Weighted Tensor Decomposition for Learning Latent Variables with Partial Data

Omer Gottesman
Paulson School of Engineering
and Applied Sciences,
Harvard University

Weiewei Pan
Paulson School of Engineering
and Applied Sciences,
Harvard University

Finale Doshi-Velez
Paulson School of Engineering
and Applied Sciences,
Harvard University

Abstract

Tensor decomposition methods are popular tools for learning latent variables given only lower-order moments of the data. However, the standard assumption is that we have sufficient data to estimate these moments to high accuracy. In this work, we consider the case in which certain dimensions of the data are not always observed—common in applied settings, where not all measurements may be taken for all observations—resulting in moment estimates of varying quality. We derive a weighted tensor decomposition approach that is computationally as efficient as the non-weighted approach, and demonstrate that it outperforms methods that do not appropriately leverage these less-observed dimensions.

1 Introduction

Tensor decomposition methods decompose low-order moments of observed data to infer the latent variables from which the data is generated [Anandkumar et al., 2014, Arora et al., 2012]. They have recently gained popularity because they require only moments of the data and, in the absence of model-mismatch, come with global optimality guarantees. Of course, these guarantees require that the moments are well-estimated—which corresponds to the amount of available data. Previous works have investigated the dependence of the inference quality on the amount of collected data (and the related uncertainty in the moment estimates) [Anandkumar et al., 2014, Kolda and Bader, 2009, Podosinnikova et al., 2015]. However, these works typically assume that every dimension is observed in every datum.

In this work, we consider the scenario in which some dimen-

sions are always observed, while others may be available for only a few observations. Such situations naturally arise in real world applications. In clinical settings, for example, easily obtained measurements such as temperature and blood pressure may be available for all patients, but the results of a blood test may be available for only a small number of patients. Similarly, billing codes may be available for all patients, while information extracted from unstructured notes only for a subset. In biology applications, one may want to combine counts from whole genome sequencing from a smaller study with counts of the subset of SNPs from a larger population.

Let us suppose that we are only interested in latent variable models over the dimensions that are completely observed. In the medical setting example we presented earlier, that would correspond to learning a model to explain only the patients' temperature and blood pressure data. In such a case, would using the data on other vitals be helpful or detrimental to learning? We demonstrate that depending on the frequency at which a dimension is missing in the data, including its associated estimated moments may help or hinder the estimation of latent variable model parameters for the rest of the model.

Of course, the more interesting question is whether we can do better than sometimes ignoring dimensions with missingness—that is, can the lower quality moment estimates from these incompletely-observed dimensions still be used to improve our inference about the structure of the complete dimensions? We introduce a weighted tensor decomposition method (WTPM) which weights elements in the moment estimates proportionally to the frequency at which they are observed. It has identical computation requirements to the unweighted approach, and performs as well as or better than both unweighted approaches, i.e. including or ignoring the dimensions which are sometimes missing. We present experimental results for two commonly used latent variable models: Gaussian mixtures and the Gamma-Poisson model.

2 Background and Setting

2.1 Tensor decomposition methods

The principle at the heart of learning latent variable models via tensor decomposition is simple: the moments of many such models can be written as a tensor decomposition of the models parameters in the form

$$\mathbf{S} = \sum_k s_k \mathbf{a}_k \mathbf{a}_k^T, \tag{1}$$

$$\mathbf{T} = \sum_k t_k \mathbf{a}_k \otimes \mathbf{a}_k \otimes \mathbf{a}_k. \tag{2}$$

Here, $\mathbf{a}_k$ denotes columns of the parameter matrix $\mathbf{A} \in \mathbb{R}^{D \times K}$, representing the kth latent parameter in a D dimensional space. $\mathbf{A}$ can then be inferred from the data - denoted for a dataset of N samples by the matrix $\mathbf{X} \in \mathbb{R}^{D \times N}$ - by computing the empirical estimates of the moments $\hat{\mathbf{S}}(\mathbf{X})$ and $\hat{\mathbf{T}}(\mathbf{X})$, and calculating their decomposition. The procedure is quite general and can be applied whenever the expression for the empirical estimates of the moments can be derived and computed from the data. In this work we present results for mixtures of spherical Gaussians [Hsu and Kakade, 2013] and the Gamma-Poisson (GP) model [Podosinnikova et al., 2015]. A description of the generative models and the form of the empirical estimates can be found in appendix A.

Several methods exist to estimate the decomposition of the tensor estimates [Hyv et al., 1999, Cardoso and Souloumiac, 1993, Anandkumar et al., 2014]. In this work, we use the method described in Anandkumar et al. [2014], which provides a polytime algorithm to learn the parameters from the moment estimates. The method consists of two stages. In the first stage the eigenstructure of $\hat{\mathbf{S}}$ is used to compute a whitening matrix, $\mathbf{W}$, which is used to reduce $\hat{\mathbf{T}}$ to a $K \times K \times K$ tensor with an orthogonal decomposition, $\hat{\mathbf{T}}_c$. In the second stage, the tensor power method (TPM) - the tensor analog of the matrix power method - is used to calculate the eigenvectors of $\hat{\mathbf{T}}_c$, which can be transformed into the columns of the parameter matrix $\hat{\mathbf{A}}$ using $\mathbf{W}$ once again. (For more details see Anandkumar et al. [2014]). We emphasize that our results are not specific to this choice of decomposition method.

2.2 Missingness

In the following, we assume that there exist a certain set of *complete dimensions* that are always recorded in every observation. It is the relationships between these dimensions that we are most interested in recovering. The remaining *incomplete dimensions* are missing completely-at-random with probability $(1 - p_d)$. That is, the probability that one of these dimensions is not recorded in observation n is independent of all other dimensions in $\mathbf{x}_n$, the data set $\mathbf{X}$, and the specific value of x_{nd}. A schematic illustration of such a dataset is presented in Figure 1 (a). To calculate
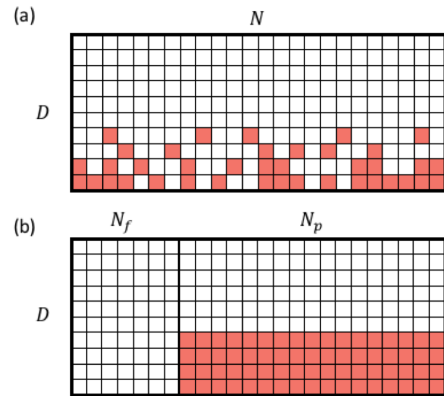

Figure 1: (a) Basic uncorrelated structure of missing data. Colored elements represent missing data elements which are not observed. Specifically, every dimension d^{th} in a sampled data point x_n (column n in $\mathbf{X}$) is present with some probability p_d, independently of the missingness of all other elements in the data. (b) Correlated missing data structure in the toy example.

p_d empirically we simply count the number of samples in which a given dimension is observed, and divide by the number of samples, N.

3 The Value of Incomplete Data: Motivating Example

In Section 1, we asked a simple question: if some of our observations are incomplete—that is, we have observed only a few of the possible dimensions—then is it better to just discard them? Or are there instances in which the additional poorly-estimated moments are valuable for recovering parameters associated with the complete dimensions? In this section, we demonstrate that the answer depends on *how* poorly estimated those additional moments are—there are regimes in which the additional moments only add noise, and regimes in which they assist with the overall parameter recovery.

We consider a very simple missingness pattern for the purpose of illustration, shown schematically in Figure 1 (b). Specifically, we consider a data set that consists of N_f full observations, for which all dimensions are observed, and N_p partial observations, for which some dimensions are never observed. (Note that this structure is not missing-at-random, which we will require for our derivations of Lemma 4.1, but makes it simpler to illustrate the idea.)

Our metric for the reconstruction error of the parameter matrix $\mathbf{A}$ incorporates *only* the complete dimensions:

$$\varepsilon_c \equiv \sum_k \frac{\hat{\mathbf{a}}_{c,k} \cdot \mathbf{a}_{c,k}}{||\hat{\mathbf{a}}_{c,k}|| \cdot ||\mathbf{a}_{c,k}||}, \tag{3}$$

where $\mathbf{a}_{c,k}$ is the vector comprised only of the *complete*
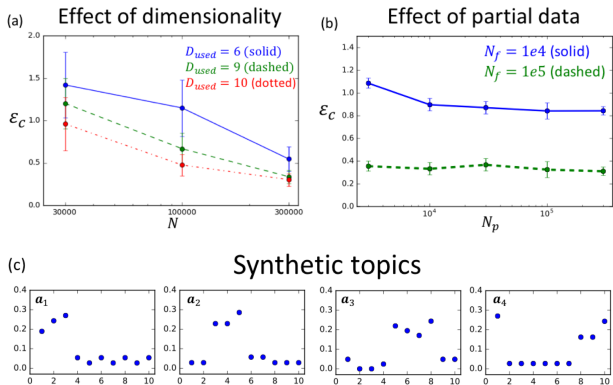
Omer Gottesman, Weiewei Pan, Finale Doshi-Velez

Figure 2: Effect of dimensionality and additional partially-observed data. In (a), the additional correlations by observing additional dimensions reduces the reconstruction error on the six dimensions of interest, for any number of samples N. In (b), increasing the number of partial samples (N_p), for which 4 out of $D = 10$ elements are missing, has a limited effect on reconstruction error. Each data point is an average over 20 runs in which the data was sampled independently, and error-bars represent the standard deviation of the inference errors for each run. (c) Structure of $\mathbf{A}$ in the synthetic GP example. The topics elements a_{dk} are plotted as a function of d.

Figure 3: Reconstruction error for complete dimensions, ε_c, vs. N_p. For a small number of partial samples, it is advantageous to perform the inference using all dimensions of the data (full dimensionality method - blue), whereas when there is a large amount of partial data it is best to perform the tensor decomposition on the lower dimensionality problem (partial dimensionality method - green). $N_f = 1e4$. The schematics illustrate both methods, with the bold box indicating the elements of the second order tensor being analyzed, and red boxes represent elements associated with the incomplete dimensions. Each data point is an average over 20 runs.

elements in the kth latent parameter. We measure the reconstruction error as the sum of angles in high dimensional space between the true and inferred parameters in order to avoid ambiguities due to scaling and normalization. (In particular, the columns of $\mathbf{A}$ in the GP model are normalized to sum to one, so it is necessary to choose a measure of error which is independent of normalization.)

In our illustration, we use the GP model and consider the recovery of a single topics matrix $\hat{\mathbf{A}} \in \mathbb{R}^{10 \times 4}$, shown in Figure 2 (c) (In Appendix D, we demonstrate that our results also hold across a large number of randomly generated topic matrices $\mathbf{A}$).

We consider two options for recovering the parameters corresponding to the complete dimensions, $\mathbf{A}_c$. The first, which we call the *full dimensionality method*, makes use of all the data available. That is, if certain dimensions of an observation $\mathbf{x}_n$ are missing, we still use the non-missing dimensions to assist in estimating the tensors $\hat{\mathbf{S}}$ and $\hat{\mathbf{T}}$. (See the schematic insets in Figure 3 to see what elements of the second order tensor we take into account when performing the decomposition). Once we have learned the topics for all dimensions, we can observe only the complete dimensions which are of interest.

The full dimensionality method results in decomposing tensors in which each element may have a different uncertainty. In contrast, the second approach—which we call the *partial dimensionality method*—discards any dimensions that are
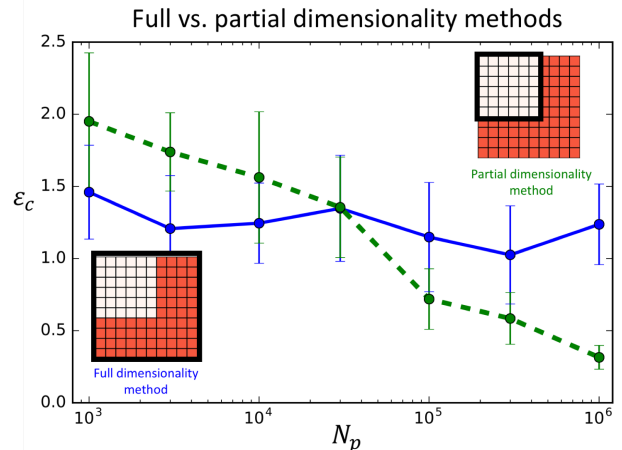
not always present in the data and performs the inference on the lower dimensional problem consisting of only the elements which are always observed. Tensor elements in the partial dimensionality method will have equal uncertainties, but will include fewer correlations.

In Figure 2, we provide some intuition for the factors affecting the performance of the two methods. In Figure 2 (a) we illustrate the effect of using only a subset of the dimensions for inference. Our goal is to infer the topic structure of six out of the ten dimensions in the data. To perform such inference, we must use the dimensions whose structure we try to infer, but may ignore some of the other dimensions. Figure 2 (a) demonstrates that for a given number of samples, ignoring the extra dimensions leads to larger inference error, as important information about the structure of the data is ignored.

To contrast the simple case where no data is missing, in Figure 2 (b), we demonstrate the effect of partial data on the topics recovery error. We present the reconstruction error using the full dimensionality method where for N_f full samples all the dimensions in the data are observed, and for N_p partial samples only 6 of the 10 dimensions are observed in the data. These partial data allow us to better estimate the moments associated only with those 6 dimensions, but the error in the remaining moments is not

reduced by these additional observations. The relatively flat lines suggest that the recovery error is dominated by the moments with the highest uncertainty—something that the partial data cannot fix—and therefore collecting additional partial data does little to improve the inference of the full dimensionality method.

Finally, in Figure 3, we plot the reconstruction error as a function of N_p at a constant N_f for both the full dimensionality and the partial dimensionality methods. As seen in Figure 2 (b), additional partial samples do not help the full dimensionality method, which is dominated by the least accurate tensor elements. In contrast, with the partial dimensionality method, the topics reconstruction error steadily decreases, as the estimates $\hat{S}$ and $\hat{T}$ converge to their true value (and as long as that subset of the true topics matrix A associated with those dimensions satisfies the incoherence requirements in Anandkumar et al. [2014], $\hat{A}_c$ will converge to A_c). However, for small amounts of partial data, because we are effectively working on data with lower dimensionality, our reconstruction error is worse compared to the results of the full dimensionality model—or put another way, when all the tensor elements have similar uncertainties, it makes sense to use all the correlations that we have available. The existence of this cross-over suggests an opportunity for an algorithm that can attain the better recovery in all cases. We derive such an algorithm in Section 4.

4 The Weighted Tensor Power Method (WTPM)

In Section 3, we demonstrated that the choice of whether to use the dimensions with missing data depends on the *relative* quality of the moment estimates. In section 3 we gave an intuition for why if the dimensions with missing data did not have *significantly* less data than the dimensions with complete data, it would make sense to include those moments in order to get a more complete correlation structure of the data. However, since the quality of the inference is dominated by the noisiest tensor elements, it does not make sense to include tensor elements that may be of significantly lower quality—as would be the case if the dimensions with missing data had significantly less data associated with them than the complete dimensions.

In this section we introduce the *Weighted Tensor Power Method* (WTPM) for spectral learning with missing data. It is more general than the motivating example in Section 3 in that we allow different dimensions to have different levels of missingness, but we shall also need to require that each dimension is missing at random. (This assumption is required for our derivation. We will show that our approach is robust to other missingness structures in Section 5.1). The key idea behind our approach is the intuition that we would like to use all the moments of the data, but assign higher importance to moments for which we have smaller uncer-

tainties. A natural and intuitive way to perform this task is by performing tensor decomposition on weighted tensors. The weighting consists of multiplying every element in the empirical tensor estimates, $\hat{S}$ and $\hat{T}$, by some weight which represents the importance we give that specific element. Thus, the elements of the weighted tensors, $\hat{S}^*$ and $\hat{T}^*$, are given by

$$\hat{S}^*_{d_1 d_2} = w_{d_1 d_2} \hat{S}_{d_1 d_2} \tag{4}$$

$$\hat{T}^*_{d_1 d_2 d_3} = w_{d_1 d_2 d_3} \hat{T}_{d_1 d_2 d_3}. \tag{5}$$

To apply the efficient algorithms of standard tensor decomposition methods, the weighting scheme must also maintain the structure of the tensors as sums of K rank 1 tensors, similar to moment equations (1) and (2). To maintain such a structure, we write the weights as a product of the weights associated with every dimension contributing to the tensor:

$$\begin{aligned} w_{d_1 d_2} &= w_{d_1} w_{d_2} \\ w_{d_1 d_2 d_3} &= w_{d_1} w_{d_2} w_{d_3}. \end{aligned} \tag{6}$$

This weighting scheme allows us to rewrite the moments as

$$\begin{aligned} S^*_{d_1 d_2} = w_{d_1 d_2} S_{d_1 d_2} &= \sum_k s_k w_{d_1 d_2} a_{d_1 k} a_{d_2 k} \\ &= \sum_k s_k (w_{d_1} a_{d_1 k})(w_{d_2} a_{d_2 k}) = \sum_k s_k a^*_{d_1 k} a^*_{d_2 k}. \end{aligned}$$

where we defined $a^*_{dk} = a_{dk} w_d$. An analogous expression can be written for the third-order tensor T^*. Written in this form, it is clear that the weighted tensors also have the structure of a sum of K rank 1 tensors, and therefore can be decomposed using standard tensor decomposition methods.

This form of the weights enjoys the additional advantage of allowing us to view the weighting of the moments tensors as a rescaling of the dimensions in the topics and data, $x^*_{dn} = x_{dn} w_d$ and $a^*_{dk} = a_{dk} w_d$, providing an intuitive explanation for the effect of weighting the moments. This view also allows us to perform the weighting on the data itself, multiplying each dimension by w_i, rather than on the moments.

Finally, once we perform the decomposition and learn the rescaled topic matrix, we can rescale this matrix back by dividing its dth row by w_d to reconstruct the *entire* topic matrix, and not just the part corresponding to the complete dimensions. However, this full reconstruction is very susceptible to noise as error in reconstructing the incomplete dimensions will be amplified when dividing by the small weights associated with them.

Running Time A consequence of viewing the weighting as a rescaling of the data is that the WTPM has the same running time as a standard tensor decomposition method, as it can be implemented by scaling each dimension of the data by w_d and then applying one's tensor decomposition method of choice.

Choice of Weights The question remains of what weights w_d to assign to each dimension. Our approach is to choose weights that minimize the inference error. In this way, optimal weights can be computed for any model for which one can derive expressions for the inference error as a function of moment estimations errors and any other parameters affected by the weighting.

Demonstrating our approach, we first present the choice of optimal weights for the Gamma-Poisson model. Podosinnikova et al. [2015] drew and expanded on sample complexity bounds results from Anandkumar et al. [2012] to show that the inference error has two main contributions, one which scales as $\mathbb{E}\left[||\hat{\mathbf{S}} - \mathbf{S}||_F\right]/L^2$, and one which scales as $\mathbb{E}\left[||\hat{\mathbf{T}} - \mathbf{T}||_F\right]/L^3$. By choosing our weights to minimize these two ratios we can derive the weights which minimize the topics reconstruction error:

Lemma 4.1. *Assume that the magnitude of all elements in* **A** *are comparable, and that the probabilities of elements in the data to be missing are independent of each other. For high dimensional problems, an intuitive upper bound $\phi(\mathbf{w})$, of the expected topic reconstruction error, E_{infer}, under the GP model is locally minimized by setting:*

$$w_d \propto p_d,$$

where p_d is the probability of an element in the dth dimension to be observed in the data.

Proof. The upper bound $\phi(\mathbf{w})$ on E_{infer} is essentially the sum of the Jensen gaps of the expected sampling errors, $\mathbb{E}\left[||\hat{\mathbf{S}} - \mathbf{S}||_F\right]$ and $\mathbb{E}\left[||\hat{\mathbf{T}} - \mathbf{T}||_F\right]$.

See Appendix E for details.

An immediate corollary of Lemma 4.1 is that weighting the dimensions according to degrees of missingness is provably better for inference than the unweighted model. In Appendix E, we show that under stronger assumptions, where we neglect the structure in the data, we can obtain globally optimal weighting by the choice $w_d \propto \sqrt{p_d}$. But in experimentation, we find the performance of both to be indistinguishable.

We note that in our actual WTPM we choose not to rescale the complete dimensions, thus we simply set the proportionality constant to 1 and use the scaling $w_d = p_d$. In practice, we find that choosing the weights according to Lemma 4.1 yields good results even when there is a large variance in the magnitude of the elements of **A**.

Additionally, we find that the weighting proposed in Lemma 4.1 leads to good inference results for most models, including Gaussian mixtures. Thus, in many cases, one may forgo derivations involving analytical expressions for the inference error in favor of the the heuristic of choosing

$w_d = p_d$. For example, while inference error bounds (in terms of the weights) exist for Gaussian Mixtures, the necessary conditions for these bounds to hold render optimization intractable.

Finally, we note that the mixture of Gaussians also requires an estimation of the variance, and reweighing dimensions will change a spherical Gaussian into an elliptical one. We provide details on how the weighting affects the variance estimation process in appendix C.

5 Results

In this section we compare our WTPM with the full and partial dimensionality methods described in Section 3. We test our method on synthetic, semi-synthetic, and real data sets for different structures of missing data and show that in all cases the WTPM interpolates between the more naive methods, always performing as well as or better than the best of them.

5.1 Synthetic Examples

Gaussian Mixtures Figure 4 (left) compares the WTPM with the full and partial dimensionality methods for synthetic data of Gaussian mixtures with $D = 10$, $K = 4$ and 4 missing dimensions. The inference error for the complete dimensions, ε_c is plotted vs. the number of samples, N. Each sub-figure shows the results for a different missingness pattern, given as a vector, $\mathbf{p}$, representing the probability of each dimension to be present in the data. The first six dimensions in each instance are complete dimensions and are always observed ($p = 1$), and the last four incomplete dimensions are present in the data with different probabilities. The centers of all Gaussians were sampled from a normal distribution $A_{ij} \sim \mathcal{N}(0, 100)$ for all (i, j), and the variance of the spherical Gaussians is $\sigma^2 = 100$. $\boldsymbol{\pi}$ was sampled from a Dirichlet distribution, $\boldsymbol{\pi} \sim \mathrm{Dir}(\mathbf{1})$. In all cases, the WTPM always did as well as or better than both the full and partial dimensionality method. We emphasize that because in some instances the partial dimensionality method outperforms the full dimensionality method and in some instances the opposite is true, always matching the best of the two methods with one algorithm is already advantageous, even when the WTPM does not outperform the two other methods.

Gamma-Poisson Next, we investigate the performance of our WTPM and compare it to the performance of the full and partial dimensionality methods for data generated from the topics presented in Figure 2. In Figure 4 (right) we present results for experiments on four different choices of missingness patterns, given by the vector $\mathbf{p}$. When the incomplete dimensions are missing with high probabilities (small p for the incomplete elements, Figure 4 (a-b)), the moments including the incomplete dimensions are very noisy

and including them in the inference by using the full dimensionality method (blue) leads to a higher reconstruction error compared with the partial dimensionality method (green). The opposite is true when the incomplete dimensions are missing with low probability (large p for the incomplete elements, Figure 4 (d)) and the full dimensionality method performs better by using the additional correlations which are less noisy in this case.

As with the Gaussian mixtures, the WTPM (red) does as well as or better than the best of the two baselines in all cases. In cases where values of p for the incomplete dimension are distributed over a wide range of values (Figure 4 (c)), the WTPM significantly outperforms both other methods, as it optimizes the amount of information in every moment, while properly down-weighting each moment according to its uncertainty.

Finally, in Figure 5(a), we present results based on the missingness pattern in Figure 1(b) and the synthetic model in Figure 3, where all partial dimensions are either present or missing together. This missingness violates the assumptions required in Lemma 4.1, the WTPM is robust to this violation and outperforms both full and partial dimensionality methods in this case as well.

5.2 Semi-synthetic Examples: 20 Newsgroups and Autism Comorbidities

To investigate the effect of incomplete dimensions within the kinds of sparse topic structure found in real data, we adopt a similar approach to Arora et al. [2013] and run still carefully-controlled experiments on semi-synthetic data. We use two real data sets to create semi-synthetic data for the WTPM: word counts from 4 categories of the 20-Newsgroups collection of news documents [Mitchell, 1997], filtered for words which appear in at least 5% of the documents ($D = 175$, $K = 4$, original $N = 3,652$); and counts of 64 common diagnoses from a collection electronic health records of children with autism at ages 5 and 10 [Doshi-Velez et al., 2014] ($D = 128$, $K = 5$, original $N = 5,475$).

To create the semi-synthetic data from these data, we first run a standard tensor decomposition on a real dataset to learn the parameters of a GP model ($\mathbf{D}$, $\mathbf{c}$ and b), mimicking the sparse data structure common to real world applications. To create the missing data structure, we first chose a set of missing dimensions for each data set. For the ASD data we chose the 64 dimensions corresponding to observations of symptoms at age 5. For the 20-Newsgroups data we randomly chose 116 out of the 175 dimensions. For each one of the incomplete dimensions, p_d was sampled uniformly between 0 to 0.9, resulting in a complicated structure of incomplete dimensions, with some dimensions missing in an insignificant amount of data, and some dimensions nearly completely unobserved. We emphasize that while the original data sets used to learn the topics are small, the semi-synthetic data set is large, allowing us to test the performance of our algorithm for a large amount of data.

Figure 5 compares the WTPM with the full and partial dimensionality methods for the semi-synthetic data and a random structure of missingness in the data. The WTPM performs as well or better than both the full and partial dimensionality method across the entire wide range of missingness patterns tested.

5.3 Real Data Example: Sepsis ICU data

As a final demonstration of the WTPM, we use the method to learn the structure of data on sepsis patients collected in ICUs. The data is obtained from the Medical Information Mart for Intensive Care III (MIMIC-III) database [Johnson et al., 2017]. The database contains more than 18,000 patients, and every patient's vitals were sampled on average at 14 different times, resulting in about 250,000 samples. Each sample is represented as a 47 dimensional vector containing information on a patient such as the vitals, lab results, demographics, etc. We wish to learn the structure of the data by modeling it as a mixture of Gaussians. The data collected in the ICU is often incomplete, and the missingness rate for different dimensions ranges from 0 to 0.75, making it an ideal dataset to test with our WTPM. We define as complete dimensions all the dimensions whose missingness rate is smaller than 0.05, resulting in 24 complete dimensions. We learn a mixture of Gaussian model for the complete dimensions with $K = 6$ topics using the full and partial dimensionality methods, as well the WTPM.

In Figure 6 we demonstrate that the WTPM performs as well as the full dimensionality method, which performs significantly better than the partial dimensionality method. The WTPM performs well despite the noisiness of the data: the missingness rate spans a wide range of values across different dimensions and is non-zero even for most of the dimensions we define as complete. Furthermore, given the nature of data collection in the ICU, we believe the missingness pattern is likely to be strongly correlated; the WTPM is robust to these assumption violations.

6 Discussion

The derivation of the ideal weights presented in Lemma 4.1 assumes the missingness of different elements is uncorrelated, and also ignores the structure of the topics matrix. However, we have demonstrated that the WTPM is robust to violations of these assumptions: Figure 5 demonstrates that the WTPM method works with a perfectly correlated missingness pattern (an element is missing in a sample if and only if all other incomplete dimensions in the sample are missing). The effectiveness of the WTPM on the semi-synthetic data generated from sparse topics matrices with significant magnitude differences between their non-zero
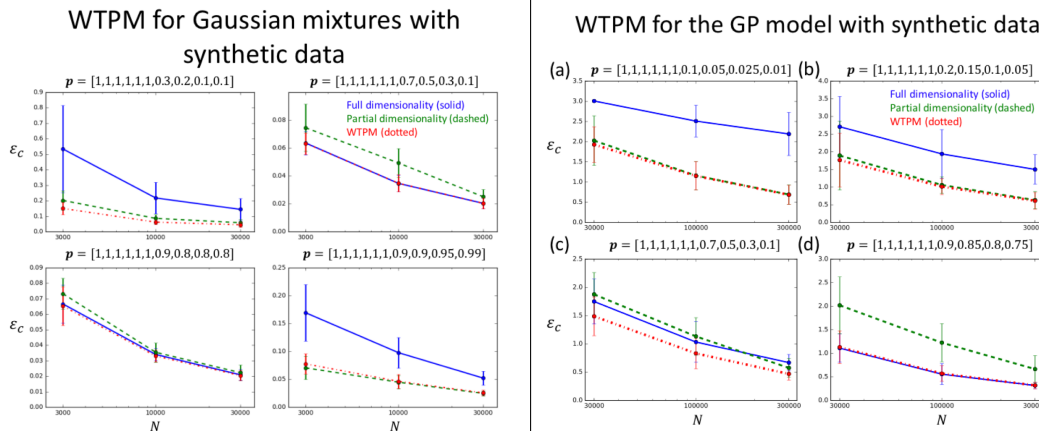
Omer Gottesman, Weiewei Pan, Finale Doshi-Velez

Figure 4: Reconstruction error for the complete dimensions of b synthetic data, ε_c, vs. N, where every incomplete dimension has a different probability, p_d, to be observed for a mixture of Gaussians model (left) and gamma-Poisson model (right). Each data point is an average over 20 runs. Whether the partial or full dimensionality method does better depends on the missingness rate; our WTPM *always* performs as well as the better of the baselines.
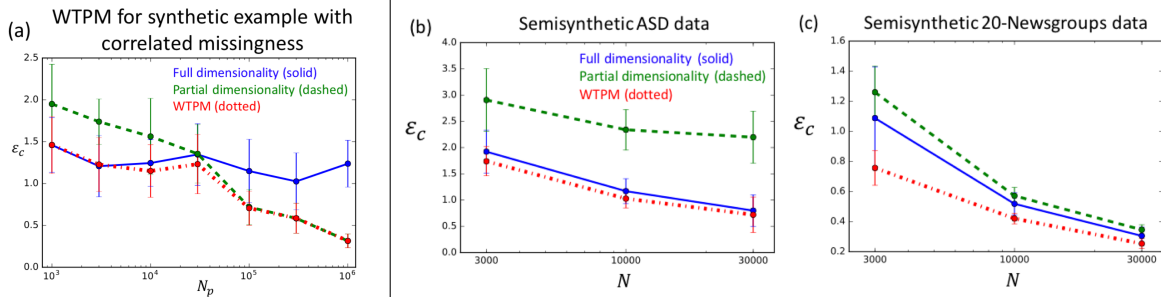

Figure 5: (a) Comparison of the WTPM with the full and partial dimensionality methods for synthetic topics with correlated missingness. Despite the violation of the missingness independence assumption, the WTPM still performs as well as the better of the two baselines. (b-c) Reconstruction error for complete dimensions of semi-synthetic data, ε_c, vs. N, in cases where every incomplete dimension has a different probability, p_d, to be observed. p_d is sampled uniformly from $[0, 0.9]$ for 64 out of the 128 dimensions in the ASD data (left) and for 116 out of the 175 dimensions in the 20-Newsgroups data (right).

elements demonstrates the insensitivity to the structure of the topics matrix. In future work, it would be interesting to check whether the effectiveness of the WTPM can be further improved by taking into account the structure of the topics and correlations in the missingness. For example, taking into account the structure of the data could mean that it would be advantageous to weight differently dimensions which on average have higher values compared with other dimensions.

While we focused on Gaussian mixtures and the gamma-Poisson model in this work, the core idea behind the WTPM can be applied to method-of-moment-based inference for other models as well. The form of the sample complexity bounds for LDA [Anandkumar et al., 2012], HMM [Hsu et al., 2012] and other topic models such as correlated topic models and Pachinko allocation [Arora et al., 2013, Blei and Lafferty, 2007, Li and McCallum, 2006] are quite similar to the bounds calculated by Podosinnikova et al. [2015], and

we therefore expect the application of the WTPM to these models to only require a derivation similar to that presented in Appendix E.

Another interesting direction is asking what we can say about the parameters of the dimensions that are incompletely observed. In this work, we demonstrated that lower-quality moment estimates from the incomplete dimensions could assist with the recovery of the parameters associated with the complete dimensions. The WTPM approach does give us values for all of the parameters—including ones associated with incomplete dimensions. These parameters can be straightforwardly up-weighted by $\frac{1}{w_d}$ to get parameter-values for the original data scaling. However, if those parameters are poorly estimated—which is likely, if their associated moment estimates were poor—this rescaling will not result in smaller errors compared with a standard tensor decomposition approach. It is an open question if it is possible to get better recovery of these parameters as well.
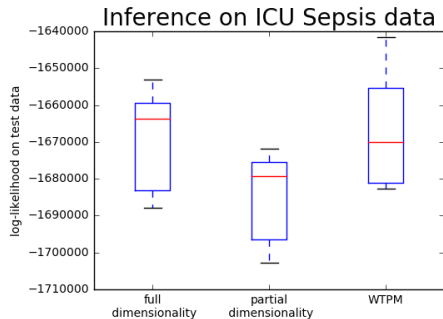
Figure 6: Log-likelihood of held-out test data for the full and partial dimensionality methods, as well as the WTPM. The box plot represents results from 20 different partitioning of the data in training and test data sets, where 0.2 of the data was held out. The WTPM performs as well as the full dimensionality method and outperforms the partial dimensionality method.

Finally, we note that given an implementation of the tensor power method described in Anandkumar et al. [2014], performing the WTPM is very straightforward, and requires only multiplying the empirical estimates $\hat{\mathbf{S}}$ and $\hat{\mathbf{T}}$ by $\mathbf{p} \otimes \mathbf{p}$ and $\mathbf{p} \otimes \mathbf{p} \otimes \mathbf{p}$ respectively, before performing the decomposition. In practice, $\hat{\mathbf{T}}$ is frequently not calculated explicitly in order to avoid a cubic dependence of the runtime on the dimensionality. Rather, it is possible to use an implicit calculation of $\hat{\mathbf{T}}$ that reduces the runtime to $O(DK + nnz(\mathbf{X}))$, where $nnz(\mathbf{X})$ is the number of non-zero elements in $\mathbf{X}$ [Zou et al., 2013]. However, the method proposed in Zou et al. [2013], as well as other methods which do not explicitly calculate $\hat{\mathbf{T}}$ can only work when there are no missing elements in the data. While the explicit calculation of the third moments does not scale well with dimensionality of the data, its runtime is independent of document length once the data is converted into word-count format, and should therefore out-perform inference methods such as Gibbs-sampling for data consisting of very long documents.

7 Related Work

Tensor decomposition has been studied for nearly a century, but has recently gained popularity within many different scientific communities [Kolda and Bader, 2009]. In particular, a series of works have shown that tensor decomposition methods can recover the parameters of many classes of latent variable models [Anandkumar et al., 2014, Arora et al., 2012, Collins and Cohen, 2012, Zou et al., 2013, Parikh et al., 2011, Podosinnikova et al., 2015]. These works generally provide connections between the quality of the moment estimates and the model parameter estimates as a step toward deriving high-probability bounds on the quality of model parameter estimates; however, none of them address the question of what to do when moment estimates are of

only moderate and unequal quality.

The tensor decomposition methods used in this paper are connected to the general method of moments (GMM) [Hansen, 1982] as both methods learn the latent parameters of a distribution from its moments, but while GMM is very general and can be applied to a much larger range of statistical moments, the methods discussed here utilize the unique moments structure of latent variable models such as GP and GM to achieve more efficient learning.

More broadly, it is well-known that decompositions of empirical moments may result in model parameter estimates that are far from the maximum-likelihood parameter values, especially in the presence of model mismatch. Zhang et al. [2014] and Shaban et al. [2015] both show that in certain circumstances, using the spectral parameter estimates as initialization for an optimization procedure can result in the discovery of the true maximum-likelihood parameter settings. Tran et al. [2016] use an M-estimation framework to regularize the parameter estimates. In the specific case of HMMs, Huang and Schneider [2013] show how to combine cross-sectional and sequence data to produce better parameter estimates. Cheng et al. [2015] investigate the effect uncertainties in the second order tensor elements have on its spectral structure in order to bound the numbers of topics. Again, none of these works address inference in the presence of unequal quality moment estimates.

Finally, there exists a large literature relating to algorithms for tensor decompositions. Jain and Oh [2014] discuss decompositions when tensor elements are missing, distinct from our case in which certain dimensions are not always recorded. Weighted matrix factorization methods such as Srebro et al. [2003], are closely related in spirit to the algorithm we presented and perform a weighted decomposition of higher order tensors. Some works such as Chu and Ghahramani [2009], Rai et al. [2014] used a Bayesian approach to decompose partially observed tensors, but these works focused on missingness in the tensors themselves, rather than on missingness in the data from which the tensors are estimated.

8 Conclusion

In this work, we derived and presented the Weighted Tensor Decomposition Method (WTDM) for method-of-moment inference in latent variables with missing data. In situations where some dimensions are incompletely observed, we show that we can still leverage the associated low quality moment estimates to get better parameter estimation than simply ignoring that information, while minimizing inference errors due to moments that are too noisy. Our method is easy to implement and enjoys the same large-sample recovery guarantees as standard tensor decomposition-based inference methods; we demonstrate its benefits for two models and our derivations are general to other models as well.

References

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.

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.

Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models–going beyond svd. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 1–10. IEEE, 2012.

Sanjeev Arora, Rong Ge, Yonatan Halpern, David M Mimno, Ankur Moitra, David Sontag, Yichen Wu, and Michael Zhu. A practical algorithm for topic modeling with provable guarantees. In *ICML (2)*, pages 280–288, 2013.

David M Blei and John D Lafferty. A correlated topic model of science. *The Annals of Applied Statistics*, pages 17–35, 2007.

Jean-François Cardoso and Antoine Souloumiac. Blind beamforming for non-gaussian signals. In *IEE Proceedings F (Radar and Signal Processing)*, volume 140, pages 362–370. IET, 1993.

Dehua Cheng, Xinran He, and Yan Liu. Model selection for topic models via spectral decomposition. In *AISTATS*, 2015.

Wei Chu and Zoubin Ghahramani. Probabilistic models for incomplete multi-dimensional arrays. In *Artificial Intelligence and Statistics*, pages 89–96, 2009.

Michael Collins and Shay B Cohen. Tensor decomposition for fast parsing with latent-variable pcfgs. In *Advances in Neural Information Processing Systems*, pages 2519–2527, 2012.

Finale Doshi-Velez, Yaorong Ge, and Isaac Kohane. Comorbidity clusters in autism spectrum disorders: an electronic health record time-series analysis. *Pediatrics*, 133(1):e54–e63, 2014.

Lars Peter Hansen. Large sample properties of generalized method of moments estimators. *Econometrica: Journal of the Econometric Society*, pages 1029–1054, 1982.

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

Daniel Hsu, Sham M Kakade, and Tong Zhang. A spectral algorithm for learning hidden markov models. *Journal of Computer and System Sciences*, 78(5):1460–1480, 2012.

Tzu-Kuo Huang and Jeff G Schneider. Spectral learning of hidden markov models from dynamic and static data. In *ICML (3)*, pages 630–638, 2013.

Aapo Hyv et al. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, 1999.

Prateek Jain and Sewoong Oh. Provable tensor factorization with missing data. In *Advances in Neural Information Processing Systems*, pages 1431–1439, 2014.

Alistair EW Johnson, David J Stone, Leo A Celi, and Tom J Pollard. The mimic code repository: enabling reproducibility in critical care research. *Journal of the American Medical Informatics Association*, page ocx084, 2017.

Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

Wei Li and Andrew McCallum. Pachinko allocation: Dag-structured mixture models of topic correlations. In *Proceedings of the 23rd international conference on Machine learning*, pages 577–584. ACM, 2006.

Tom Mitchell. UCI machine learning repository: Twenty newsgroups, 1997. URL https://archive.ics.uci.edu/ml/datasets/Twenty+Newsgroups.

Ankur P Parikh, Le Song, and Eric P Xing. A spectral algorithm for latent tree graphical models. In *Proceedings of the 28th International Conference on Machine Learning*, pages 1065–1072, 2011.

Anastasia Podosinnikova, Francis Bach, and Simon Lacoste-Julien. Rethinking lda: moment matching for discrete ica. In *Advances in Neural Information Processing Systems*, pages 514–522, 2015.

Piyush Rai, Yingjian Wang, Shengbo Guo, Gary Chen, David Dunson, and Lawrence Carin. Scalable bayesian low-rank decomposition of incomplete multiway tensors. In *International Conference on Machine Learning*, pages 1800–1808, 2014.

Amirreza Shaban, Mehrdad Farajtabar, Bo Xie, Le Song, and Byron Boots. Learning latent variable models by improving spectral solutions with exterior point method. In *UAI*, pages 792–801, 2015.

Nathan Srebro, Tommi Jaakkola, et al. Weighted low-rank approximations. In *Icml*, volume 3, pages 720–727, 2003.

Dustin Tran, Minjae Kim, and Finale Doshi-Velez. Spectral m-estimation with applications to hidden markov models. *AISTATS*, 2016.

Yuchen Zhang, Xi Chen, Denny Zhou, and Michael I Jordan. Spectral methods meet em: A provably optimal algorithm for crowdsourcing. In *Advances in neural information processing systems*, pages 1260–1268, 2014.

James Y Zou, Daniel J Hsu, David C Parkes, and Ryan P Adams. Contrastive learning using spectral methods. In *Advances in Neural Information Processing Systems*, pages 2238–2246, 2013.