion of CACM. The second dataset we used is the 20Newsgroups dataset. It consists of postings on 20 Usenet newsgroups, on diverse topics such as computers, religion, and politics. In order to evaluate quantitatively the performance of the algorithm, we had to set a 'ground truth' for our datasets by assigning topic mixtures to the documents in the datasets. We settled on assigning a single topic per document, which corresponds to $\alpha_0 = 0$. Each document in Classic3/Classic4 was assigned with a topic label determined by the collection of abstracts it came from (therefore, $K = 3$ for Classic3 and $K$=4 for Classic4). For 20Newsgroups, it did not seem appropriate to assign a separate topic for each newsgroup, since there is much topical overlap among groups. For instance, `comp.windows.x` and `comp.os.ms-windows` seem to share a great deal of vocabulary. Instead, we collected the groups into $K = 6$ topics, following [8]. These topics are found in Appendix D.

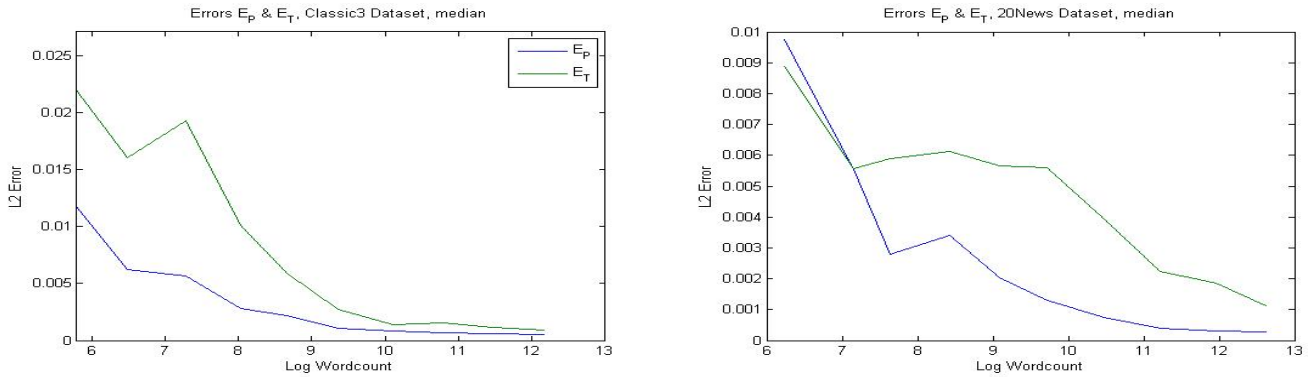


Figure 1: empirical deviation of $M_2$ (blue line) and $M_3$ (green line) for CLASSIC3 (left) and 20NEWSGROUPS (right)

### 6.2 Procedure

Our overall empirical algorithm was as follows:

1. Construct empirical moments $\hat{M}_2$, $\hat{M}_3$(implicitly)
2. **Whiten**: Let $\hat{W} = A\Sigma^{-1/2}$ where $A\Sigma A^T$ is the SVD of $\hat{M}_2$.
3. **Tensor Decomposition**: **(SVD Method)** Calculate the left singular vectors of $\hat{W}\hat{M}_3(\hat{W}\theta)\hat{W}$ as in section 4.2.2.

**(Tensor Power Method, using deflation)** Calculate the eigenvectors of $\hat{M}_3(\hat{W}, \hat{W}, \hat{W})$ as in Section 4.2.1, extracting one pair at a time.

**(Tensor Power Method, simultaneous)** Calculate the eigenvectors of $\hat{M}_3(\hat{W}, \hat{W}, \hat{W})$ as in Section 4.2.1, without deflating.

4. **Reconstruct**: $Z_i = (\hat{W}v_i)^T\hat{M}_3(\hat{W}v_i)(\hat{W}v_i)$ and $\hat{\phi}_i = (\hat{W}^+)^T v_i/Z_i$.

We randomly divided CLASSIC3 and 20NEWSGROUPS data into three folds, each composed of one-third of the documents. We then used a cross-validation scheme, where we tested the algorithm on each combination of two folds, while using the documents and topic labels of the third fold as held-out data to compute an estimate of the "ground truth" moments and latent variable matrix $\Phi$. Suppose we wish to estimate the ground-truth distribution of our data from $|Docs|$ different documents in our held-out data, and we have a label vector $y \in \mathcal{R}^{|Docs|}$ as well as a count vector $c^\ell$ for each document, where $c_i^\ell$ is the count of word type $i$ in document $\ell$. Then we estimate the $i^{th}$ element of $\phi_k$ as

$$\tilde{\phi}_{k,i} = \frac{\sum_{\ell=1}^{|Docs|} c_i^\ell \mathbf{1}_{y_i=k}}{\sum_{i=1}^{|Voc|}\sum_{\ell=1}^{|Docs|} c_i^\ell \mathbf{1}_{y_i=k}}. \tag{4}$$

To assess the performance of the three decomposition techniques and compare it to the sample complexity bounds, we computed the $\hat{\phi}_k$ and empirical moments for varying sample sizes, on a logarithmic scale, using each of the three methods with $\alpha_0 = 0$ . We then recorded the L2 error between the ground truth estimates of the moments and $\tilde{\phi}_k$'s derived from the held-out data, and the empirical moments and $\hat{\phi}_k$'s returned by the three tensor decomposition techniques. Note that tensor decomposition methods only return the matrix $\hat{\Phi}$ up to a permutation; we used a bipartite matching algorithm, the HUNGARIAN algorithm [7], to match the $\hat{\phi}_k$'s to the $\tilde{\phi}_k$'s.

## 7 Results and Conclusion

1.

Qualitative results showing that our implementation recovers semantically reasonable topics are included in Appendix E. **??** and 2 show the the L2 errors in estimation of the empirical moments and the L2 error in estimation of the topic vectors, respectively. As can be seen, all the errors fall off with sample size, but they do not quite fall off as $1/\sqrt{(}N)$. Performance seems roughly the same for all three methods, and despite the polynomial dependence of the SVD method on the number of topics, the performance of this method seems similar on both datasets.

There are several possible explanations for our results. First of all, we must question the quality of the "ground truth" we estimated. Mislabeling is a problem, and the ground truth topics approached by LDA need not correspond to our topic labels at all. Secondly, a misspecified $K$ will make a big difference in our implementation. For example, consider a single-topic model (where $\alpha_0$ is approaches 0) where we estimate a "ground truth" consisting of two topics (say about library science and dynamics) using labeled documents, with topic probability distributions $\phi_1$ and $\phi_2$ , while actually the documents of the second topic have two subcategories whose probability distribution are $\phi_{2,1}$, $\phi_{2,2}$ (say aerodynamics and thermodynamics) that rarely if ever mix in the same document. Then the true $K$ is 3, while the $K$ we used for our evaluation is 2. Moreover,

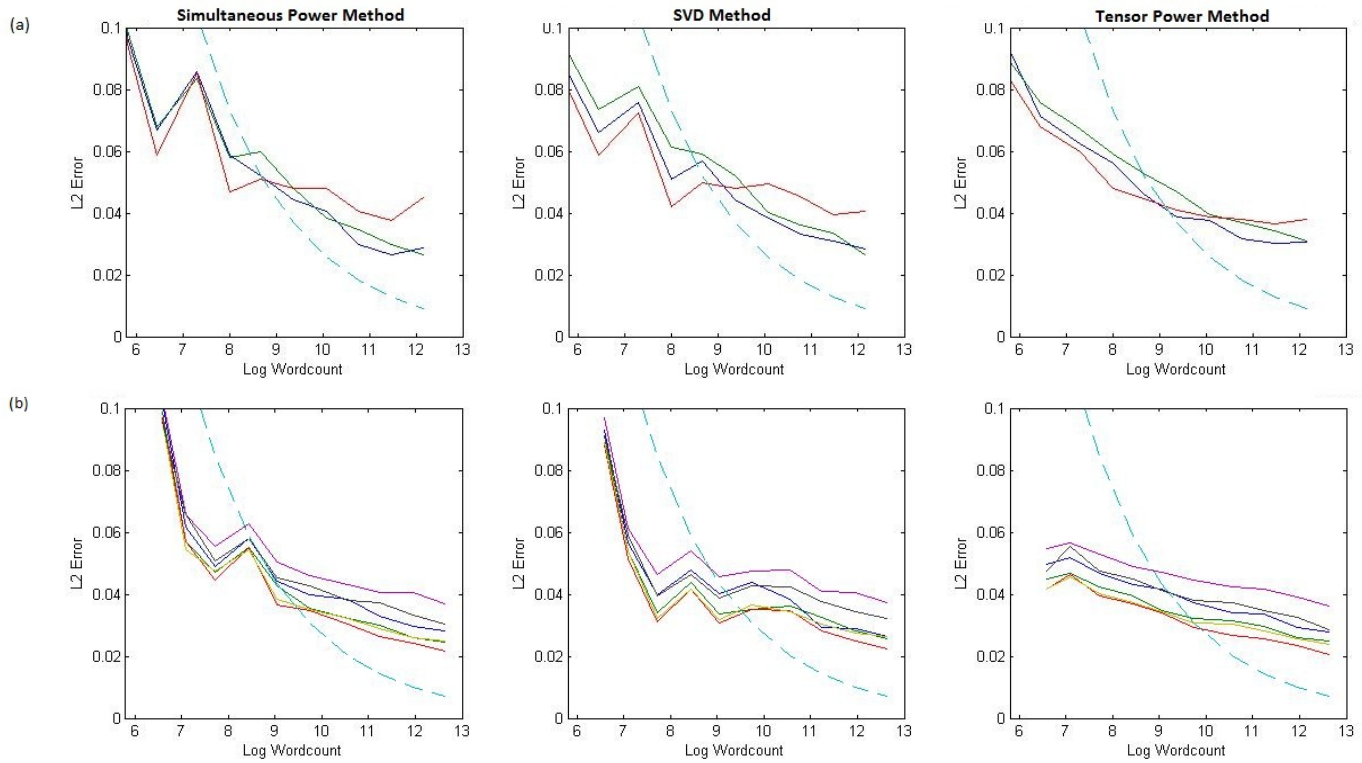


Figure 2: $||\hat{\phi}_k - \tilde{\phi}_k||$ for CLASSIC3 (top) and 20NEWSGROUPS (bottom). The dashed line represents $O(1/\sqrt{N})$.

if we look at the $M_2$ matrix, we assume that

$$M_2 = \frac{1}{\alpha_0(\alpha_0+1)}(\alpha_1\phi_1\phi_1^T + \alpha_2\phi_2\phi_2^T)$$

but actually

$$M_2 = \frac{1}{\alpha_0(\alpha_0+1)}(\alpha_1\phi_1\phi_1^T + \alpha_{2,1}\phi_{2,1}\phi_{2,1}^T + \alpha_{2,2}\phi_{2,2}\phi_{2,2}^T)$$

Suppose, say $\alpha_1 = 0.2\alpha_0$, $\alpha_{2,1} = \alpha_{2,2} = 0.4\alpha_0$ and the $\phi_1, \phi_{2,1}, \phi_{2,2}$ do not overlap in thier support sets (which ensures they are orthogonal), then the $W$ extracted using the SVD of $M_2$ may favor $\phi_{2,1}$ and $\phi_{2,2}$, and its columns will be exactly ($\phi_{2,1}$, $\phi_{2,2}$). Running the algorithm with $K = 3$, we may recover these two subtopics and overlook the main topic $\phi_1$, which has a critical impact on further testing of the deviations. It is not clear to us in general what happens when $K$ used by the algorithm is less than the true $K$. But if $K$ used by the algorithm is greater than the true $K$, then it can be guaranteed that we recover all "main" topics, but some of them may be found as the combination of several subtopics extracted due to the inherent structure of data, i.e. it will reduce false negative topics. The price we pay is that it may produce more false positive topics, i.e. some topics that are not reasonable. So how to select optimal $K$ for the algorithm, if we do not know it beforehand, is an open question. In this respect, it is interesting to note that the rank of $M_2$ would be $K$ in the absence of noise.

To address whether substructure within our ground truth topics is responsible for our results, we used the LDA generative model to simulate a data set of the same size as the CLASSIC3 dataset, using estimated $\phi_k$'s computed from the CLASSIC3 corpus using equation (4) and setting $\alpha_0 = 0$. These results are in 3. Performance now seems to be closer to the theoretical bound, but still not exact. This suggests there may be other reasons for deviation from the theoretical results.

It might be instructive to run a traditional EM- or MCMC-based approach for learning LDA on our data, in order to compare to our tensor decomposition results. However, how to objectively compare the solutions found by these fundamentally different methods is not straightforward. Finally, we note that we have not fine-tuned the iterative parameters of the tensor power approach, which could give this method a boost in performance.

Further work needs to be done in order to assess tensor decomposition approaches for LDA. In addition to the suggestions above, it would also be interesting to compare the efficiency with which the tensor decomposition approach approaches the ground truth, compared with an EM-based approach, as the fact that tensor decomposition approaches avoid the problem of local minima is a big advantage of this method.

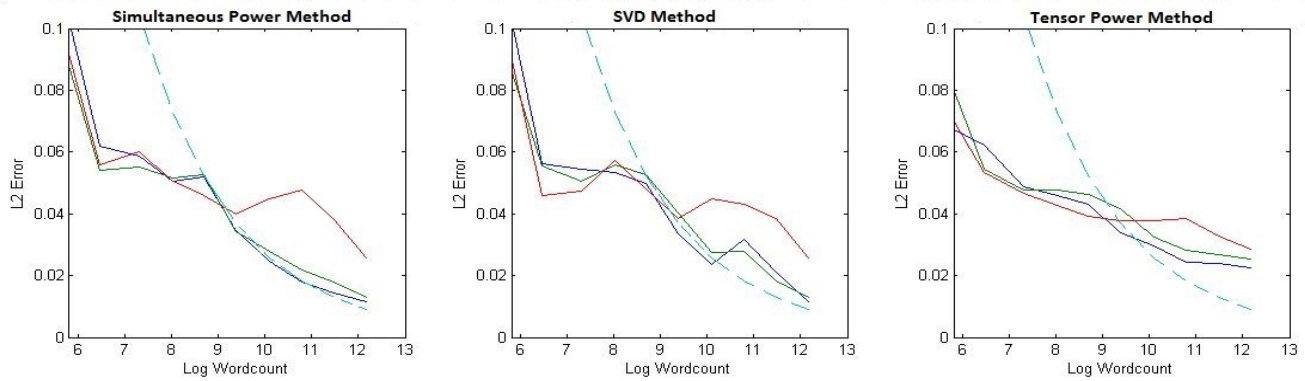


Figure 3: $||\hat{\phi}_k - \tilde{\phi}_k||$ for data simulated from CLASSIC3. The dashed line represents $O(1/\sqrt{N})$.

# References


[1] A. Anandkumar, D.P. Foster, D. Hsu, S.M. Kakade, Yi-Kai Liu. Two SVDs Suffice: Spectral Decompositions for Probabilistic Topic Modeling and Latent Dirichlet Allocation. ArXiv Report, arXiv: 1204.6703v3, 2012.

[2] A. Anandkumar, R. Ge, D. Hsu, S.M. Kakade, M. Telgarsky. Tensor Decompositions for Learning Latent Variable Models. ArXiv Report, arXiv: 1210.7559, 2012.

[3] A. Anandkumar, D.Hsu, S.M. Kakade. A Method of Moments for Mixture Models and Hidden Markov Models. ArXiv Report, arXiv: 1203.0683, 2012.

[4] D.M. Blei, A.Y. Ng, M.I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3:993-1022, 2003.

[5] T. Griffiths. Gibbs sampling in the generative model of Latent Dirichlet Allocation. Technical report, Stanford University, 2004.

[6] D. Hsu, S.M. Kakade. Learning Mixtures of Spherical Gaussians: Moment Methods and Spectral Decompositions. To appear in the 4th *Innovations in Theoretical Computer Science (ITCS)*, 2013. ArXiv Report, arXiv: 1206.5766.

[7] H.W. Kuhn. The Hungarian Method for the assignment problem . *Naval Researh Logistics Quarterly*, 3: 253-258, 1956.

[8] J. Rennie. 20Newsgroups dataset http://qwone.com/ jason/20Newsgroups/

# A Cross-Moments of Dirichlet-Distributed Vectors

If h is drawn from a Dirichlet distribution and the concentration parameters are known, then these moments could easily be calculated:

If $h \sim Dirichlet(\alpha_1, \alpha_2, .., \alpha_K)$, then

$$E[\mathbf{h}_i] = \frac{\alpha_i}{\alpha_0}$$

$$E[h \otimes h]_{i,j} = E[\mathbf{h}_i \mathbf{h}_j] = \begin{cases} \frac{\alpha_i \alpha_j}{\alpha_0(\alpha_0+1)}, & i \neq j \\ \frac{\alpha_i(\alpha_i-1)}{\alpha_0(\alpha_0+1)}, & i = j \end{cases}$$

$$E[h \otimes h \otimes h]_{i,j,k} = E[\mathbf{h}_i \mathbf{h}_j \mathbf{h}_k] = \begin{cases} \frac{\alpha_i \alpha_j \alpha_k}{\alpha_0(\alpha_0+1)(\alpha_0+2)}, & i, j, k \text{ distinct} \\ \frac{\alpha_i(\alpha_i+1)\alpha_j}{\alpha_0(\alpha_0+1)(\alpha_0+2)}, & i = j \neq k \\ \frac{\alpha_i(\alpha_i+1)(\alpha_i+2)}{\alpha_0(\alpha_0+1)(\alpha_0+2)}, & i = j = k \end{cases}$$

# B Derivation of Empirical Moment Estimators and Products

The equations listed below are needed for implicit calculation of power iteration just for completeness. In general they are useful for real implementation, but have nothing to do with theoretical guarantees.

## B.1 Notes:

For a specific document $\ell$ ($V_l$), $\sum_i^{|Voc|}, \sum_{i,j}^{|Voc|}$ can be re-written as $\sum_i^{|Voc_l|}, \sum_{i,j}^{|Voc_l|}$($Voc_l$ is the number of distinct word types used in document $\ell$), because we only need to care about words that occurred in document $\ell$ .

## B.2 Empirical First and Second Moments

$$\hat{M}_1 = \mathbf{E}[\mathbf{x}_1] = \frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{|V_l|}c_l$$

$$\hat{M}_2 = \mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2] - \frac{\alpha_0}{\alpha_0+1}\hat{M}_1 \otimes \hat{M}_1$$

$$= \frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{|V_l|(|V_l|-1)}(c_l \otimes c_l - diag(c_l)) - \frac{\alpha_0}{\alpha_0+1}\hat{M}_1 \otimes \hat{M}_1$$

## B.3 Empirical Third Moment and Its Multilinear Products

$$\hat{M}_3 = \mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2 \otimes \mathbf{x}_3] - \frac{\alpha_0}{\alpha_0+2}(\mathbf{E}[\mathbf{x}_1 \otimes \mathbf{x}_2 \otimes M_1] + \mathbf{E}[\mathbf{x}_1 \otimes M_1 \otimes \mathbf{x}_2] + \mathbf{E}[\hat{M}_1 \otimes \mathbf{x}_1 \otimes \mathbf{x}_2]) + \frac{\alpha_0^2}{(\alpha_0+1)(\alpha_0+2)}(\hat{M}_1 \otimes \hat{M}_1 \otimes \hat{M}_1)$$

$$= \frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)(|V_l|-2)}[c_l \otimes c_l \otimes c_l + 2\sum_i^{|Voc|} c_{li}(e_i \otimes \mathbf{e_i} \otimes \mathbf{e_i}) - \sum_{i,j}^{|Voc|} c_{li}c_{lj}(e_i \otimes \mathbf{e_i} \otimes e_j) - \sum_{i,j}^{|Voc|} c_{li}c_{lj}(e_i \otimes e_j \otimes \mathbf{e_i}) - \sum_{i,j}^{|Voc|} c_{li}c_{lj}(e_j \otimes \mathbf{e_i}$$

$$-\frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)}\frac{\alpha_0}{\alpha_0+2}[c_l \otimes c_l \otimes \hat{M}_1 + c_l \otimes \hat{M}_1 \otimes c_l + \hat{M}_1 \otimes c_l \otimes c_l - \sum_i^{|Voc|} c_{li}(\mathbf{e_i} \otimes \mathbf{e_i} \otimes \hat{M}_1 + \mathbf{e_i} \otimes \hat{M}_1 \otimes \mathbf{e_i} + \hat{M}_1 \otimes \mathbf{e_i} \otimes \mathbf{e_i})]$$

$$+\frac{2\alpha_0^2}{(\alpha_0+1)(\alpha_0+2)}(M_1 \otimes \hat{M}_1 \otimes \hat{M}_1)$$

$$\hat{M}_3(I,I,\eta) = \frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)(|V_l|-2)}[c_l \otimes c_l(\eta^T c_l) + 2diag(c_l \circ \eta) - c_l^T \eta diag(c_l) - (c_l \circ \eta) \otimes c_l - c_l \otimes (c_l \circ \eta)]$$

$$-\frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)}\frac{\alpha_0}{\alpha_0+2}[(c_l \otimes c_l - diag(c_l))(\eta^T M_1) + (c_l \otimes c_l - diag(c_l))\eta\hat{M}_1^T + \hat{M}_1\eta^T(c_l \otimes c_l - diag(c_l))]$$

$$+\frac{2\alpha_0^2}{(\alpha_0+1)(\alpha_0+2)}(\eta^T\hat{M}_1)(\hat{M}_1 \otimes \hat{M}_1)$$

$$\hat{M}_3(\hat{W},\hat{W},\eta) = \frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)(|V_l|-2)}$$

$$[(\hat{W}^T c_l) \otimes (\hat{W}^T c_l)(\eta^T c_l) + 2\hat{W}^T diag(c_l \circ \eta)\hat{W} - (c_l^T\eta)\hat{W}^T diag(c_l)\hat{W} - (\hat{W}^T(c_l \circ \eta)) \otimes (\hat{W}^T c_l) - (\hat{W}^T c_l) \otimes (\hat{W}^T(c_l \circ \eta))]$$

$$-\frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)}\frac{\alpha_0}{\alpha_0+2}[((\hat{W}^T c_l) \otimes (\hat{W}^T c_l) - \hat{W}^T diag(c_l)\hat{W})(\eta^T\hat{M}_1) + \hat{W}^T(c_l \otimes c_l - diag(c_l))\eta\hat{M}_1^T\hat{W} + \hat{W}^T M_1\eta^T(c_l \otimes c_l - diag$$

$$+\frac{2\alpha_0^2}{(\alpha_0+1)(\alpha_0+2)}(\eta^T\hat{M}_1)((\hat{W}^T\hat{M}_1)\otimes(\hat{W}^T\hat{M}_1))$$

$$\hat{M}_3(1,\eta,\eta)=\frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)(|V_l|-2)}[c_l(\eta^T c_l)^2+2(c_l\circ c_l\circ\eta)-2(c_l\circ\eta)(c_l^T\eta)-c_l(c_l^T(\eta\circ\eta))]$$

$$-\frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)}\frac{\alpha_0}{\alpha_0+2}[(\eta^T M_1)[(c_l^T\eta)c_l-(c_l\circ\eta)]+[c_l(c_l^T\eta)-(c_l\circ\eta)](M_1{}^T\eta)+\hat{M}_1(\eta^T c_l)^2-\hat{M}_1\eta^T(c_l\circ\eta)]$$

$$+\frac{2\alpha_0^2}{(\alpha_0+1)(\alpha_0+2)}(\eta^T\hat{M}_1)^2\hat{M}_1$$

$$\hat{M}_3(\eta,\eta,\eta)=\frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)(|V_l|-2)}[(\eta^T c_l)^3+2((\eta\circ\eta)^T(c_l\circ c_l))-3(c_l^T\eta)(c_l^T(\eta\circ\eta))]$$

$$-\frac{1}{|Docs|}\sum_{l=1}^{|Docs|}\frac{1}{(|V_l|)(|V_l|-1)}\frac{\alpha_0}{\alpha_0+2}[3(\eta^T M_1)[(c_l^T\eta)^2-c_l^T(\eta\circ\eta)]]+\frac{2\alpha_0^2}{(\alpha_0+1)(\alpha_0+2)}(\eta^T M_1)^3$$

# C Other Approaches for Tensor Decomposition

## C.1 Pseudo-Inverse Method[6]

Consider $G = (M_2)^{\dagger\frac{1}{2}} M_3(I, I, \eta)(M_2)^{\dagger\frac{1}{2}}$, suppose $(\sqrt{\frac{\alpha_1}{\alpha_0(\alpha_0+1)}}\phi_1, \ldots, \sqrt{\frac{\alpha_K}{\alpha_0(\alpha_0+1)}}\phi_K) = USV^T$ is the "thin" SVD, then

$$M_2 = US^2U^T, M_3(I, I, \eta) = \frac{2}{\alpha_0+2} USV^T diag(\eta^T\phi_1, \ldots, \eta^T\phi_K)VSU^T$$

$$G = \frac{2}{\alpha_0+2} UV^T diag(\eta^T\phi_1, \ldots, \eta^T\phi_K)VU^T$$

can be treated as a "thin" SVD. So if we do SVD in $G$ and get the singular vectors with respect to nonzero singular values, we can get $(r_1, \ldots, r_K) = UV^T$ up to column permutation and signs. Note that $M_2^{\frac{1}{2}} = USU^T$, then we calculate $USU^T(r_1, \ldots, r_K)$, which equals $USV^T$ up to column permutation and signs. Then we use

$$\frac{\lambda_k}{\eta^T(M_2)^{\frac{1}{2}}v_k} = \gamma_k$$

to recover $\alpha_k$ and

$$\frac{2}{\alpha_0+2}\frac{\lambda_k}{\eta^T(M_2)^{\frac{1}{2}}v_k}(M_2)^{\frac{1}{2}}v_k = \phi_k$$

to recover $\phi_k$. We can also use $M_1$ to refine the results of $\alpha_k$ as well, because $M_1 = \frac{1}{\alpha_0}(\phi_1, \ldots, \phi_K)\alpha$, so $\alpha = \alpha_0(\phi_1, \ldots, \phi_K)^\dagger M_1)$. Note this method may be computationally intractable if we explicitly calculate G.

## C.2 Eigenvector Method[3]

Suppose $U$ is the orthonormal base of $M_2$'s column space. (We can do SVD on $M_2$ to find U, or more generally, if we can get a $U$ whose column space is $M_2$'s column space (which is also $\Phi$'s column space), similar technique applies.)

Consider the following matrix:

$$(U^T M_3(I, I, \eta)U)(U^T M_2 U)^{-1}$$

$$= \frac{2}{\alpha_0(\alpha_0+1)(\alpha_0+2)}(U^T\Phi)diag(\alpha)diag(\Phi^T\eta)(U^T\Phi)^T(\frac{1}{\alpha_0(\alpha_0+1)}(U^T\Phi)diag(\alpha)(U^T\Phi)^T)^{-1}$$

$$= \frac{2}{\alpha_0+2}(U^T\Phi)diag(\Phi^T\eta)(U^T\Phi)^{-1}$$

If we extract the eigenvectors of $(U^T M_3(I, I, \eta)U)(U^T M_2 U)^{-1}$ as $(r_1, \ldots, r_K)$, we can see they are columns of $(U^T\Phi)$, up to column permutation and scaling, then $Ur_k = UU^T\phi_k = \phi_k$.

Note that in this method we cannot directly recover $\alpha$, and we can only normalize $\phi_k$ explicitly, because the eigenvectors have one degree of freedom in scaling.

## C.3 Eigenvalue Method Using Simultaneous Diagonalization[3]

Same as the eigenvectors approach, but we consider eigenvalues instead of eigenvectors. Note that if we do diagonalization with different values of $\eta$, we can get different $(\Phi^T\eta)$. To randomly choose $\eta_k, k = 1, \ldots, K$, for simplicity, we choose $\theta_k, k = 1, \ldots, K$ uniformly on unit sphere, then obtain $\eta_k = U\theta_k$. Denote $\Theta = (\theta_1, \ldots, \theta_K)^T$.

Then we observe $K$ vectors $\Phi^T\eta_k = t_k$, denoted as $L_k$. Note that $\phi_k = Uc_k$, so we can get $c_k$, because:

$$\begin{pmatrix}\eta_1^T \\ \ldots \\ \eta_K^T\end{pmatrix}(\phi_1, \ldots, \phi_K) = \begin{pmatrix}\theta_1^T \\ \ldots \\ \theta_K^T\end{pmatrix}U^T U(c_1, \ldots, c_K) = L$$

$$(c_1, \ldots, c_K) = \Theta^{-1}L$$

$$(\phi_1, \ldots, \phi_K) = U(c_1, \ldots, c_K) = U\Theta^{-1}L$$

A subtle issue is that we must diagonalize these matrices simultaneously. For example, if we deal with empirical moments, we can use one single $P$ which diagonalizes $(U^T M_3(I, I, U\theta_1)U)(U^T M_2 U)^{-1}$, then although $P^{-1}(U^T M_3(I, I, U\theta_k)U)(U^T M_2 U)^{-1}P$ are not perfectly diagonal for other $\theta_k, i \geq 2$, we still consider their diagonal elements. Note that if we diagonalize them individually, the order of the eigenvalues will be shuffled for each individual matrix, so that we cannot safely recover the $\phi_k$.

# D Partitioning of 20Newsgroups

20Newsgroups was partitioned into six classes, following [8]:

| | | |
|---|---|---|
| comp.graphics<br>comp.os.ms-windows.misc<br>comp.sys.ibm.pc.hardware<br>comp.sys.mac.hardware<br>comp.windows.x | rec.autos<br>rec.motorcycles<br>rec.sport.baseball<br>rec.sport.hockey | sci.crypt<br>sci.electronics<br>sci.med<br>sci.space |
| misc.forsale | talk.politics.misc<br>talk.politics.guns<br>talk.politics.mideast | talk.religion.misc<br>alt.atheism<br>soc.religion.christian |

# E Illustrative results

The results of simultaneous power method, ECA and tensor deflation seem very similiar, so we present only the results of simultaneous power method here. The following tables show results of the method with different number of topics k.

## E.1 Classic4

- k = 10

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| inform | system | flow | case |
| librari | program | layer | system |
| system | comput | boundari | result |
| comput | languag | pressur | method |
| data | method | number | flow |
| scienc | gener | heat | present |
| studi | problem | solut | problem |
| user | data | equat | time |
| research | algorithm | mach | studi |
| method | present | theori | patient |
| servic | design | present | algorithm |
| retriev | time | bodi | effect |
| develop | structur | shock | solut |
| program | develop | transfer | bodi |
| search | paper | effect | growth |
| base | techniqu | result | model |
| book | discuss | method | techniqu |
| index | function | laminar | obtain |
| process | equat | plate | cell |
| oper | oper | wave | develop |

While 3 of the natural topics are recovered, MED is barely present. Topic 4 is a mixture of topics.

- k = 20

| 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|
| cell | flow | librari | algorithm | languag | number |
| structur | boundari | inform | method | program | librari |
| studi | layer | book | program | comput | problem |
| data | heat | studi | time | gener | list |
| scienc | number | research | present | problem | bodi |
| activ | solut | journal | data | sort | journal |
| line | plate | public | number | fortran | titl |
| patient | effect | univers | paper | system | creep |
| bodi | equat | develop | system | algorithm | catalog |
| relat | theori | academ | result | list | method |
| high | transfer | librarian | inform | structur | buckl |
| normal | laminar | system | set | string | function |
| marrow | problem | report | problem | featur | scienc |
| languag | compress | work | tabl | present | librarian |
| bone | point | catalog | languag | translat | point |
| type | pressur | cost | structur | process | work |
| growth | surfac | present | bodi | user | column |
| rat | dimension | decis | oper | design | time |
| tissu | veloc | scienc | flow | rule | boundari |
| strain | case | paper | gener | file | period |

We chose to display 6 of the 20 topics. We see from columns 1-4 that all natural topics are recovered, but we also get duplicate topics as in column 5 and mixed topics as in column 6.

## E.2 20Newsgroups

For 20Newsgroups, k=10 and k=20 produced fairly similar results.

- k = 10

| 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|
| drive | window | game | kei | god | mail |
| disk | run | team | chip | christian | list |
| hard | problem | win | bit | jesu | sale |
| floppi | card | two | order | point | address |
| system | file | plai | encrypt | post | window |
| file | applic | player | clipper | question | phone |
| do | monitor | come | phone | mean | post |
| format | video | run | secur | exist | file |
| scsi | mail | score | gun | church | run |
| control | manag | season | govern | christ | email |
| problem | system | cub | simm | jew | info |
| question | line | last | escrow | bibl | interest |
| comput | program | hockei | number | find | send |
| set | screen | dai | public | law | question |
| softwar | do | seri | de | group | number |
| compress | font | world | nsa | religion | do |
| switch | color | suck | two | state | read |
| bit | question | tie | mean | show | group |
| origin | driver | record | run | word | advanc |
| copi | set | put | call | answer | back |

- k = 20

| 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|
| game | window | god | drive | mail | card | car |
| team | problem | christian | disk | list | video | file |
| run | run | jesu | hard | sale | color | problem |
| win | system | call | floppi | run | vga | question |
| two | applic | mean | system | address | driver | two |
| plai | driver | christ | scsi | interest | window | bike |
| player | manag | car | format | info | mail | monitor |
| come | do | read | do | group | graphic | bui |
| last | color | irq | question | phone | monitor | post |
| window | file | live | file | post | mode | last |
| score | video | bibl | control | chip | cach | opinion |
| kei | mous | group | softwar | type | bui | god |
| file | monitor | religion | set | send | speed | dai |
| season | win | church | sale | file | address | gener |
| seri | graphic | doesn | origin | advanc | phone | ride |
| system | font | sin | come | question | set | road |
| hockei | program | sound | mac | read | problem | didn |
| name | screen | post | power | inform | number | great |
| into | set | never | boot | problem | list | mac |
| sound | card | love | program | email | fpu | softwar |