

# ESTIMATION AND UNCERTAINTY QUANTIFICATION IN TENSOR COMPLETION WITH SIDE INFORMATION

by

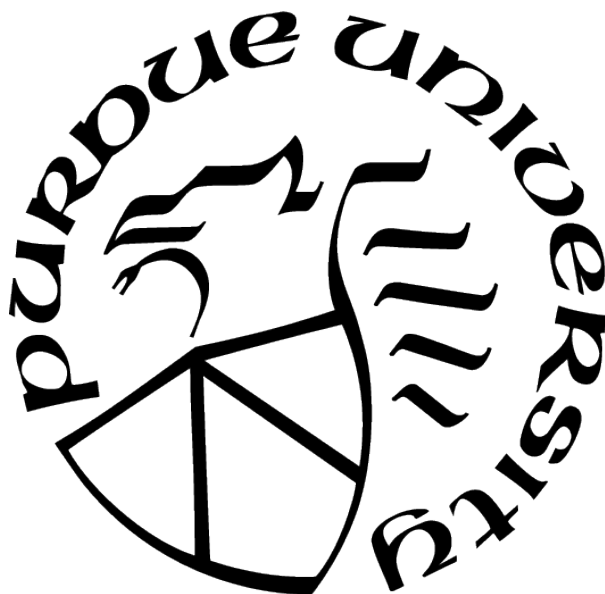
Somnooma Hilda Marie Bernadette Ibriga

A Dissertation

*Submitted to the Faculty of Purdue University*

*In Partial Fulfillment of the Requirements for the degree of*

Doctor of Philosophy



Department of Statistics

West Lafayette, Indiana

August 2021

**THE PURDUE UNIVERSITY GRADUATE SCHOOL  
STATEMENT OF COMMITTEE APPROVAL**

**Dr. Wei Sun, Co-Chair**

Krannet School of Management, Purdue University

**Dr. Bruce Craig, Co-chair**

Department of Statistics, Purdue University

**Dr. Jun Xie**

Department of Statistics, Purdue University

**Dr. Anindya Bhadra**

Department of Statistics, Purdue University

**Approved by:**

Dr. Jun Xie

To my family and to my country Burkina Faso

## ACKNOWLEDGMENTS

*“Nug bi yend ka wuk zom ye.”* (Mossi proverb)

trans: *“It takes a community to succeed.”*

This work is the product of years of support from people from different walks of life who believed in me and my ability to earn this degree.

On the academic front, I would like to express the deepest appreciation to my academic advisor, Dr. Wei Sun for his guidance, knowledge sharing, but also his understanding and compassion during my doctoral studies. Thank you, Dr. Sun, for emulating the role of a stellar advisor and research mentor. You are an academic and personal role model to me. Thank you to my co-advisor, Dr. Bruce Craig, who was always ready to step in and advocate for my cause. The time spent under your supervision in the consulting service has been invaluable and you are an essential part of the reason why I love the job of consultancy. Thank you to my committee members Dr. Jun Xie and Dr. Anindya Bhadra for their support and constructive feedback throughout the research process.

Beside my committee, I have been lucky to be surrounded by supportive and helpful faculty and staff in the department. I would also like to recognize two faculty mentors in the department namely, Dr. Arman Sabbaghi and Dr. Mark Ward who have been generous with their time and knowledge and have been instrumental to my academic success. I also thank Dr. Guang Cheng and his big data research group for offering me a place to learn and present my work on numerous occasions. A special thank you to the colleagues and friends I have made in the department, whose support and friendship have been invaluable especially, Jeanine Gngang, Dr. Qi Lui, Dr. Raquel Ferreira, Kara Keller, Dr. Hakeem Abdul Wahab, Yumin Zhang, Dr. Evidence Matangi, Bingjing Tang and Dr. Chizoku Iwaki.

Also, a special thanks to the mentors who helped me discover and nurture my love for mathematics, Mr. Issiaka Hie, Dr. Erin Martin, Dr. Micah James and Dr. Giovanni Petris.

I would not have made it this far in my academic journey without the support of my family. To my parents Dr. Luc and Justine Ibriga, who always believed in my dreams and trusted me to take the first step toward this degree, by allowing me to move to India on my own at the tender age of 16 in the pursuit of education. You have imparted in me the love for

learning and knowledge sharing. Thank you for being the force behind my accomplishments. I am still learning from your sense of integrity, perseverance and resilience.

To my aunt Felicity Kompaore who has opened her house in the US to me and made it a home away from home. Thank you to my sisters Latifa and Miranda Ibriga and my cousins Dr. Alexia Nikyema, Sabine Ouedraogo and Lydia Ouedraogo, who offered me their strong support and a safe space to escape to throughout my academic journey. To the friends turned sisters Marianne Bampire, Ann Amenuvor, Tekber Ahmed Saleh, Halimatou Bachir Abdou and Kawuia Ouedraogo Nebie, whose support has been constant throughout my studies. Thank you for the unfaltering support from the rest of my family in Burkina Faso and around the world.

My graduate training would not be complete without my teaching, consulting and internship experience. My genuine gratitude to my internship team at Asana, specially to my managers, Sarah White, Erin Akinici and Dr. Andrew Fiore, whose leadership, trust and guidance have enriched my professional growth.

Finally, I would like to thank my grandparents both alive and deceased whose continual blessings were always felt and whose belief in the power of education have been passed from generation to generation and has culminated into this work.

# TABLE OF CONTENTS

LIST OF TABLES . . . . .	9
LIST OF FIGURES . . . . .	10
ABBREVIATIONS . . . . .	11
ABSTRACT . . . . .	12
1 INTRODUCTION . . . . .	14
1.1 Role of Side Information in Tensor Completion . . . . .	14
1.2 The Need for Uncertainty Quantification in Tensor Recovery . . . . .	17
2 BACKGROUND ON TENSOR COMPLETION . . . . .	19
2.1 Existing Tensor Completion Results . . . . .	19
2.1.1 Tensor Completion with Side Information: . . . . .	20
2.1.2 Tensor Completion with Theoretical Guarantees: . . . . .	21
2.1.3 Tensor completion and uncertainty quantification . . . . .	22
2.2 Notation and Tensor Algebra . . . . .	23
3 COVARIATE ASSISTED SPARSE TENSOR COMPLETION . . . . .	26
3.1 Methodology . . . . .	26
3.1.1 Model . . . . .	26
3.1.2 Algorithm . . . . .	28
3.1.3 Initialization Procedure . . . . .	31
3.1.4 Rank and Cardinality Tuning . . . . .	32
3.2 Theoretical Analysis . . . . .	32
3.3 Case 1: Non-sparse Tensor and Matrix with Equal Weights . . . . .	33
3.3.1 Assumptions . . . . .	33
3.3.2 Main Theoretical Results . . . . .	36
3.4 Case 2: Sparse Tensor and Matrix with General Weights . . . . .	37
3.4.1 Assumptions . . . . .	37

3.4.2	Main Theoretical Results . . . . .	39
3.4.3	Discussion . . . . .	40
3.5	Simulations . . . . .	40
3.5.1	Missing Percent . . . . .	43
3.5.2	Noise Level . . . . .	44
3.5.3	Component Size . . . . .	47
3.5.4	Rank . . . . .	47
3.6	Real Data Analysis . . . . .	48
3.7	Proof of Main Theorems . . . . .	53
3.7.1	Proof of Theorem 3.3.1 . . . . .	53
3.7.2	Key Lemmas . . . . .	54
3.7.3	Proof of Theorem 3.4.1 . . . . .	57
3.8	Additional Results . . . . .	59
3.8.1	Proof of Lemma 1 . . . . .	59
3.8.2	Proof of Lemma 2 . . . . .	62
3.8.3	Proof of Lemma 3 . . . . .	68
3.8.4	Proof of Lemma 4 . . . . .	72
3.9	Auxillary Lemmas . . . . .	80
4	UNCERTAINTY QUANTIFICATION IN COVARIATE ASSISTED TENSOR COMPLETION . . . . .	88
4.1	Methodology . . . . .	89
4.1.1	Model and Algorithm . . . . .	90
4.2	Theoretical Analysis . . . . .	93
4.2.1	Assumptions . . . . .	93
4.2.2	Distributional Guarantee of Tensor Factors . . . . .	96
4.2.3	Confidence Interval for Tensor Factors . . . . .	97
4.3	Simulations . . . . .	100
4.3.1	Empirical Distribution . . . . .	100
4.3.2	Empirical Coverage Rate of Confidence Intervals . . . . .	102

4.3.3	Tightness of Confidence Intervals . . . . .	104
4.4	Proof of Main Theorem . . . . .	106
4.4.1	Proof of Theorem 4.2.1: Gaussian Noise . . . . .	106
4.4.2	Proof of Theorem 4.2.2: General Noise . . . . .	109
4.4.3	Proof of Theorem 4.2.3: Confidence Intervals for Tensor Factors . . . . .	109
4.5	Additional Results . . . . .	111
4.5.1	Proof of Lemma 14 . . . . .	111
4.5.2	Proof of Lemma 15 . . . . .	114
4.5.3	Proof of Lemma 16 . . . . .	118
4.5.4	Proof of Lemma 17 . . . . .	120
4.5.5	Proof of Lemma 18 . . . . .	122
4.5.6	Proof of Lemma 19 . . . . .	123
4.6	Auxillary Lemmas . . . . .	128
4.7	Additional Material . . . . .	134
4.7.1	Lyapunov-type Bound . . . . .	135
4.7.2	The Leave-One-Out Method . . . . .	135
5	CONCLUDING REMARKS . . . . .	137
	REFERENCES . . . . .	140
	VITA . . . . .	146



## LIST OF TABLES

3.1	Statistical error of shared tensor component in Theorem 3.4.1 under various conditions. Improvement represent improvement over the recovery error of the non-shared components . . . . .	41
3.2	Estimation errors with varying missing percentages. Reported values are the average and standard deviation (in parentheses) of tensor, tensor components and weight recovery error based on 30 data replications. <b>COSTCO</b> : the proposed method; <b>tenALSsparse</b> : sparse version of the tensor completion method by [24]; <b>OPT</b> : the gradient based all at once optimization method of [36]; symbol ( $\tilde{A}$ ) used to put shared tensor-matrix component <b>A</b> in emphasis. . . . .	44
3.3	Estimation errors with varying noise levels of error matrix and error tensor. Reported values are the average and standard deviation (in parentheses) of estimation errors. <b>COSTCO</b> : the proposed method; <b>tenALSsparse</b> : sparse version of the tensor completion method by [24]; <b>OPT</b> : the gradient based all at once optimization method of [36]. . . . .	46
3.4	Estimation errors of <b>COSTCO</b> with varying coupled dimension $d_1$ . . . . .	47
3.5	Estimation errors of <b>COSTCO</b> with varying rank. . . . .	48
3.6	Top ten words for 7 chosen topics. Top words were obtained through LDA. . . .	50
4.1	Empirical coverage rate for 95% confidence interval of <b>COSTCO</b> versus <b>tenALS</b> with varying tensor noise parameter $\sigma$ . . . . .	103
4.2	Empirical coverage rate for 95% confidence interval of <b>COSTCO</b> versus <b>tenALS</b> with varying tensor reveal probabilities $p$ . . . . .	103

## LIST OF FIGURES

1.1	A. sparse (user $\times$ ad $\times$ device) CTR tensor with missing entries; B. sparse CTR tensor with missing entries coupled with matrix of ad covariates. The red cells represent missing entries; blue cells represent zeros, grey cells represent non-zero entries. . . . .	15
2.1	Fibers of a third-order tensor. Image obtained from [29] . . . . .	24
2.2	Slices of a third-order tensor. Image obtained from [29] . . . . .	24
3.1	Illustration of <b>COSTCO</b> showing recovery procedure for missing entries through joint tensor matrix decomposition; red cells represent missing entries. The tensor and matrix are coupled along the first mode and the components $\mathbf{a}_r$ , $r \in [R]$ are shared by the tensor and matrix decomposition. . . . .	30
3.2	Illustration of missing data and sparsity in our ad CTR tensor. . . . .	49
3.3	Scatter plot of the ad latent components obtained from three methods. Different clusters are represented via different colors. . . . .	51
3.4	Result of ad clusters obtained using different methods . . . . .	52
4.1	A. Order-3 (user $\times$ ad $\times$ device) tensor with missing entries; B. Order-3 tensor with missing entries coupled with matrices of ad, user and device covariates $\mathbf{M}_a$ , $\mathbf{M}_b$ , $\mathbf{M}_c$ respectively. The red cells represent missing entries; grey and white cells represent non-zero entries. . . . .	89
4.2	Illustration of <b>COSTCO</b> showing recovery procedure for missing entries through joint decomposition of a rank 1 tensor and rank 1 matrices; red cells represent missing entries. The tensor and matrices $\mathbf{M}_a$ , $\mathbf{M}_b$ and $\mathbf{M}_c$ are coupled along mode 1, mode 2 and mode 3 respectively. The components $\mathbf{a}$ , $\mathbf{b}$ and $\mathbf{c}$ are shared by the tensor and matrices $\mathbf{M}_a$ , $\mathbf{M}_b$ and $\mathbf{M}_c$ decomposition respectively. . . . .	92
4.3	Q-Q (quantile-quantile) plots of normalized factor entry error with $p = 0.1$ and $\sigma_T = \sigma_M = 0.01$ . . . . .	101
4.4	Width of constructed confidence interval with varying tensor noise level for <b>COSTCO</b> versus <b>tenALS</b> . . . . .	104
4.5	Width of constructed confidence interval with varying tensor reveal probability for <b>COSTCO</b> versus <b>tenALS</b> . . . . .	105
4.6	Illustration of the leave one out procedure applied to a third order tensor and a matrix. The tensor was sliced along the third mode. The green shades values represent tensor and matrix entries which still contain some noise. Whereas the non shades values are replaced by the true value of tensor. The missing probability is set to $p = 1$ for the sake of the illustration. . . . .	136

## ABBREVIATIONS

ALS	Alternating Least Squares
COSTCO	Covariate-assisted Sparse Tensor Completion
CP	CANDECOMP/PARAFAC or Canonical polyadic decomposition
RTPM	Robust Tensor Power Method
SVD	Singular Value Decomposition
TenALS	Tensor ALS

# ABSTRACT

This work aims to provide solutions to two significant issues in the effective use and practical application of tensor completion as a machine learning method. The first solution addresses the challenge in designing fast and accurate recovery methods in tensor completion in the presence of highly sparse and highly missing data. The second takes on the need for robust uncertainty quantification methods for the recovered tensor.

## **Covariate-assisted Sparse Tensor Completion**

In the first part of the dissertation, we aim to provably complete a sparse and highly-missing tensor in the presence of covariate information along tensor modes. Our motivation originates from online advertising where users click-through-rates (CTR) on ads over various devices form a CTR tensor that can have up to 96% missing entries and has many zeros on non-missing entries. These features makes the standalone tensor completion method unsatisfactory. However, beside the CTR tensor, additional ad features or user characteristics are often available. We propose Covariate-assisted Sparse Tensor Completion (COSTCO) to incorporate covariate information in the recovery of the sparse tensor. The key idea is to jointly extract latent components from both the tensor and the covariate matrix to learn a synthetic representation. Theoretically, we derive the error bound for the recovered tensor components and explicitly quantify the improvements on both the reveal probability condition and the tensor recovery accuracy due to covariates. Finally, we apply COSTCO to an advertisement dataset from a major internet platform consisting of a CTR tensor and ad covariate matrix, leading to 23% accuracy improvement over the baseline methodology. An important by-product of our method is that clustering analysis on ad latent components from COSTCO reveal interesting and new ad clusters, that link different product industries which are not formed in existing clustering methods. Such findings could be directly useful for better ad planning procedures.

## **Uncertainty Quantification in Covariate-assisted Tensor Completion**

In the second part of the dissertation, we propose a framework for uncertainty quantification for the imputed tensor factors obtained from completing a tensor with covariate information. We characterize the distribution of the non-convex estimator obtained from using the algorithm

COSTCO down to fine scales. This distributional theory in turn allows us to construct proven valid and tight confidence intervals for the unseen tensor factors. The proposed inferential procedure enjoys several important features: (1) it is fully adaptive to noise heteroscedasticity, (2) it is data-driven and automatically adapts to unknown noise distributions and (3) in the high missing data regime, the inclusion of side information in the tensor completion model yields tighter confidence intervals compared to those obtained from standalone tensor completion methods.

# 1. INTRODUCTION

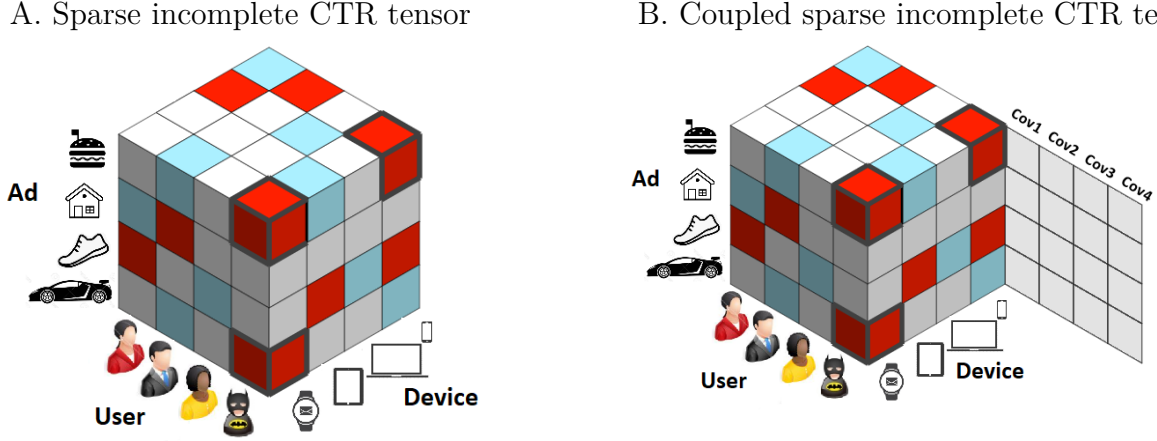
Low-rank tensor completion aims to impute missing entries of a partially observed tensor by forming a low-rank decomposition on the observed entries. It has been widely used in various scientific and business applications, including recommender systems [1]–[3], neuroimaging analysis [4]–[6], signal processing [7], [8], social network analysis [9], [10], personalized medicine [11], [12], and time series analysis [13]. We refer to the recent surveys on tensors for more real applications [14], [15].

In spite of its popularity, it is also well known that when the percent of missing entries in the tensor is very high, a standalone tensor completion method often fails to yield desirable recovery results. Fortunately, in many real applications, we also have access to some side covariate information. This dissertation focuses on the effective integration of this additional information in the tensor completion problem.

## 1.1 Role of Side Information in Tensor Completion

Our motivation originates from online advertising applications, where advertisement (ad) information is usually described by both users’ click behavior data and ad characteristics data. More formally, the users click data, referred to as the click-through rate (CTR) of the ads, quantifies the user click behavior on different ads, various platforms, different devices or over time etc. The CTR data are therefore often represented as a tensor of three, four or five dimensions, e.g., the user  $\times$  ad  $\times$  device tensor shown in Figure 1.1. The ad characteristic data on the other hand is usually represented in the form of a matrix which contains context information for each ad or background information for each user. Typically in online advertising not all users are presented with all ads, thus creating many missing entries in the CTR tensor. Moreover, users typically engage with only a small subset of the ads presented to them. Low rates of ads engagement is a common phenomenon in online advertising which begets a highly sparse CTR tensor (many zero entries) with high percentage of missing entries. For instance, in our real data described in Section 3.6, the ad CTR tensor has 96% missing entries and is highly sparse with only 40% of the revealed entries being nonzero. We show in Chapters 3.5 and 3.6 that methods using a standalone tensor completion

often fail at recovering the missing entries of a tensor with such high missing rate. On the contrary the ad characteristic matrix is usually relatively complete and dense. It therefore becomes advantageous to incorporate the ad characteristic information in a model to recover the missing entries of the CTR tensor. The structure of the sparse CTR tensor with missing entries coupled with the ad characteristic data is illustrated in Figure 1.1. As shown in Figure 1.1 the two sources of data; CTR tensor and ad covariates matrix are coupled along the ad mode.



**Figure 1.1.** A. sparse (user  $\times$  ad  $\times$  device) CTR tensor with missing entries; B. sparse CTR tensor with missing entries coupled with matrix of ad covariates. The red cells represent missing entries; blue cells represent zeros, grey cells represent non-zero entries.

In the first part of this dissertation, we aim to complete a sparse and highly-missing tensor in the presence of covariate information along tensor modes. We propose Covariate-assisted Sparse Tensor Completion (*COSTCO*) to recover missing entries in a highly sparse tensor with a large percent of missing entries. Under the low-rank assumption on both the tensor and covariate matrix, we assume the latent components corresponding to the coupled mode are shared by both the tensor and matrix decomposition. This model encourages a synthetic representation of the coupled mode by leveraging the additional covariate information into tensor completion. We formulate the parameter estimation as a non-convex optimization with sparsity constraints, and propose an efficient sparse alternating least-squares approach with an extra refinement step. Our algorithm jointly extracts latent features from both tensor

and covariate matrix and uses the covariate information to improve the recovery accuracy of the unknown tensor components. We showcase, through extensive numerical studies, that **COSTCO** is able to successfully recover entries for a tensor even with 98% missing entries.

In addition to the above methodological contributions, we also make theoretical contributions to the understanding of how side covariate information affects the performance of tensor completion. In particular, we derive the non-asymptotic error bound for the recovered tensor components and explicitly quantify the improvements on both the reveal probability condition and the tensor recovery accuracy due to additional covariate information. We show that **COSTCO** allows for a relaxation on the lower bound of the reveal probability  $p$  compared to that required in tensor completion with no covariates, see Assumption 5 for details. In the extreme case where all tensor modes are coupled with covariate matrices, we can still recover the tensor entries even when the reveal probability of the tensor is close to zero. Moreover, we present the statistical errors for the shared tensor component (corresponding to the coupled mode) and non-shared tensor components separately to demonstrate the gain brought in through the coupling of covariates information in the model. We show that given some mild assumptions on noise levels and condition numbers, our **COSTCO** guarantees an improved recovery accuracy for the shared component. Unlike existing theoretical analysis on low-rank tensors which assumes the error tensor to be Gaussian, we do not impose any distributional assumption on the error tensor or the error matrix. Our theoretical results depends on the error term only through its sparse spectral norm.

Finally, we apply our **COSTCO** to the advertising data from a major internet company to demonstrate its practical advantages. **COSTCO** makes use of both ad CTR tensor and ad covariate matrix to extract the latent component which leads to about 23% accuracy improvement in recovering the missing entries when compared to the standalone sparse tensor completion method. Moreover, an important by-product from our **COSTCO** is to use the recovered ad latent components for better ad clustering. Ad clustering is an essential task for targeted advertising that helps lead useful ad recommendation for online platform users. Cluster analysis on our ad latent components reveals interesting and new clusters that link different product industries which are not formed in existing clustering methods.



Such findings could directly help the marketing team to strategize the ad planing procedure accordingly for better ad targeting.

## 1.2 The Need for Uncertainty Quantification in Tensor Recovery

In addition to the recovery task in tensor completion, there is a need to assess the trustworthiness of these predictions in order to communicate the risk attached to the recovered components. This can be done by characterizing the distribution of the recovered tensor factors, which in turn can be used to construct confidence intervals for the recovered components. However, due to the non-convexity of most tensor completion problems, combined with the high missing entry rate, characterizing the distribution of the recovered components is a challenging problem.

In the second part of this dissertation, we propose a robust uncertainty quantification method for the recovered tensor components. We provide the theoretical work which characterizes, under mild assumptions, the distribution of recovered tensor factors and we provide a data driven method for constructing entry-wise confidence intervals for the unknown tensor factors. We then show, both theoretically and through a series of simulations, the validity of the constructed confidence intervals. This reveals the fact that our method generates shorter confidence intervals compared to those obtained using standalone tensor completion methods, primarily due to the fact that we are including covariate information.

**Dissertation Outline:** The remainder of the dissertation is organized as follows. We begin in Section 2.1 with a review of work on tensor completion and map the gap that exists for both tensor recovery and uncertainty quantification of tensor estimates in the highly sparse and highly missing data regime. In Section 2.2, we review some notations and present some preliminaries of tensor algebra. In Chapter 3, we propose **COSTCO** an algorithm for tensor completion with side information and provide theoretical guarantees for the method in Section 3.2, along with simulation and real data analysis results in Sections 3.5 and 3.6. Section 3.7 contains proof details for the theoretical analysis of Chapter 3.

In Chapter 4 we characterise the distribution of the recovered tensor components and propose a method for building confidence intervals. In Section 4.3 we test the validity of the

confidence interval and their robustness to noise level and missing entry percentage through simulations. Proof details for Chapter 4 are provided in Section 4.4.

Concluding remarks on the results derived in the dissertation as well as a discussion on potential study extension and future research in the tensor completion field are provided in Chapter 5.

## 2. BACKGROUND ON TENSOR COMPLETION

In this section we provide a literature review on tensor completion, including some background on tensor notation and tensor algebra.

### 2.1 Existing Tensor Completion Results

Low-rank tensor completion is a popular subject of theoretical study in a wide range of fields such as statistics and mathematics [16]–[20] as well as computer science and engineering [20]–[24]. In its most general form, the tensor completion problem aims at imputing missing entries of a partially observed noisy tensor. The practical uses of tensor completion methods abound and are pervasive in application driven studies in computer vision[25], [26], signal processing [7], [8], [27], recommender systems [1], [2], community detection [28] and personalized medicine [11], [12]. For instance, in recommender systems, tensor completion methods have been adapted for collaborative filtering when the scope of the dataset is beyond the traditional two dimensional (user, item) pair [2]. These tensor-based recommender systems, not only allow for an improved recommendation algorithm, but also make collaborative filtering feasible on multidimensional data. At its origin, recommender systems methods relied on matrix factorization, however recent studies have shown that tensor-based recommender systems often outperform those matrix factorization methods provided the tensor is not highly sparse. In the field of personalized medicine, the past lustrum has witnessed a marked increase in the use of tensor completion to refine treatment protocols. In a 2019 study, Wang, Zhang, Chen, *et al.* [12] use tensor completion to predict the onset of new chronic diseases for individual patients. This is done by representing high-order interactions of patients, chronic diseases and patient-specific characteristics as a tensor with missing data. Tensor completion is therefore used in their work to reveal latent patterns of co-occurring chronic diseases which enables a more effective prediction of the onset of chronic diseases in a patient.

The overwhelming majority of work in the tensor completion literature relies on a critical low-rank assumption [14]. Solving the completion problem is therefore often formulated as a rank constrained optimization problem which is known to be NP-hard [29], [30]. To circumvent such a hurdle, researchers often relax the problem by assuming knowledge of the

tensor rank, which allows the completion problem to be solved in polynomial time [23]. The Alternating Least-Square methods (ALS) is an example of such tensor completion method with rank relaxation. The ALS method approaches the tensor completion problem by simply conducting the CANDECOMP/PARAFAC (CP) tensor decomposition on the tensor with missing entries [31]. In their theoretical work on tensor completion, Jain and Oh [24] used the ALS approach to derive theoretical guarantees for noiseless, symmetric, orthogonal tensors with missing entries. However, their algorithm and theoretical analysis do not address the case of the non-orthogonal and noisy tensor which completion problem is acknowledged to be non trivial. Although the simplicity of the ALS method makes it an attractive option for tensor completion, it is also well known that as the percent of missing entries increases, a standalone ALS method often fails at yielding desirable recovery results.

On the other hand, Singh and Gordon [32] and Smilde, Westerhuis, and Boqué [33] showed in their theoretical analysis of matrix completion that allowing side information along with the observed matrix entries improved the convergence rate of the matrix completion algorithm and reduced the required number of observed entries for perfect recovery for a  $n \times n$  matrix from  $O(n \ln(2n))$  to  $O(\ln(n))$ . Therefore, it is natural to explore an ALS based tensor completion algorithm that allows the inclusion of side information under the form of a covariate matrix.

### 2.1.1 Tensor Completion with Side Information:

The simultaneous extraction of latent information from multiple sources of data can be interpreted as a form of data fusion [34]–[45]. Among them, there are a few works related to tensor completion with side information. The most related work to our approach is the gradient-based all-at-once optimization method proposed by Acar, Kolda, and Dunlavy [36], which updates the matrix and tensor components simultaneously. We assess its performance to in our experiments in Section 3.5 and find that it is consistently inferior to COSTCO. Zhou, Qian, Shen, *et al.* [39] proposed a Riemannian conjugate gradient descent algorithm to solve the tensor completion problem in the presence of side information. However, their procedure does not address the tensor completion problem in the presence of a high percent

of missing entries combined with high sparsity. Choi, Jang, and Kang [42] developed a fast and scalable algorithm for the estimation of shared latent features in coupled tensor matrix model. However, their approach does not allow missing entries, therefore only works for complete data.

Importantly, all the aforementioned works do not provide any theoretical analysis for their methods. Kishan, Makoto, and Hiroshi [41] proposed a convex coupled tensor-matrix completion method through the use of coupled norms and derived its excess risk bound. In a more general setting, Huang, Liu, and Zhu [43] applied the tensor ring decomposition method on the coupled tensor-tensor problem and derived the excess risk bound. However, the methods considered in these two works do not account for noise in the tensor or matrix, (i.e., their model is noiseless), nor do they consider the sparse tensor case. To the best of our knowledge, our work is the first method with theoretical guarantee, that is tailored for completing a highly sparse and highly missing tensor in the presence of covariate information.

### 2.1.2 Tensor Completion with Theoretical Guarantees:

Our theoretical analysis is related to a list of recent theoretical work in standalone tensor completion that does not incorporate covariate information [16]–[18], [24], [46]. In particular, Jain and Oh [24] provided recovery guarantee for symmetric and orthogonal tensors with missing entries, but did not explore recovery for the tensor completion with coupled covariates nor did they address the case of the non-orthogonal, noisy and sparse tensor. Zhang [18] established a sharp recovery error for a special tensor completion problem, where the missing pattern was not uniformly missing but followed a cross structure. Xia and Yuan [46] proved exact recovery for the noiseless tensor completion problem under a uniform random sampling schema. Unlike our analysis which is based on the CP model, they do not address the noisy tensor case and analyze the completion problem under the Tucker model representation which leads to different assumptions than those required in our case. In their recent work, Xia, Yuan, and Zhang [16] proposed a two-step algorithm (a spectral initialization method followed by the power method) for the noisy Tensor completion case and established the optimal statistical rate in low-rank tensor completion. Different from our model, they assumed the

error tensor to be sub-gaussian and did consider sparsity in tensor completion. Cai, Li, Poor, *et al.* [17] also independently proposed a provable two stage algorithm (initialization followed by gradient descent) for the noisy tensor completion problem. These two works provide ground breaking theoretical contributions to tensor completion. Importantly, none of the aforementioned work accommodates the inclusion of covariate information in the tensor completion model. The coupled sparse tensor and matrix formulation in our **COSTCO** poses unique difficulties in the theoretical analysis. The unequal weights of the tensor and matrix prevent us from obtaining a close-form solution for the alternative least-squares problem compared to the traditional tensor completion. Moreover, the presence of non-orthogonality, general noise, and sparsity in our model introduce additional challenges. These make our theoretical analysis far from a simple extension to the standard tensor completion problem as it calls for new techniques and assumptions.

### 2.1.3 Tensor completion and uncertainty quantification

The problem of uncertainty quantification for recovered tensor factors and corresponding tensor entries is quite challenging. This is reflected in the negligible numbers of works in the literature that address this problem. Only in recent years have a couple works with theoretical guarantees appeared on the subject. Due to the non-convexity of most tensor completion problems, characterizing the distribution of the recovered components becomes challenging. Cai, Poor, and Chen [47] take on this task for the case of the standalone tensor completion, by proposing a construction technique for confidence intervals of the tensor factor entries through the use of a debiased estimator obtained from their two-stage completion algorithm [17]. Beside the aforementioned work, we are only aware of the work of Xia, Zhang, and Zhou [48] who approach the inference task in tensor estimation through the use of tensor regression estimation method under the Tucker model. They develop confidence regions for the singular subspace of the tensor factors based on the asymptotic distribution of the estimates obtained from alternating minimization algorithm. However, they do not address the case of the high missing percent regime and the reliance on tensor matricization in their

algorithm yields non-optimal sample complexity results.

## 2.2 Notation and Tensor Algebra

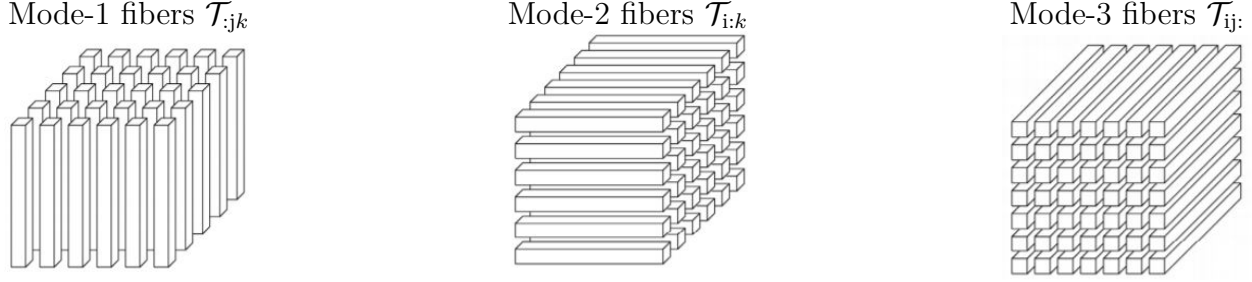
In this section, we introduce some notation, and review some background on tensors. Throughout the dissertation, we denote tensors by Euler script letters, e.g.,  $\mathcal{T}, \mathcal{E}$ . Matrices are denoted by boldface capital letters, e.g.,  $\mathbf{A}, \mathbf{B}, \mathbf{C}$ ; vectors are represented with boldface lowercase letters, e.g.,  $\mathbf{a}, \mathbf{v}$ , and scalars are denoted by lowercase letters, e.g.,  $a, \lambda$ . Furthermore, the  $n \times n$  identity matrix  $\mathbf{I}_n$  is simply written as  $\mathbf{I}$  when the dimension can be easily implied from the context.

Following Kolda and Bader [29], we use the term tensor to refer to a multidimensional array; a concept that generalizes the notion of matrices and vectors to higher dimensions. A first-order tensor is a vector, a second-order tensor is a matrix and a third-order tensor is a three dimensional array. Each order of a tensor is referred to as a mode. For example a matrix (second-order tensor) has two modes with mode-1 and mode-2 being the dimensions represented by the rows and columns of the matrix respectively.

Let  $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$  be a third-order non-symmetric tensor. We denote its  $(i, j, k)$ th entry as  $\mathcal{T}_{ijk}$ . A tensor fiber refers to a higher order analogue of matrix row and column and is obtained by fixing all but one of the indices of the tensor (see Figure 2.1). For the tensor  $\mathcal{T}$  defined above, the mode-1 fiber is given by  $\mathcal{T}_{:jk}$ ; the mode-2 fiber by  $\mathcal{T}_{i:k}$  and mode-3 fiber by  $\mathcal{T}_{ij:}$ . Next the slices of the tensor  $\mathcal{T}$  are obtained by fixing all but two of the tensor indices (see Figure 2.2). For example the frontal, lateral and horizontal slices of the tensor  $\mathcal{T}$  as denoted as  $\mathcal{T}_{::k}$ ,  $\mathcal{T}_{:j:}$  and  $\mathcal{T}_{i::}$ .

We define three different types of tensor vector products. For vectors  $\mathbf{u} \in \mathbb{R}^{n_1}, \mathbf{v} \in \mathbb{R}^{n_2}, \mathbf{w} \in \mathbb{R}^{n_3}$ , the mode-1, mode-2 and mode-3, tensor-vector product is a matrix defined as a combinations of tensor slices:  $\mathcal{T} \times_1 \mathbf{u} = \sum_{i=1}^{n_1} \mathbf{u}_i \mathcal{T}_{i::}$ ,  $\mathcal{T} \times_2 \mathbf{v} = \sum_{j=1}^{n_2} \mathbf{v}_j \mathcal{T}_{:j:}$ ,  $\mathcal{T} \times_3 \mathbf{w} = \sum_{k=1}^{n_3} \mathbf{w}_k \mathcal{T}_{::k}$ . The tensor multiplying two vectors along its two modes is a vector defined as:  $\mathcal{T} \times_2 \mathbf{v} \times_3 \mathbf{w} = \sum_{j,k} \mathbf{v}_j \mathbf{w}_k \mathcal{T}_{:jk}$ ,  $\mathcal{T} \times_1 \mathbf{u} \times_2 \mathbf{v} = \sum_{i,j} \mathbf{u}_i \mathbf{v}_j \mathcal{T}_{ij:}$ ,  $\mathcal{T} \times_1 \mathbf{u} \times_3 \mathbf{w} = \sum_{i,k} \mathbf{u}_i \mathbf{w}_k \mathcal{T}_{i:k}$ .

Finally the tensor-tensor product is a scalar defined as  $\mathcal{T} \times_1 \mathbf{u} \times_2 \mathbf{v} \times_3 \mathbf{w} = \sum_{i,j,k} \mathbf{u}_i \mathbf{v}_j \mathbf{w}_k \mathcal{T}_{ijk}$ . We denote  $\|\mathbf{M}\|$  and  $\|\mathbf{M}\|_F$  to be the spectral norm and the Frobenius norm of a matrix  $\mathbf{M}$ ,



**Figure 2.1.** Fibers of a third-order tensor. Image obtained from [29]

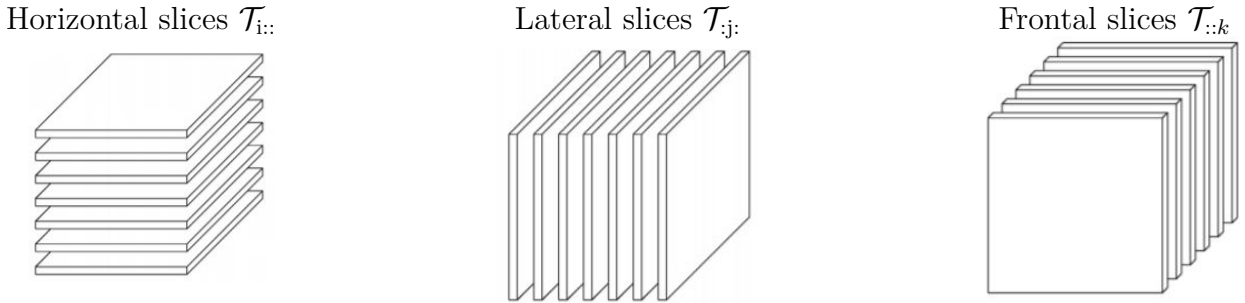
respectively. The spectral norm of a tensor  $\mathcal{T}$  is defined as

$$\|\mathcal{T}\| := \sup_{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=\|\mathbf{w}\|_2=1} \left| \mathcal{T} \times_1 \mathbf{u} \times_2 \mathbf{v} \times_3 \mathbf{w} \right|, \quad (2.1)$$

and its Frobenius norm is  $\|\mathcal{T}\|_F := \left( \sum_{i,j,k} \mathcal{T}_{ijk}^2 \right)^{1/2}$ . We define the sparse spectral norm of a matrix  $\mathbf{M}$  as  $\|\mathbf{M}\|_{<d_1>} := \sup_{\|\mathbf{u}\|_2=1, \|\mathbf{u}\|_0=d_1} \|\mathbf{M} \times_1 \mathbf{u}\|_2$  and the sparse spectral norm of a tensor  $\mathcal{T}$  as

$$\|\mathcal{T}\|_{<d_1, d_2, d_3>} := \sup_{\substack{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=\|\mathbf{w}\|_2=1 \\ \|\mathbf{u}\|_0=d_1, \|\mathbf{v}\|_0=d_2, \|\mathbf{w}\|_0=d_3}} \left| \mathcal{T} \times_1 \mathbf{u} \times_2 \mathbf{v} \times_3 \mathbf{w} \right|,$$

where  $d_1 < n_1$ ,  $d_2 < n_2$ ,  $d_3 < n_3$ . When  $d_1 = d_2 = d_3 = d$ , we simplify  $\|\mathcal{T}\|_{<d, d, d>}$  as  $\|\mathcal{T}\|_{<d>}$ .



**Figure 2.2.** Slices of a third-order tensor. Image obtained from [29]



Given a third-order tensor  $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , we denote its CP decomposition as

$$\mathcal{T} = \sum_{r \in [R]} \lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r, \quad (2.2)$$

where  $[R]$  indicates the set of integer numbers  $\{1, \dots, R\}$ , and  $\otimes$  denotes the outer product of two vectors. For example, the outer product of three vectors  $\mathbf{a}_r \in \mathbb{R}^{n_1}$ ,  $\mathbf{b}_r \in \mathbb{R}^{n_2}$  and  $\mathbf{c}_r \in \mathbb{R}^{n_3}$  forms a third order tensor of dimension  $n_1 \times n_2 \times n_3$  whose  $(i, j, k)^{\text{th}}$  entry is equal to  $a_{ri} \times b_{rj} \times c_{rk}$  where  $a_{ri}$  is the  $i^{\text{th}}$  entry of  $\mathbf{a}_r$ . In (2.2),  $\mathbf{a}_r, \mathbf{b}_r, \mathbf{c}_r$  are of unit norm; that is  $\|\mathbf{a}_r\|_2 = \|\mathbf{b}_r\|_2 = \|\mathbf{c}_r\|_2 = 1$  for all  $r \in [R]$ ;  $\lambda_r \in \mathbb{R}^+$  is the  $r^{\text{th}}$  decomposition weight of the tensor. We denote matrices  $\mathbf{A} \in \mathbb{R}^{n_1 \times R}$ ,  $\mathbf{B} \in \mathbb{R}^{n_2 \times R}$  and  $\mathbf{C} \in \mathbb{R}^{n_3 \times R}$  whose columns are  $\mathbf{a}_r, \mathbf{b}_r$  and  $\mathbf{c}_r$  for  $r \in [R]$  respectively as,

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_R] \quad \mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_R] \quad \mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_R].$$

### 3. COVARIATE ASSISTED SPARSE TENSOR COMPLETION

In this chapter we propose **COSTCO**, an algorithm which aims to complete a sparse tensor with missing data coupled to covariate information matrice(s) along mode(s) of the tensor. The model, optimization problem and algorithm, along with procedures for initialization and parameter tuning are provided in Section 3.1. Section 3.2 presents the main theoretical results. Section 3.5 contains a series of simulation studies and Section 3.6 applies our algorithm to an advertisement data set to illustrate its practical advantages. All proof details are provided in Section 3.7.

#### 3.1 Methodology

In this section we introduce our sparse tensor completion model when covariate information is available and propose a non-convex optimization for parameter estimation. Our algorithm employs an alternative updating approach and incorporates a refinement step to boost the algorithm performance. For conciseness in the proofs derivations, we present this work for the special case in which one covariate matrix is coupled along one mode of the tensor. However our method can be generalised to the case in which all tensor modes are coupled to covariate matrices as is the case in the second part of this dissertation in Chapter 4.

##### 3.1.1 Model

We observe a third-order tensor  $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$  and a covariate matrix  $\mathbf{M} \in \mathbb{R}^{n_1 \times n_v}$  corresponding to the feature information along the first mode of the tensor  $\mathcal{T}$ . Here, without loss of generality, we consider the case where the tensor has three modes and the tensor and the matrix are coupled along the first mode. Our method can be easily extended to the case where more than one mode of the tensor has a covariates matrix.

Let  $\Omega$  be the subset of indexes of the tensor  $\mathcal{T}$  for which entries are not missing. We define a projection function  $P_\Omega(\mathcal{T})$  that projects the tensor onto the observed set  $\Omega$ , such that

$$[P_\Omega(\mathcal{T})]_{ijk} = \begin{cases} \mathcal{T}_{ijk} & \text{if } (i, j, k) \in \Omega \\ 0 & \text{otherwise.} \end{cases} \quad (3.1)$$

In other words  $P_\Omega(\cdot)$  is a function that is applied element-wise to the tensor entries and indicates which entries of the tensor are missing. We assume a noisy observation model, where the observed tensor and matrix are noisy versions of their true counterparts. That is,

$$P_\Omega(\mathcal{T}) = P_\Omega(\mathcal{T}^* + \mathcal{E}_T); \quad \mathbf{M} = \mathbf{M}^* + \mathcal{E}_M, \quad (3.2)$$

where  $\mathcal{E}_T$  and  $\mathcal{E}_M$  are the error tensor and the error matrix respectively;  $\mathcal{T}^*$  and  $\mathbf{M}^*$  are the true tensor and the true matrix, which are assumed to have low-rank decomposition structures [29];

$$\mathcal{T}^* = \sum_{r \in [R]} \lambda_r^* \mathbf{a}_r^* \otimes \mathbf{b}_r^* \otimes \mathbf{c}_r^*; \quad \mathbf{M}^* = \sum_{r \in [R]} \omega_r^* \mathbf{a}_r^* \otimes \mathbf{v}_r^*, \quad (3.3)$$

where  $\lambda_r^*$  and  $\omega_r^* \in \mathbb{R}^+$ , and  $\mathbf{a}_r^* \in \mathbb{R}^{n_1}$ ,  $\mathbf{b}_r^* \in \mathbb{R}^{n_2}$ ,  $\mathbf{c}_r^* \in \mathbb{R}^{n_3}$  and  $\mathbf{v}_r^* \in \mathbb{R}^{n_v}$  with  $\|\mathbf{a}_r^*\|_2 = \|\mathbf{b}_r^*\|_2 = \|\mathbf{c}_r^*\|_2 = \|\mathbf{v}_r^*\|_2 = 1$  for all  $r \in [R]$  with  $R$  representing the rank of the tensor and matrix. As motivated from the online advertisement application, we impose an important sparsity structure on the tensor and matrix components  $\mathbf{a}_r^*$ ,  $\mathbf{b}_r^*$ ,  $\mathbf{c}_r^*$  and  $\mathbf{v}_r^*$  such that they belong to the set  $\mathcal{S}(n, d_i)$  with  $i = 1, 2, 3, v$ , where

$$\mathcal{S}(n, d_i) := \left\{ \mathbf{u} \in \mathbb{R}^{n_i} \mid \|\mathbf{u}\|_2 = 1, \sum_{j=1}^{n_i} 1_{\{\mathbf{u}_j \neq 0\}} \leq d_i \right\}. \quad (3.4)$$

The values  $d_i$  for  $i = 1, 2, 3, v$  are considered to be the true sparsity parameters for the tensor and matrix latent components.

Given a tensor  $\mathcal{T}$  with many missing entries and a covariate matrix  $\mathbf{M}$ , our goal is to recover the true tensor  $\mathcal{T}^*$  as well as its sparse latent components. We formulate the model estimation as a joint sparse matrix and tensor decomposition problem. This comes down to

finding a sparse and low-rank approximation to the tensor and matrix that are coupled in the first mode.

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{V}} \left\{ \|P_{\Omega}(\mathcal{T}) - P_{\Omega}\left(\sum_{r \in [R]} \lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r\right)\|_F^2 + \|\mathbf{M} - \sum_{r \in [R]} \omega_r \mathbf{a}_r \otimes \mathbf{v}_r\|_F^2 \right\} \quad (3.5)$$

subject to  $\|\mathbf{a}_r\|_2 = \|\mathbf{b}_r\|_2 = \|\mathbf{c}_r\|_2 = \|\mathbf{v}_r\|_2 = 1, \|\mathbf{a}_r\|_0 \leq s_1, \|\mathbf{b}_r\|_0 \leq s_2, \|\mathbf{c}_r\|_0 \leq s_3, \|\mathbf{v}_r\|_0 \leq s_v$ .

Here  $s_i, i = 1, 2, 3, v$ , are the sparsity parameters and can be tuned via a data-driven way. The problem in (3.5) is a non-convex optimization when considering all parameters at once, however the objective function is convex in each parameter while other parameters are fixed. Such multi-convex property motivates us to consider an efficient alternative updating algorithm.

### 3.1.2 Algorithm

In order to solve the optimization problem formulated in (3.5), we use an Alternating Least-Squares (ALS) approach and incorporate an extra refinement step as introduced in Jain and Oh [24]. In each iteration of ALS, all but one of the components are fixed and the optimization problem reduces to a convex least-squares problem. to order to enforce  $\ell_0$  norm penalization in the optimization and therefore sparsity, we apply a truncation step after each component update similar to that used in Sun, Lu, Liu, *et al.* [49], Zhang and Han [50], and Hao, Zhang, and Cheng [51]. For a vector  $\mathbf{u} \in \mathbb{R}^n$  and an index set  $F \subseteq [n]$  we define  $\text{Truncate}(\mathbf{u}, F)$  such that its  $i$ -th entry is

$$[\text{Truncate}(\mathbf{u}, F)]_i = \begin{cases} \mathbf{u}_i & \text{if } i \in F \\ 0, & \text{otherwise.} \end{cases}$$

For a scalar  $s < n$ , we denote  $\text{Truncate}(\mathbf{u}, s) = \text{Truncate}(\mathbf{u}, \text{supp}(\mathbf{u}, s))$ , where  $\text{supp}(\mathbf{u}, s)$  is the set of indices of  $\mathbf{u}$  which have the largest  $s$  absolute values. For example, consider  $\mathbf{u} = (0.1, 0.2, 0.5, -0.6)^\top$ , we have  $\text{supp}(\mathbf{u}, 2) = \{3, 4\}$  and  $\text{Truncate}(\mathbf{u}, 2) = (0, 0, 0.5, -0.6)^\top$ . Note that existing sparse tensor models encourage the sparsity either via a Lasso penalized

approach [52], dimension reduction approach [53], or sketching [54]. We extend the truncation-based sparsity approach in traditional high-dimensional vector models [55], [56] and tensor factorization [49]–[51] to the tensor completion problem. As shown in [49], [56], the truncation-based sparsity approach often leads to improved estimation performance in practice.

---

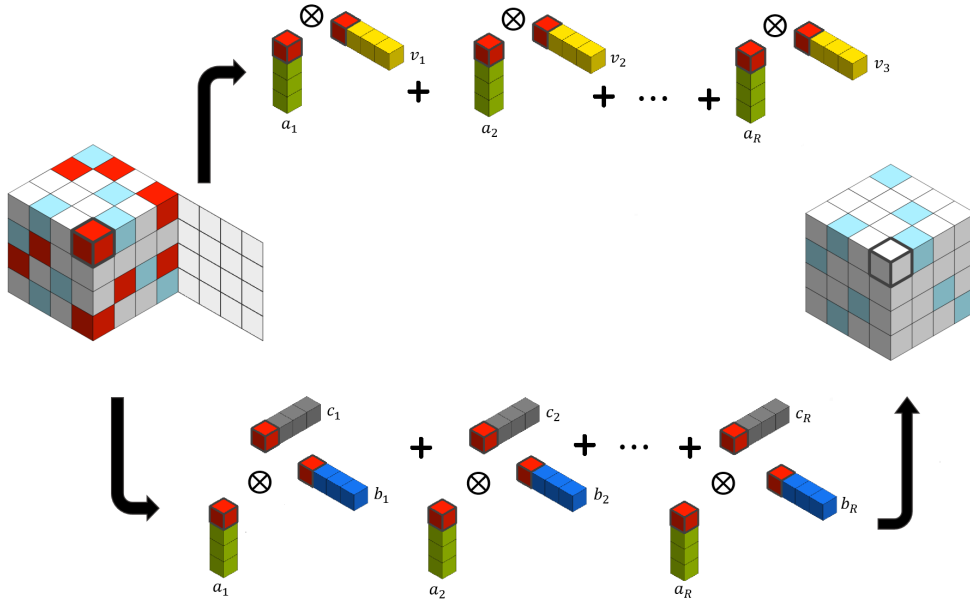
**Algorithm 1** COSTCO: Covariate-assisted Sparse Tensor Completion for Solving (3.5)

---

- 1: **Input:** Observed tensor  $P_\Omega(\mathcal{T}) \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , observed matrix  $\mathbf{M} \in \mathbb{R}^{n_1 \times n_v}$ , maximal number of iterations  $\tau$ , tolerance  $tol$ , rank  $R$ , and cardinality  $(s_1, s_2, s_3, s_v)$ .
  - 2: Initialize  $(\lambda_1, \dots, \lambda_r)$ ,  $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ ,  $(\omega_1, \dots, \omega_r)$ ,  $\mathbf{V}$ .
  - 3:  $\mathbf{a}_r, \mathbf{b}_r, \mathbf{c}_r, \mathbf{v}_r \leftarrow$  the  $r^{\text{th}}$  columns of  $\mathbf{A}, \mathbf{B}, \mathbf{C}$  and  $\mathbf{V}$  respectively,  $\forall r \in [R]$
  - 4: **While**  $t \leq \tau$  and  $\left( \frac{\|\mathbf{A}_{old} - \mathbf{A}\|_F}{\|\mathbf{A}_{old}\|_F} + \frac{\|\mathbf{B}_{old} - \mathbf{B}\|_F}{\|\mathbf{B}_{old}\|_F} + \frac{\|\mathbf{C}_{old} - \mathbf{C}\|_F}{\|\mathbf{C}_{old}\|_F} \right) \geq tol$ ,
  - 5:      $\mathbf{A}_{old} \leftarrow \mathbf{A}, \quad \mathbf{B}_{old} \leftarrow \mathbf{B}, \quad \mathbf{C}_{old} \leftarrow \mathbf{C}, \quad \mathbf{V}_{old} \leftarrow \mathbf{V}$
  - 6:     **For**  $r = 1, \dots, R$
  - 7:          $\text{res}_T \leftarrow P_\Omega(\mathcal{T}) - P_\Omega(\sum_{m \neq r} \lambda_m \mathbf{a}_m \otimes \mathbf{b}_m \otimes \mathbf{c}_m)$      and      $\text{res}_M \leftarrow \mathbf{M} - \sum_{m \neq r} \omega_m \mathbf{a}_m \otimes \mathbf{v}_m$
  - 8:          $\tilde{\mathbf{a}}_r \leftarrow \frac{\lambda_r \text{res}_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r) + \omega_r \text{res}_M \mathbf{v}_r}{\lambda_r^2 P_\Omega(\mathbf{I}, \mathbf{b}_r^2, \mathbf{c}_r^2) + \omega_r^2}$
  - 9:          $\tilde{\mathbf{a}}_r \leftarrow \text{Truncate}(\tilde{\mathbf{a}}_r, s_1), \quad \mathbf{a}_r \leftarrow \tilde{\mathbf{a}}_r / \|\tilde{\mathbf{a}}_r\|_2$
  - 10:          $\tilde{\mathbf{b}}_r \leftarrow \frac{\text{res}_T(\mathbf{a}_r, \mathbf{I}, \mathbf{c}_r)}{P_\Omega(\mathbf{a}_r^2, \mathbf{I}, \mathbf{c}_r^2)}, \quad \tilde{\mathbf{c}}_r \leftarrow \frac{\text{res}_T(\mathbf{a}_r, \mathbf{b}_r, \mathbf{I})}{P_\Omega(\mathbf{a}_r^2, \mathbf{b}_r^2, \mathbf{I})}$      and      $\tilde{\mathbf{v}}_r \leftarrow \text{res}_M^\top \mathbf{a}_r$
  - 11:          $\tilde{\mathbf{b}}_r \leftarrow \text{Truncate}(\tilde{\mathbf{b}}_r, s_2) \quad \tilde{\mathbf{c}}_r \leftarrow \text{Truncate}(\tilde{\mathbf{c}}_r, s_3), \quad \tilde{\mathbf{v}}_r \leftarrow \text{Truncate}(\tilde{\mathbf{v}}_r, s_v)$
  - 12:          $\lambda_r \leftarrow \|\mathbf{c}_r\|_{2_2} \quad \omega_r \leftarrow \|\mathbf{v}_r\|_2$
  - 13:          $\mathbf{b}_r \leftarrow \tilde{\mathbf{b}}_r / \|\tilde{\mathbf{b}}_r\|_2, \quad \mathbf{c}_r \leftarrow \tilde{\mathbf{c}}_r / \|\tilde{\mathbf{c}}_r\|_2, \quad \mathbf{v}_r \leftarrow \tilde{\mathbf{v}}_r / \|\tilde{\mathbf{v}}_r\|_2$
  - 14:     **End For**
  - 15: **End While**
- 

Our COSTCO in Algorithm 1 takes a matrix  $\mathbf{M}$  and a tensor  $\mathcal{T}$  with missing entries as input and computes the components of the matrix and tensor. Due to the non-convexity of the optimization problem, there could be multiple local optima. In our algorithm we initialize the tensor and matrix components using the procedure in Section 3.1.3 which is shown through extensive simulations to provide good starting values for the tensor and matrix components. Line 6 of the algorithm has an inner loop on  $r \in [R]$  which loops on each tensor rank. This inner loop on  $r$  performs an “extra refinement” step that was first introduced in Jain and Oh [24] for tensor completion; and is, therein, proved to improve the error bounds of tensor recovery.

The main component updates are performed in Lines 8 and 10 which are solutions to the least-squares problem while other parameters are fixed. Note that the horizontal double line in Lines 8 and 10 indicate element-wise fraction and the squaring in the denominator



**Figure 3.1.** Illustration of **COSTCO** showing recovery procedure for missing entries through joint tensor matrix decomposition; red cells represent missing entries. The tensor and matrix are coupled along the first mode and the components  $\mathbf{a}_r$ ,  $r \in [R]$  are shared by the tensor and matrix decomposition.

applies entry-wise on the vectors. After obtaining these non-sparse components, Lines 9 and 11 perform the truncation operator to encourage the sparsity on the latent components. The detailed derivation of this algorithm is shown in Lemma 1 in the supplementary material. Finally, the algorithm stops if either the maximum number of iterations  $\tau$  is reached or the normalized Frobenius norm difference of the current and previous components are below a threshold  $tol$ .

Figure 3.1 is an illustration of **COSTCO** that reveals the intuition behind the working of Algorithm 1. As the percentage of missing entries in the tensor increases, recovering the tensor components using only the observed tensor entries leads to a reduction in the accuracy of the recovered tensor components. However, with **COSTCO**, we leverage the additional latent information coming from the matrix of covariates on the shared mode. The signal obtained from the matrix contributes in improving the recovery of the shared components and indirectly that of the non-shared components as well. This observation is reflected on Line 8 of Algorithm 1 for the shared component update, where we see in the denominator

that even when  $P_{\Omega}(\mathbf{I}, \mathbf{b}_r^2, \mathbf{c}_r^2)$  is close to zero (meaning most entries of the tensor are missing) the denominator remains a non-zero value due to the signal from the covariate matrix. In this case we are still able to estimate the shared component  $\mathbf{a}_r$ . This would not be the case without the addition of the covariates matrix information, where the denominator for the update would only be  $P_{\Omega}(\mathbf{I}, \mathbf{b}_r^2, \mathbf{c}_r^2)$  which is close to zero. Therefore, a standalone tensor completion algorithm would become unstable. In the more general case where all three modes of the tensor are coupled to their own covariates matrices, it is easy to see from the illustration in Figure 3.1 that the missing percentage of the tensor could be close to 100%. This is because in such case, the covariates matrix components could still be used in the algorithm to recover the tensor components for all three modes and therefore recover the tensor entries.

### 3.1.3 Initialization Procedure

This section presents details about the method used for the initialization procedure on Line 2 of Algorithm 1. Unlike matrix completion, success in designing an efficient and accurate algorithm for the tensor completion problem is contingent to starting with a good initial estimates. In fact, the convergence rate of low-rank tensor algorithms is typically written as a function of the tensor components weights as well as the initialization error [16], [17], [24], [49], [57]. It is therefore imperative to design an initialization procedure efficient enough to help rule out local stationary points and produce initial component estimates within a local region of the global solution. However such initialization procedures should also be simple enough so not to dominate the computation complexity of the main algorithm.

We use to our advantage, the fact that in our model, the tensor and matrix share at least one mode and use the singular value decomposition (SVD) [58], [59] of the observed matrix  $\mathbf{M}$  to initialize the shared components of the tensor  $\mathbf{A}$  along with the matrix weights  $\omega_1, \dots, \omega_R$  and matrix component  $\mathbf{V}$  respectively. We then use the robust tensor power method (RTPM) from Anandkumar, Ge, Hsu, *et al.* [57] to initialize the non-shared components  $\mathbf{B}$  and  $\mathbf{C}$  and the tensor weights. This is done by setting all missing entries in the tensor to be zero before running RTPM. In practice we show in our simulations in Section 3.5 that this is an adequate

initialization procedure and produces much better initials compared to a random initialization scheme. In the more general case where all tensor modes have covariate matrices, the SVD on the covariate matrices can be used to initialize all the tensor components. In this case, the RTPM for non-shared components initialization would not be needed.

### 3.1.4 Rank and Cardinality Tuning

Our **COSTCO** method relies on two key parameters: the rank  $R$  and the sparsity parameters. It has been shown that exact tensor rank calculation is a NP-hard problem [29]. In this section, following the tuning method in [49], [60], we provide a BIC-type criterion to tune these parameters. Given a pre-specified set of rank values  $\mathcal{R}$  and a pre-specified set of cardinality values  $\mathcal{S}$ , we choose the parameters which minimizes

$$BIC = \log \left( \frac{\|P_\Omega(\mathcal{T} - \sum_{r \in [R]} \lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r)\|_F^2}{n_1 n_2 n_3} + \frac{\|\mathbf{M} - \sum_{r \in [R]} \omega_r \mathbf{a}_r \otimes \mathbf{v}_r\|_F^2}{n_1 n_v} \right) \quad (3.6)$$

$$+ \frac{\log(n_1 n_2 n_3 + n_1 n_v)}{(n_1 n_2 n_3 + n_1 n_v) \sum_{r \in [R]} (\|\mathbf{a}\|_0 + \|\mathbf{b}\|_0 + \|\mathbf{c}\|_0 + \|\mathbf{v}\|_0)}$$

To further speed up the computation, in practice, we tune these parameters sequentially. That is, we first fix  $s_i = n_i$  and tune the rank  $R$  via (3.6). Then given the tuned rank, we tune the sparsity parameters. This tuning procedure works very well through simulation studies in Section 3.5.

## 3.2 Theoretical Analysis

In this section, we derive the error bound of the recovered tensor components obtained from Algorithm 1. We present the recovery results for the estimated shared components  $\mathbf{a}_r$  and non-shared tensor components  $\mathbf{b}_r$  and  $\mathbf{c}_r$  separately to highlight the sharp improvement in recovery accuracy resulting from incorporating the covariate information.

The theory is presented in two phases, first we focus on a simplified case in which the true tensor and matrix components  $\mathbf{a}_r^*$ ,  $\mathbf{b}_r^*$ ,  $\mathbf{c}_r^*$  and  $\mathbf{v}_r^*$  are non-sparse and both tensor and



matrix weights are equal (i.e,  $\omega_r^* = \lambda_r^*$ ,  $\forall r \in [R]$ ). Presenting this simplified case allows us to showcase clearly the interplay between the reveal probability, the tensor and matrix dimensions as well as how the noises in the tensor and matrix affect the statistical and computational errors of the algorithm. In the second case, we then present the results for the general scenario where the tensor and matrix weights are allowed to be unequal and the tensor and matrix components are assumed to be sparse.

### 3.3 Case 1: Non-sparse Tensor and Matrix with Equal Weights

Before presenting the theorem for the simplified case, we introduce assumptions on the true tensor  $\mathcal{T}^*$  and matrix  $\mathbf{M}^*$  and then discuss their utility. Denote  $n := \max(n_1, n_2, n_3, n_v)$ .

#### 3.3.1 Assumptions

**Assumption 1:** (Tensor and matrix structure)

- i. Assume  $\mathcal{T}^*$  and  $\mathbf{M}^*$  are specified as in (3.3) with unique low-rank decomposition up to a permutation, and assume rank  $R = o(n^{1/2})$  and  $\lambda_r^* = \omega_r^*$  (equal weight),  $\forall r \in [R]$ .
- ii. The entries of the decomposed components for both  $\mathcal{T}^*$  and  $\mathbf{M}^*$  satisfy the  $\mu$ -mass condition,

$$\max_r \{ \|\mathbf{a}_r^*\|_\infty, \|\mathbf{b}_r^*\|_\infty, \|\mathbf{c}_r^*\|_\infty, \|\mathbf{v}_r^*\|_\infty \} \leq \frac{\mu}{\sqrt{n}},$$

where  $\mu$  is a constant.

- iii. The components across ranks for both  $\mathcal{T}^*$  and  $\mathbf{M}^*$  meet the incoherence condition,

$$\max_{i \neq j} \left\{ |\langle \mathbf{a}_i^*, \mathbf{a}_j^* \rangle|, |\langle \mathbf{b}_i^*, \mathbf{b}_j^* \rangle|, |\langle \mathbf{c}_i^*, \mathbf{c}_j^* \rangle|, |\langle \mathbf{v}_i^*, \mathbf{v}_j^* \rangle| \right\} \leq \frac{c_0}{\sqrt{n}},$$

where  $c_0$  is a constant.

Assumption (1i) is a common assumption in the tensor decomposition literature to ensure identifiability [24], [29], [49], [57]. It imposes the condition that the tensor admits a low rank CP decomposition that is unique. This is the case of the undercomplete tensor decomposition,

where the rank of the tensor is assumed to be lower than the dimension of the component. The condition  $\lambda_r^* = \omega_r^*$  is a simplification of the problem that allows us to simplify the derivation and sharpen the convergence rate compared to that in the general non-equal weight case (described in Section 3.4). Assumption (1ii) ensures that the mass of the tensor is not contained in only a few entries and is necessary if one hopes to recover any of the non-share components of the tensor with acceptable accuracy. Assumption (1iii) is related to the non-orthogonality of the tensor components and imposes a soft orthogonality condition on the tensor and matrix components. That is, the tensor components are allowed to be correlated only to a certain degree. Anandkumar, Ge, and Janzamin [61] and [49] show that such a condition is met when the tensor and matrix component are randomly generated from a Gaussian distribution. Both the  $\mu$ -mass condition and the incoherence conditions have been commonly assumed in low-rank tensor models [17], [24], [46], [47], [49], [57].

**Assumption 2:** (Reveal probability)

Denote  $\lambda_{min}^* := \min_{r \in [R]} \{\lambda_r^*\}$  and  $\lambda_{max}^* := \max_{r \in [R]} \{\lambda_r^*\}$ . We assume that each entry  $(i, j, k)$  of the tensor  $\mathcal{T}^*$  for all  $i \in [n_1]$ ,  $j \in [n_2]$  and  $k \in [n_3]$  is observed with equal probability  $p$  which satisfies,

$$p \geq \frac{CR^2 \mu^3 \lambda_{max}^{*2} \log^2(n)}{(\lambda_{min}^* + \omega_{min}^*)^2 n^{3/2}},$$

where  $C$  is a constant.

Assumption 2 guarantees that the tensor entries are revealed uniformly at random with probability  $p$ . The lower bound on  $p$  is an increasing function of the tensor rank since recovering tensors with a larger rank is a harder problem which requires more observed entries. The bound on  $p$  is also an increasing function of the  $\mu$ -mass parameter since a larger  $\mu$ -mass parameter in Assumption (1ii) indicates a smaller signal in each tensor entry and hence more reveal entries for accurate component recovery would be needed. Moreover, the bound on  $p$  is a decreasing function of the tensor component dimension  $n$  and relates as  $n^{-3/2}$  up to a logarithm term. This is the optimal dependence on the dimension in tensor completion literature [24], [46]. Most importantly, the lower bound on  $p$  is relaxed when the minimal weight  $\lambda_{min}^*$  of the tensor or the minimal weight  $\omega_{min}^*$  of the matrix increases.

This reflects a critical difference when compared to the lower bound condition required in traditional tensor completion [24], [46] which corresponds to the case  $\omega_{\min}^* = 0$ . It shows the advantage of coupling the matrix of covariates for the tensor completion. This new lower bound on  $p$  translates to requiring less observed entries for the tensor recovery in the presence of covariates. Note that in the present simplified case  $\omega_r^* = \lambda_r^*$ , we still choose to write  $\omega_{\min}^*$  explicitly in the lower bound condition to showcase the effect of the covariate information. The improvement on  $p$  over existing literature will be clearer in Assumption 5 for the general non-equal weight case.

### Assumptions 3 (Initialization error)

Define the initialization errors for the tensor components as  $\epsilon_{0_T} := \max_{r \in [R]} \{ \|\mathbf{a}_r^0 - \mathbf{a}_r^*\|_2, \|\mathbf{b}_r^0 - \mathbf{b}_r^*\|_2, \|\mathbf{c}_r^0 - \mathbf{c}_r^*\|_2, \frac{|\lambda_r^0 - \lambda_r^*|}{\lambda_r^*} \}$  and the initialization error for the matrix components as  $\epsilon_{0_M} := \max_{r \in [R]} \{ \|\mathbf{v}_r^0 - \mathbf{v}_r^*\|_2, \frac{|\omega_r^0 - \omega_r^*|}{\omega_r^*} \}$ . Assume that

$$\epsilon_0 := \max\{\epsilon_{0_T}, \epsilon_{0_M}\} \leq \frac{\lambda_{\min}^*}{100R\lambda_{\max}^*} - \frac{c_0}{3\sqrt{n}}. \quad (3.7)$$

Here the component  $c_0/\sqrt{n}$  is due to the non-orthogonality of the tensor factors. When the components are orthogonal, we allow a larger initialization error. This observation aligns with the common knowledge in tensor recovery as the problem is known to be harder for non-orthogonal tensor factorization [61]. Similarly, a larger rank  $R$  of the tensor leads to a harder problem and a stronger condition on the initialization error. Under Assumption (1i)  $R = o(n^{1/2})$ , when the condition number  $\lambda_{\max}^*/\lambda_{\min}^* = O(1)$ , this initial condition reduces to  $\epsilon_0 = O(1/R)$ . As shown in [24], [61], the robust tensor power method initialization procedure used in our Algorithm satisfies  $O(1/R)$  error bound.

### Assumptions 4 (Signal-to-noise condition)

We assume that spectral norm of the noise tensor and matrix satisfy the following condition

$$\|\mathcal{E}_T\| \leq \frac{\lambda_{\min}^*(p+1)}{p} \quad \text{and} \quad \|\mathcal{E}_M\| \leq \lambda_{\min}^*(p+1). \quad (3.8)$$

Assumption 4 can be considered a variant of the commonly used signal to noise ratio in noisy tensor decomposition.

### 3.3.2 Main Theoretical Results

**Theorem 3.3.1** (Non-sparse tensor and matrix components with equal weights). *Assuming Assumptions 1, 2, 3 and 4 are met. After running  $\Omega\left(\log_2\left(\frac{(p+1)\lambda_{\min}^*\epsilon_0}{p\|\mathcal{E}_T\|+\|\mathcal{E}_M\|}\vee\frac{\lambda_{\min}^*\epsilon_0}{\|\mathcal{E}_T\|}\right)\right)$  iterations of Algorithm 1 with  $s_i = n_i$ , for  $i = 1, 2, 3, v$ , we have*

- **Shared Component  $\mathbf{a}_r$ :**

$$\max_{r \in [R]} (\|\mathbf{a}_r - \mathbf{a}_r^*\|_2) = \mathcal{O}\left(\frac{p\|\mathcal{E}_T\| + \|\mathcal{E}_M\|}{[p+1]\lambda_{\min}^*}\right), \quad (3.9)$$

where  $\|\mathcal{E}_T\|$ ,  $\|\mathcal{E}_M\|$  is the spectral norm of the error tensor and error matrix, respectively.

- **Non-Shared Components  $\mathbf{b}_r$ ,  $\mathbf{c}_r$ :**

$$\max_{r \in [R]} \left( \|\mathbf{b}_r - \mathbf{b}_r^*\|_2, \|\mathbf{c}_r - \mathbf{c}_r^*\|_2, \frac{|\lambda_r - \lambda_r^*|}{\lambda_r^*} \right) = \mathcal{O}\left(\frac{\|\mathcal{E}_T\|}{\lambda_{\min}^*}\right). \quad (3.10)$$

Theorem 3.3.1 indicates that the shared component error is a weighed average of the spectral norm of the error tensor and error matrix. Whereas the non-shared component error is simply a function of the error tensor. In the extreme case in which the covariates matrix  $\mathbf{M}$  is noiseless, then the recovery error of the shared component becomes,

$$\frac{p\|\mathcal{E}_T\|}{\lambda_{\min}^*(p+1)},$$

which is much smaller than the recovery error of the non-shared component  $\frac{\|\mathcal{E}_T\|}{\lambda_{\min}^*}$ . Moreover even in the case in which the coupled covariates matrix is not noiseless, since  $p \leq 1$  we notice an improvement in the statistical error of the recovered shared component compared to that of the non-shared components as long as the spectral norm of the error matrix is no larger than the spectral norm of the error tensor. To see why that is usually the case in practice, recall that  $\mathcal{E}_M \in \mathbb{R}^{n_1 \times n_v}$  and  $\mathcal{E}_T \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ , hence when entries of  $\mathcal{E}_M$  and  $\mathcal{E}_T$  are of the

same scale,  $\|\mathcal{E}_M\|$  is much smaller than  $\|\mathcal{E}_T\|$  as  $n_v$  is much smaller than  $n_2 \times n_3$  in our real application. In the next subsection, we consider the general weight case and will explicitly showcase the improvement of the shared component over traditional completion methods due to the additional covariate information.

### 3.4 Case 2: Sparse Tensor and Matrix with General Weights

We now present the result for the general case with low rank sparse tensor and matrix  $\mathcal{T}^*$  and  $\mathbf{M}^*$  and the weights of the tensor and matrix are allowed to be unequal. The theoretical analysis for the general case is much more challenging than that covered in Theorem 3.3.1. For example, unlike the setting in Case 1, we are no longer able to derive the closed form solution to the optimization problem in (3.5) for the shared tensor component. Instead, we construct an intermediate estimate in the analysis of the shared component recovery. Fortunately, this general result allows us to explicitly quantify the improvement due to the covariates on the missing percentage requirement and the final error bound.

The following conditions are needed for the general scenario. Recall that  $d = \max\{d_1, d_2, d_3, d_v\}$  is the maximal true sparsity parameter defined in (3.4) and define  $s := \max\{s_1, s_2, s_3, s_v\}$ .

#### 3.4.1 Assumptions

**Assumption 5** (sparse tensor and matrix structure)

- i. Assume  $\mathcal{T}^*$  and  $\mathbf{M}^*$  have the sparse structure in (3.3) and (3.4) with unique low-rank decomposition up to a permutation, and assume rank  $R = o(d^{1/2})$ .
- ii. The entries of the decomposed components for  $\mathcal{T}^*$  satisfy the following  $\mu$ -mass condition

$$\max_r \{\|\mathbf{a}_r^*\|_\infty, \|\mathbf{b}_r^*\|_\infty, \|\mathbf{c}_r^*\|_\infty, \|\mathbf{v}_r^*\|_\infty\} \leq \frac{\mu}{\sqrt{d}}.$$

- iii. The components across ranks for both  $\mathcal{T}^*$  and  $\mathbf{M}^*$  meet the incoherence condition,

$$\max_{i \neq j} \left\{ |\langle \mathbf{a}_i^*, \mathbf{a}_j^* \rangle|, |\langle \mathbf{b}_j^*, \mathbf{b}_i^* \rangle|, |\langle \mathbf{c}_j^*, \mathbf{c}_i^* \rangle|, |\langle \mathbf{v}_j^*, \mathbf{v}_i^* \rangle| \right\} \leq \frac{c_0}{\sqrt{d}}.$$

Notice that since the components of tensor and matrix are assumed to be sparse, the  $\mu$ -mass and incoherence condition are functions of the maximum number of non-zero elements  $d$  in the tensor and matrix components rather than the dimension  $n$ . In the case in which  $d \ll n$ , this constitutes a milder assumption compared to Assumptions 1(ii) and 1(iii).

**Assumption 6** (Reveal probability)

We assume that each tensor entry  $(i, j, k)$  for all  $i \in [n_1]$ ,  $j \in [n_2]$  and  $k \in [n_3]$  is observed with equal probability  $p$  which satisfies,

$$p \geq \frac{CR^2\mu^3\lambda_{max}^{*2}\log^2(d)}{(\lambda_{min}^* + \omega_{min}^*)^2d^{3/2}}. \quad (3.11)$$

Similar to the equal-weight case, the required lower bound on the reveal probability in (3.11) improves the established lower bound for the tensor completion with no covariates matrix. Specifically, [24], [62], [63] show that the lower bound for non-sparse tensor completion is of the order  $\frac{\lambda_{max}^{*2}\log^2(n)}{\lambda_{min}^{*2}n^{3/2}}$  while our lower bound is of the order  $\frac{\lambda_{max}^{*2}\log^2(n)}{(\lambda_{min}^* + \omega_{min}^*)^2n^{3/2}}$  when the components are not sparse ( $d = n$ ). This highlights the fact that a weaker assumption on the reveal probability is required in the presence of covariates matrix than in the case with no covariates.

An interesting phenomenon is that when the minimal weight of the matrix  $\omega_{min}^*$  is very large, we could allow the reveal probability to be even close to zero. As demonstrated in our simulations, our COSTCO is still satisfactory even when 98% of the tensor entries are missing, while the traditional tensor completion method fails with more than 90% missing entries. Moreover, in the sparse case, the lower bound is now a decreasing function of the sparsity parameter  $d$ . This is intuitive as when  $d$  decreases, the non-zero tensor components will concentrate on fewer dimensions which makes the tensor recovery problem harder.

**Assumption 7** (Initialization error)

Assume that

$$\epsilon_0 := \max\{\epsilon_{0_T}, \epsilon_{0_M}\} \leq \frac{95/96\lambda_{min}^{*2} + \omega_{min}^{*2}}{144R(\lambda_{max}^{*2} + \omega_{max}^{*2})} - \frac{c_0}{3\sqrt{d}}, \quad (3.12)$$

with  $\epsilon_{0_T}$  and  $\epsilon_{0_M}$  as defined in Assumption 3.

Compared to that in Assumption 3, the initialization condition for Case 2 is slightly stronger. This is reflected on two parts. First, the term  $c_0/\sqrt{d}$  is due to the non-orthogonality of sparse tensor components and is larger in the sparse case. This requires a stronger condition on the rank  $R$  as shown in Assumption (1i) in order to ensure the positivity of the right-hand side of (3.12). Second, the ratio  $(95/96\lambda_{min}^{*2} + \omega_{min}^{*2})/144(\lambda_{max}^{*2} + \omega_{max}^{*2})$  is smaller than  $\lambda_{min}^*/(100\lambda_{max}^*)$  in Assumption 3. Even when  $\lambda_r^* = \omega_r^*$  and  $d = n$ , this condition is still slightly stronger than Assumption 3 since  $\lambda_{min}^{*2}/\lambda_{max}^{*2} < \lambda_{min}^*/\lambda_{max}^*$ . This additional term is due to handling the non-equal weights. Fortunately, when condition numbers  $\lambda_{max}^*/\lambda_{min}^* = O(1)$  and  $\omega_{max}^*/\omega_{min}^* = O(1)$ , we have  $\epsilon_0 = O(1/R)$ , which is again satisfied by the initialization procedure in our algorithm.

#### Assumption 8 (Signal-to-noise condition)

We assume that the sparse spectral norm of the noise tensor and noise matrix satisfy

$$\|\mathcal{E}_T\|_{<d+s>} \leq \frac{\lambda_{min}^{*2}p + \omega_{min}^{*2}}{\lambda_{max}^*p} \quad \text{and} \quad \|\mathcal{E}_M\|_{<d+s>} \leq \frac{\lambda_{min}^{*2}p + \omega_{min}^{*2}}{\omega_{max}^*}. \quad (3.13)$$

Assumption 8 can be considered a variant of the commonly used signal to noise ratio in noisy tensor decomposition with the caveat that the tensor and matrix noise level should be bounded by a function of the tensor and matrix signals (weights).

### 3.4.2 Main Theoretical Results

**Theorem 3.4.1** (Sparse tensor and matrix components with general weights). *Assuming assumptions 5, 6, 7 and 8 are met. After running  $\Omega\left(\log_2\left(\frac{[p\lambda_{min}^{*2} + \omega_{min}^{*2}]\epsilon_0}{p\lambda_{max}^*\|\mathcal{E}_T\|_{<d+s>} + \omega_{max}^*\|\mathcal{E}_M\|_{<d+s>}} \vee \frac{\lambda_{min}^*\epsilon_0}{\|\mathcal{E}_T\|_{<d+s>}\epsilon_T}\right)\right)$  iterations of Algorithm 1 with  $s_i \geq d_i$ , for  $i = 1, 2, 3, v$ , we have*

- **Shared Component  $\mathbf{a}_r$ :**

$$\max_{r \in [R]} (\|\mathbf{a}_r - \mathbf{a}_r^*\|_2) = \mathcal{O}\left(\frac{p\lambda_{max}^*\|\mathcal{E}_T\|_{<d+s>} + \omega_{max}^*\|\mathcal{E}_M\|_{<d+s>}}{p\lambda_{min}^{*2} + \omega_{min}^{*2}}\right), \quad (3.14)$$

where  $\|\mathcal{E}_T\|_{<d+s>}$ ,  $\|\mathcal{E}_M\|_{<d+s>}$  are the sparse spectral norm of error tensor  $\mathcal{E}_T$  and error matrix  $\mathcal{E}_M$ . Remind that the sparse spectral norm is defined in Section 2.2.

- **Non-Shared Components**  $\mathbf{b}_r, \mathbf{c}_r$ :

$$\max_{r \in [R]} \left( \|\mathbf{b}_r - \mathbf{b}_r^*\|_2, \|\mathbf{c}_r - \mathbf{c}_r^*\|_2, \frac{|\lambda_r - \lambda_r^*|}{\lambda_r^*} \right) = \mathcal{O} \left( \frac{\|\mathcal{E}_T\|_{<d+s>}}{\lambda_{\min}^*} \right). \quad (3.15)$$

### 3.4.3 Discussion

Similar to that in Theorem 3.3.1, the statistical error for the shared tensor component in Theorem 3.4.1 is a weighed average of the sparse spectral norm of the error tensor  $\mathcal{E}_T$  and error matrix  $\mathcal{E}_M$ . The key difference is that the weight is now related to  $\lambda_{\max}^*$  and  $\omega_{\max}^*$  and the spectral norm is now much smaller than the non-sparse counterparts in Theorem 3.3.1 since typically  $d + s < n$  and hence  $\|\mathcal{E}_T\|_{<d+s>} \leq \|\mathcal{E}_T\|$ . Similarly, the recovery error for the non-shared tensor component in the general case is also smaller than that in (3.10) due to a smaller spectral norm. This observation highlights the advantage of considering sparse tensor components. In addition, we highlight a few important scenarios in Table 3.1 where the error of shared tensor component is smaller than that of the non-shared component. Such scenario indicates when the additional covariate information is useful to reduce the estimation error of the tensor components. In summary, such improvement is observed when the sparse spectral norm of the error matrix is smaller than or comparable to that of the error tensor. Otherwise, it is not conclusive whether such improvement exists.

## 3.5 Simulations

In this section we evaluate the performance of our COSTCO algorithm via a series of simulations. We compare it with two competing state of the arts methods: **tenALSSparse** by Jain and Oh [24] and **OPT** by Acar, Kolda, and Dunlavy [36]. The algorithm **tenALSSparse** is an alternating minimization based method for tensor completion which incorporates a refinement step in the standard ALS method. In contrast to our method, **tenALSSparse** does not incorporate side covariate information in tensor completion. Comparing our algorithm to



**Table 3.1.** Statistical error of shared tensor component in Theorem 3.4.1 under various conditions. Improvement represent improvement over the recovery error of the non-shared components

Condition Number	Noise	Statistical Error	Improved?
$\frac{\lambda_{max}^*}{\lambda_{min}^*} = O(1)$ $\&$ $\frac{\omega_{max}^*}{\omega_{min}^*} = O(1)$	$\ \mathcal{E}_M\ _{<d+s>} = 0$	$\mathcal{O}\left(\frac{p\ \mathcal{E}_T\ _{<d+s>}}{p\lambda_{min}^* + \omega_{min}^*}\right)$	✓
	$\ \mathcal{E}_M\ _{<d+s>} = \ \mathcal{E}_T\ _{<d+s>}$	$\mathcal{O}\left(\frac{\ \mathcal{E}_T\ _{<d+s>}(p+1)}{p\lambda_{min}^* + \omega_{min}^*}\right)$	✓
	$\ \mathcal{E}_M\ _{<d+s>} < \ \mathcal{E}_T\ _{<d+s>}$	$\mathcal{O}\left(\frac{p\lambda_{max}^*\ \mathcal{E}_T\ _{<d+s>} + \omega_{max}^*\ \mathcal{E}_M\ _{<d+s>}}{p\lambda_{min}^{*2} + \omega_{min}^{*2}}\right)$	✓
	$\ \mathcal{E}_M\ _{<d+s>} > \ \mathcal{E}_T\ _{<d+s>}$	$\mathcal{O}\left(\frac{p\lambda_{max}^*\ \mathcal{E}_T\ _{<d+s>} + \omega_{max}^*\ \mathcal{E}_M\ _{<d+s>}}{p\lambda_{min}^{*2} + \omega_{min}^{*2}}\right)$	inconclusive

**tenALSsparse** helps to highlight the impact of incorporating addition information through coupling with a covariate matrix. It is also worth noting that the original algorithm from Jain and Oh [24] was built for the recovery of non-sparse tensors. In order to allow a fair comparison between our algorithm and theirs, we modify their original algorithm by introducing the same truncation scheme presented in Algorithm 1 to generate the sparse version of their algorithm.

The second comparison method is the **OPT** algorithm, which approaches the coupled matrix and tensor component recovery by solving for all components simultaneously using a gradient-based optimization approach. The all-at-once optimization method is known to be robust to rank mis-specification [14], however it is computationally less efficient than ALS based methods, especially when the tensor is highly missing [31].

In the previous sections, we discuss our models and theories via a third-order tensor to simplify the presentation. Note that **COSTCO** is applicable to tensors with more than three modes. For example, in the simulation studies, we generate a fourth-order tensor  $\mathcal{T}^* \in \mathbb{R}^{d_1 \times 30 \times 30 \times 30}$  and a matrix  $\mathbf{M}^* \in \mathbb{R}^{d_1 \times 30}$ . We assume that the matrix and the tensor share components across the first mode just as is the case in the aforementioned sections. In order to form the tensor  $\mathcal{T}^*$  and the matrix  $\mathbf{M}^*$ , we draw each entry of  $\mathbf{A}^* \in \mathbb{R}^{d_1 \times R}$ ,  $\mathbf{B}^* \in \mathbb{R}^{30 \times R}$ ,  $\mathbf{C}^* \in \mathbb{R}^{30 \times R}$ ,  $\mathbf{D}^* \in \mathbb{R}^{30 \times R}$  and  $\mathbf{V}^* \in \mathbb{R}^{30 \times R}$ , from the iid standard normal distribution.

We enforce sparsity to the tensor components by keeping only the top 40% of the entries in each column in  $\mathbf{B}^*$ ,  $\mathbf{C}^*$  and  $\mathbf{D}^*$  and set the rest of the entries to zero.

In all of our simulations we consider the coupled modes  $\mathbf{A}^*$  to be dense to mimic the real data scenario in Section 3.6 where the coupled matrix is dense. We define  $\lambda_1^*, \dots, \lambda_R^*$  and  $\omega_1^*, \dots, \omega_R^*$  as the product of the non-normalized component norms in each mode, that is,  $\lambda_r^* = \|\mathbf{a}_r^*\|_2 \times \|\mathbf{b}_r^*\|_2 \times \|\mathbf{c}_r^*\|_2 \times \|\mathbf{d}_r^*\|_2$  and  $\omega_r^* = \|\mathbf{a}_r^*\|_2 \times \|\mathbf{v}_r^*\|_2$ .

We then normalize each of the columns of  $\mathbf{A}^*$ ,  $\mathbf{B}^*$ ,  $\mathbf{C}^*$ ,  $\mathbf{D}^*$ ,  $\mathbf{V}^*$  to unit norm. To illustrate, the first mode component matrix  $\mathbf{A}^*$  becomes  $\mathbf{A}^* = [\frac{\mathbf{a}_1^*}{\|\mathbf{a}_1^*\|_2}, \dots, \frac{\mathbf{a}_R^*}{\|\mathbf{a}_R^*\|_2}]$ . The sparse tensor  $\mathcal{T}^*$  and matrix  $\mathbf{M}^*$  are then formed as  $\mathcal{T}^* = \sum_{r \in [R]} \lambda_r^* \mathbf{a}_r^* \otimes \mathbf{b}_r^* \otimes \mathbf{c}_r^* \otimes \mathbf{d}_r^*$  and  $\mathbf{M}^* = \sum_{r \in [R]} \omega_r^* \mathbf{a}_r^* \otimes \mathbf{v}_r^*$ . We then add noise to the tensor and matrix using the following setup  $\mathcal{T} = \mathcal{T}^* + \eta_T \mathcal{N}_T \frac{\|\mathcal{T}^*\|_F}{\|\mathcal{N}_T\|_F}$  and  $\mathbf{M} = \mathbf{M}^* + \eta_M \mathcal{N}_M \frac{\|\mathbf{M}^*\|_F}{\|\mathcal{N}_M\|_F}$ , where  $\mathcal{N}_T$  and  $\mathcal{N}_M$  are a tensor and a matrix of the same size as  $\mathcal{T}^*$  and  $\mathbf{M}^*$  respectively, whose entries are generated from the standard normal distribution. A similar noise generation procedure has been considered in Acar, Kolda, and Dunlavy [36]. We simulate the uniformly missing at random pattern in the tensor data by generating entries of the reveal tensor  $\mathbf{\Omega} \in \mathbb{R}^{d_1 \times 30 \times 30 \times 30}$  from the binomial distribution with reveal probability  $p$ . The sparse and noisy tensor  $P_\Omega(\mathcal{T})$  with missing data is finally obtained as  $P_\Omega(\mathcal{T}) = \mathcal{T} * \mathbf{\Omega}$ , where  $*$  is the element-wise multiplication.

To assess the goodness of fit for the tensor and tensor components recovery, we use the normalized Frobenius norm of the difference between the recovered component and the true component. We compute the tensor estimation error, the tensor component error and tensor weights error as:

$$\begin{aligned} \text{tensor error} &:= \|\mathcal{T}^* - \mathcal{T}\|_F / \|\mathcal{T}^*\|_F; \quad \text{component error} := \|\mathbf{U}^* - \mathbf{U}\|_F / \|\mathbf{U}^*\|_F; \\ \text{weight error} &:= \|\boldsymbol{\lambda}^* - \boldsymbol{\lambda}\|_2 / \|\boldsymbol{\lambda}^*\|_2, \end{aligned} \tag{3.16}$$

where  $\mathcal{T}$ ,  $\mathbf{U}$ , are the estimated tensor and tensor components with  $\mathbf{U} \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ , and  $\boldsymbol{\lambda} := (\lambda_1, \dots, \lambda_R)^\top$  is the vector of estimated tensor weights returned by Algorithm 1. In all simulations we return the mean error of 30 replicas of each experiment. Throughout all the experiments, we set the maximum number of iterations  $\tau$  to be 200, the tolerance  $tol$  in Algorithm 1 is set to be  $1e^{-7}$ . To avoid bad local solutions, we conduct 10 initializations for

each replicate in all methods. We set the tuning range for the rank  $R$  to be  $\{1, 2, 3, 4, 5\}$ . The tuning range for the sparsity is set to be  $\{20\%, 40\%, 60\%, 80\%, 90\%, 100\%\}$ , each value representing the percentage of non-zero entries in the latent components as performed on Lines 9 and 11 of Algorithm 1.

### 3.5.1 Missing Percent

In this first simulation we consider the case with varying levels of missing percentages. We set the dimension of the couple mode to be  $d_1 = 30$  and therefore generate  $P_\Omega(\mathcal{T}) \in \mathbb{R}^{30 \times 30 \times 30 \times 30}$ . We set the rank to be  $R = 2$  and the noise level  $\eta_T, \eta_M$  to be both 0.001. We measure the recovery error under four different settings of the reveal probability parameter  $p = \{0.2, 0.1, 0.05, 0.01\}$ . In other words, 80%, 90%, 95% and 99% of the tensor entries are missing in each setting.

Table 3.2 indicates that under all varying missing probability, our **COSTCO** algorithm provides a better fit in tensor recovery relative to **tenALSp** and **OPT**. Notably, with a higher level of missing data, missing percentage  $\geq 90$  **COSTCO** significantly outperforms both **tenALSp** and **OPT** methods of tensor recovery. This is more evident when we compare our algorithm to **tenALSp** for the case where missing percentage ranges from 90% to 98%; in these scenarios the recovery error of **COSTCO** is at least 10 folds better than that of **tenALSp**. This agrees with the two advantages of incorporating covariate information into tensor completion as we discussed in the theoretical results: (1) allowing higher missing percentage; (2) reducing estimation errors. Moreover, we notice that the estimation error for the shared component  $\text{Comp } \tilde{\mathbf{A}}$  is better than that of the non-shared components. This also aligns with the theoretical result which shows that the recovery of the couple component improves over that of non-coupled components due to additional covariate information. Finally, although **OPT** also uses coupling, it underperforms compared to **COSTCO** because the all at once optimization method suffers with unstable gradient when the missing entry percentage is large.

**Table 3.2.** Estimation errors with varying missing percentages. Reported values are the average and standard deviation (in parentheses) of tensor, tensor components and weight recovery error based on 30 data replications. **COSTCO**: the proposed method; **tenALSsparse**: sparse version of the tensor completion method by [24]; **OPT**: the gradient based all at once optimization method of [36]; symbol ( $\hat{A}$ ) used to put shared tensor-matrix component **A** in emphasis.

Missing %	Component	Estimation Error		
		COSTCO	tenALSsparse	OPT
80%	$\mathcal{T}$	<b>3.38e-05 (2.36e-12)</b>	3.66e-05 (2.73e-12)	3.56e-05 (2.31e-12)
	Comp $\hat{\mathbf{A}}$	<b>1.52e-05 (2.37e-12)</b>	2.22e-05 (3.93e-12)	<b>1.52e-05 (2.36e-12)</b>
	Comp <b>B</b>	<b>2.12e-05 (4.39e-12)</b>	2.13e-05 (3.64e-12)	2.26e-05 (5.05e-12)
	Comp <b>C</b>	<b>1.98e-05 (4.69e-12)</b>	1.99e-05 (4.83e-12)	2.24e-05 (4.35e-12)
	Comp <b>D</b>	<b>2.17e-05 (2.92e-12)</b>	2.18e-05 (2.78e-12)	2.26e-05 (2.99e-12)
	$\lambda$	1.18e-06 (4.67e-13)	<b>1.17e-06 (4.95e-13)</b>	1.18e-06 (4.67e-13)
90%	$\mathcal{T}$	<b>3.93e-05 (6.12e-12)</b>	4.47e-02 (2.71e-11)	4.94e-05 (6.07e-12)
	Comp $\hat{\mathbf{A}}$	<b>1.80e-05 (2.79e-12)</b>	5.65e-02 (2.74e-11)	<b>1.80e-05 (2.82e-12)</b>
	Comp <b>B</b>	<b>2.16e-05 (1.31e-11)</b>	4.84e-02 (2.02e-11)	3.17e-05 (1.31e-11)
	Comp <b>C</b>	<b>2.12e-05 (9.54e-12)</b>	4.96e-02 (3.22e-11)	3.13e-05 (9.75e-12)
	Comp <b>D</b>	<b>2.17e-05 (1.38e-11)</b>	5.79e-02 (2.00e-11)	3.18e-05 (1.39e-11)
	$\lambda$	<b>1.65e-06 (7.98e-13)</b>	4.84e-02 (8.31e-13)	1.65e-06 (7.98e-13)
95%	$\mathcal{T}$	<b>5.69e-05 (1.92e-11)</b>	1.19e-01 (8.70e-03)	6.93e-05 (1.90e-11)
	Comp $\hat{\mathbf{A}}$	1.92e-05 (5.60e-12)	1.44e-01 (2.01e-02)	<b>1.50e-05 (6.30e-12)</b>
	Comp <b>B</b>	<b>3.44e-05 (2.29e-11)</b>	1.28e-01 (1.61e-02)	4.45e-05 (2.30e-11)
	Comp <b>C</b>	<b>3.39e-05 (3.36e-11)</b>	1.30e-01 (1.02e-02)	4.39e-05 (3.34e-11)
	Comp <b>D</b>	<b>3.74e-05 (1.84e-11)</b>	1.40e-01 (1.39e-02)	4.74e-05 (1.80e-11)
	$\lambda$	<b>1.26e-06 (8.99e-13)</b>	1.25e-01 (1.08e-02)	1.76e-06 (8.99e-13)
98%	$\mathcal{T}$	<b>2.36e-02 (3.50e-11)</b>	5.05e-01 (1.75e-02)	5.02e-02 (1.98e-02)
	Comp $\hat{\mathbf{A}}$	<b>2.17e-02 (1.18e-11)</b>	6.58e-01 (2.03e-02)	6.87e-02 (2.61e-03)
	Comp <b>B</b>	<b>2.63e-02 (5.60e-11)</b>	6.18e-01 (1.29e-02)	6.31e-02 (2.95e-02)
	Comp <b>C</b>	<b>2.58e-02 (5.81e-11)</b>	5.89e-01 (1.49e-02)	6.27e-02 (3.86e-02)
	Comp <b>D</b>	<b>2.16e-02 (5.39e-11)</b>	5.94e-01 (2.16e-02)	6.96e-02 (2.03e-02)
	$\lambda$	<b>2.14e-02 (5.67e-13)</b>	5.19e-01 (1.75e-02)	5.00e-02 (2.14e-02)
99%	$\mathcal{T}$	<b>7.13e-01 (5.93e-11)</b>	9.99e-01 (5.35e-02)	8.80e-01 (2.33e-02)
	Comp $\hat{\mathbf{A}}$	<b>3.60e-01 (1.28e-10)</b>	1.17e+00 (1.17e-01)	4.17e-01 (4.39e-02)
	Comp <b>B</b>	<b>7.40e-01 (1.04e-10)</b>	1.14e+00 (9.65e-02)	7.94e-01 (3.70e-02)
	Comp <b>C</b>	<b>8.25e-01 (3.75e-11)</b>	1.17e+00 (9.15e-02)	9.14e-01 (3.65e-02)
	Comp <b>D</b>	<b>5.90e-01 (4.57e-11)</b>	9.77e-01 (9.83e-02)	7.12e-01 (4.51e-02)
	$\lambda$	<b>6.48e-01 (5.73e-11)</b>	9.77e-01 (6.04e-02)	8.68e-01 (2.33e-02)

### 3.5.2 Noise Level

In the next set of experiments we vary the noise level parameter for the tensor  $\eta_T$  and noise level for the matrix  $\eta_M$  to test algorithms' robustness to noise. These two parameters control the signal-to-noise ratio in the model. The missing probability for these experiments

is set to 90% and tensor rank and sparsity of the true tensor are set to  $R = 2$  and 60% respectively.

As can be seen in Table 3.3, when the tensor noise  $\eta_T$  is greater than that of the matrix noise  $\eta_M$ , our algorithm outperforms the two competing methods with a large gap in recovery error. Even when the matrix has a slightly larger noise level than the tensor ( $\eta_M = 0.01, \eta_T = 0.001$ ), **COSTCO** still outperforms the other two algorithms. It shows that in high missing data regime coupling a matrix that has a slightly larger noise than the tensor still provides enough information to improve the tensor recovery rate. On the other hand, when the matrix noise level is much higher than that of the tensor ( $\eta_M = 0.1, \eta_T = 0.001$  in Table 3.3), we observe that our algorithm **COSTCO** and the other coupled algorithm **OPT** are inferior compared to **tenALSsparse**. In this case, the recovery of the shared component **A** suffers the most in **COSTCO** and **OPT** and is responsible for the inferior tensor recovery error compared to **tenALSsparse** which does not use the coupled matrix. This is expected as a matrix with much larger noise than that of a tensor no longer brings in enough signals in the coupling and therefore makes the tensor completion problem harder than when the matrix is completely omitted from the model. Finally, an interesting phenomenon is that the noise level of the error matrix  $\eta_M$  only affects the estimation error of the shared component but not those of the non-shared components. To see it, in the last two settings in Table 3.3, when  $\eta_T$  is fixed and  $\eta_M$  increases, only the recovery accuracy of the shared component **A** significantly drops, but those of the non-shared components have no significant changes. However, in the first two settings in Table 3.3, when  $\eta_M$  is fixed and  $\eta_T$  increases, the recovery accuracy of both shared and non-shared components significantly drops. These findings agree well with our theoretical results in Theorem 3.4.1.

In the following two additional simulations, we focus solely on the recovery accuracy of the shared and non-shared tensor components under our **COSTCO** to investigate the practical effect of component dimensions size and the rank on our algorithm.

**Table 3.3.** Estimation errors with varying noise levels of error matrix and error tensor. Reported values are the average and standard deviation (in parentheses) of estimation errors. **COSTCO**: the proposed method; **tenALSparse**: sparse version of the tensor completion method by [24]; **OPT**: the gradient based all at once optimization method of [36].

Noise Level	Component	Estimation Error		
		COSTCO	tenALSparse	OPT
$\eta_M = 0.001$ $\eta_T = 0.01$	$\mathcal{T}$	<b>2.74e-04 (7.31e-10)</b>	5.37e-04 (1.00e-09)	4.74e-04 (7.31e-10)
	Comp $\ddot{A}$	<b>1.05e-04 (2.24e-10)</b>	3.17e-04 (1.13e-09)	1.05e-04 (2.24e-10)
	Comp $\mathbf{B}$	<b>2.13e-04 (8.03e-10)</b>	3.10e-04 (4.72e-10)	3.13e-04 (8.03e-10)
	Comp $\mathbf{C}$	<b>2.15e-04 (1.33e-09)</b>	3.14e-04 (1.35e-09)	3.15e-04 (1.33e-09)
	Comp $\mathbf{D}$	<b>2.21e-04 (1.43e-09)</b>	3.22e-04 (1.69e-09)	3.21e-04 (1.43e-09)
	$\lambda$	<b>1.41e-05 (6.77e-11)</b>	1.48e-05 (7.44e-11)	<b>1.41e-05 (6.77e-11)</b>
$\eta_M = 0.001$ $\eta_T = 0.1$	$\mathcal{T}$	<b>2.73e-03 (5.50e-08)</b>	5.36e-03 (8.04e-08)	4.73e-03 (5.50e-08)
	Comp $\ddot{A}$	<b>1.06e-03 (2.39e-08)</b>	3.16e-03 (1.87e-07)	1.06e-03 (2.39e-08)
	Comp $\mathbf{B}$	<b>2.03e-03 (1.25e-07)</b>	3.00e-03 (1.66e-07)	3.03e-03 (1.25e-07)
	Comp $\mathbf{C}$	<b>2.15e-03 (6.21e-08)</b>	3.10e-03 (3.68e-08)	3.15e-03 (6.21e-08)
	Comp $\mathbf{D}$	<b>2.20e-03 (1.02e-07)</b>	3.23e-03 (1.01e-07)	3.20e-03 (1.02e-07)
	$\lambda$	1.52e-04 (7.07e-09)	<b>1.46e-04 (6.09e-09)</b>	1.52e-04 (7.07e-09)
$\eta_M = 0.01$ $\eta_T = 0.001$	$\mathcal{T}$	<b>3.88e-04 (5.55e-10)</b>	5.35e-04 (6.41e-10)	4.88e-04 (5.55e-10)
	Comp $\ddot{A}$	<b>1.74e-04 (3.79e-10)</b>	3.21e-04 (8.24e-10)	1.74e-04 (3.82e-10)
	Comp $\mathbf{B}$	<b>2.17e-04 (9.18e-10)</b>	3.14e-04 (1.10e-09)	3.17e-04 (9.18e-10)
	Comp $\mathbf{C}$	<b>2.16e-04 (1.13e-09)</b>	3.16e-04 (1.44e-09)	3.16e-04 (1.13e-09)
	Comp $\mathbf{D}$	<b>2.07e-04 (8.39e-10)</b>	3.02e-04 (8.70e-10)	3.07e-04 (8.39e-10)
	$\lambda$	<b>1.49e-05 (7.21e-11)</b>	1.53e-05 (6.63e-11)	<b>1.49e-05 (7.21e-11)</b>
$\eta_M = 0.1$ $\eta_T = 0.001$	$\mathcal{T}$	9.75e-04 (1.60e-08)	<b>5.37e-04 (1.36e-09)</b>	1.28e-03 (1.60e-08)
	Comp $\ddot{A}$	1.39e-03 (2.27e-08)	<b>3.17e-04 (1.16e-09)</b>	1.39e-03 (2.27e-08)
	Comp $\mathbf{B}$	<b>2.20e-04 (1.11e-09)</b>	3.09e-04 (1.02e-09)	3.21e-04 (1.12e-09)
	Comp $\mathbf{C}$	<b>2.29e-04 (1.30e-09)</b>	3.19e-04 (1.01e-09)	3.23e-04 (1.32e-09)
	Comp $\mathbf{D}$	<b>2.24e-04 (1.20e-09)</b>	3.12e-04 (1.27e-09)	3.25e-04 (1.20e-09)
	$\lambda$	<b>1.26e-05 (7.94e-11)</b>	1.27e-05 (7.62e-11)	<b>1.26e-05 (7.94e-11)</b>

### 3.5.3 Component Size

This part of the simulation considers the effect of varying the size of the coupled components  $\mathbf{A}^*$  of the true tensor on the tensor recovery. We set the tensor missing entry percentage to be 90%; the noise level parameters are set to be  $\eta_T = 0.001$  and  $\eta_M = 0.001$  respectively and the sparsity level is kept at 60%. The complete simulation results are presented in Table 3.4. The tensor completion error improves with increasing size of the shared dimension since there is more information provided by the covariate matrix. With more and more information provided from the covariate matrix, the latent structure of the shared component dominates those of the non-shared components, making it easier to complete the whole tensor.

**Table 3.4.** Estimation errors of COSTCO with varying coupled dimension  $d_1$ .

Coupled Dimension $d_1$	Estimation Error					
	$\mathcal{T}$	Comp $\mathbf{A}$	Comp $\mathbf{B}$	Comp $\mathbf{C}$	Comp $\mathbf{D}$	$\boldsymbol{\lambda}$
20	5.64e-05 (1.24e-11)	1.77e-05 (6.09e-12)	3.67e-05 (1.41e-11)	3.51e-05 (1.88e-11)	3.68e-05 (2.20e-11)	1.60e-06 (6.09e-13)
50	3.71e-05 (3.29e-12)	1.72e-05 (2.66e-12)	2.35e-05 (2.59e-12)	2.39e-05 (4.06e-12)	2.44e-05 (4.72e-12)	1.25e-06 (5.14e-13)
100	2.66e-05 (1.43e-12)	1.73e-05 (5.69e-13)	1.72e-05 (2.86e-12)	1.76e-05 (3.50e-12)	1.77e-05 (1.96e-12)	7.65e-07 (1.34e-13)

### 3.5.4 Rank

In this case we investigate the impact of the rank of the tensor and matrix on the tensor recovery performance of our COSTCO algorithm. We set the missing percentage of the tensor to 90%, the sparsity to be 60% and the tensor and matrix noise levels  $\eta_T$  and  $\eta_M$  to be both 0.001. We still tune the rank and cardinality using the procedure in Section 3.1.4. As shown in Table 3.5, the recovery error is an increasing function of the tensor rank. It is well documented that the noisy tensor completion problem in general gets harder as the rank increases [14]. This result also aligns well with the theoretical derivation provided in Section 3.2. In Assumption 7, we see that the initialization error is a decreasing function of the rank  $R$ . Hence tensor with larger  $R$  requires the initialization algorithm to be more accurate than tensors with smaller ranks.

**Table 3.5.** Estimation errors of COSTCO with varying rank.

Tensor Rank	Estimation Error					
	$\mathcal{T}$	Comp <b>A</b>	Comp <b>B</b>	Comp <b>C</b>	Comp <b>D</b>	$\lambda$
1	4.78e-05	2.76e-05	1.97e-06	2.77e-05	2.62e-05	5.31e-06
	(1.34e-11)	(1.67e-11)	(6.72e-14)	(7.50e-12)	(1.38e-11)	(1.29e-11)
2	6.50e-05	6.78e-05	1.39e-05	6.63e-05	6.66e-05	1.26e-05
	(1.04e-11)	(6.82e-11)	(4.67e-11)	(5.07e-11)	(7.16e-11)	(3.76e-11)
3	8.57e-05	7.82e-05	2.76e-05	7.99e-05	7.81e-05	1.32e-05
	(2.52e-11)	(5.27e-11)	(1.11e-10)	(8.10e-11)	(5.97e-11)	(4.14e-11)

### 3.6 Real Data Analysis

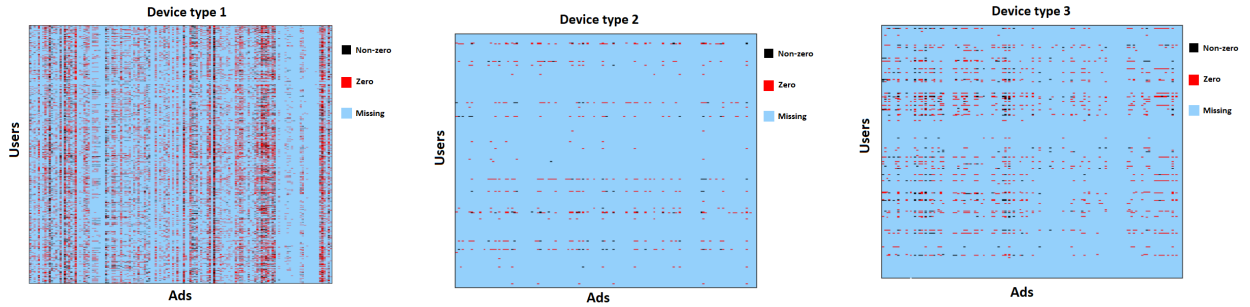
We apply our COSTCO method to an advertisement (ad) data to showcase its practical advantages. COSTCO makes use of multiple sources of ad data to extract the ad latent component which is a comprehensive representation of ads. We demonstrate that the obtained ad latent components are able to deliver interesting ad clustering results that are not achievable by a stand-alone method.

Online advertising is a type of marketing strategy that uses the internet to promote a given product to potential customers. Extracting patterns in data gathered from online advertisement allows ad platforms and companies to churn data into knowledge, which is then used to improve customer satisfaction. Clustering algorithms have been applied to the ad data to discover ad or user clusters for better ad targeting. After computing the similarity between the new ad and each ad cluster, the ad agency can determine whether a new ad should be assigned to a specific user group. Most ad-user clustering research focuses on a single correlation matrix. What makes our method different is that we not only have a third-order user-by-ad-by-device click tensor data but we also possess additional information which describe specific features of ads. Our COSTCO algorithm uses both click tensor data and ad matrix data to extract the ad latent component for better ad clustering.

The data we analyze in this section is advertising data collected from a major internet company for 4 weeks in May-June 2016. A user preference tensor was obtained by tracking the behavior of 1000 users on 140 ads accessed through 3 different devices. The  $1000 \times 140 \times 3$  tensor is formed by computing the click-through-rate (CTR) of each (user, ad, device) triplet



over the four weeks period; which is the number of times a user has clicked an ad from a certain device divided by the number of times the user has seen that ad from the specific device. As illustrated in Figure 3.2, this ad CTR tensor has 96% missing entries and is highly sparse with only 40% of the revealed entries being nonzero. A missing entry in the ad CTR data occurs when a given user is not presented with a certain ad from a specific device, while zeros (sparsity) in the ad CTR data are used to represent user choosing not to interact with an ad that was presented to them on a specific device.



**Figure 3.2.** Illustration of missing data and sparsity in our ad CTR tensor.

Beside the ad CTR tensor, we also have access to the ad text raw data that store the content of all ads. We use Latent Dirichlet Allocation (LDA) [64] to process the ad text data. LDA is an unsupervised topic modeling algorithm that attempts to describe a set of text observations as a mixture of different topics. We first follow Blei, Ng, and Jordan [64] to tune the parameters of LDA such as the number of topics and the Dirichlet distribution parameter that give the best trade-off between low perplexity value and efficient computing time. The best perplexity is obtained for 20 topics. This means that all the 140 advertisement data can be considered as a combination of 20 topics. Due to space constraints, we illustrate an example of 7 out of 20 topics in Table 3.6, and only display the top 10 words for each of the 7 topics returned by LDA. Each topic column was labeled based on overall meaning of the top words. Once trained, LDA returns a matrix that contains the proportion of topics in each ad. We use this matrix of proportions of dimension  $\mathbb{R}^{140 \times 20}$  as the ad covariate matrix that will be used jointly with the ad CTR tensor to obtain ad latent components in our COSTCO algorithm.

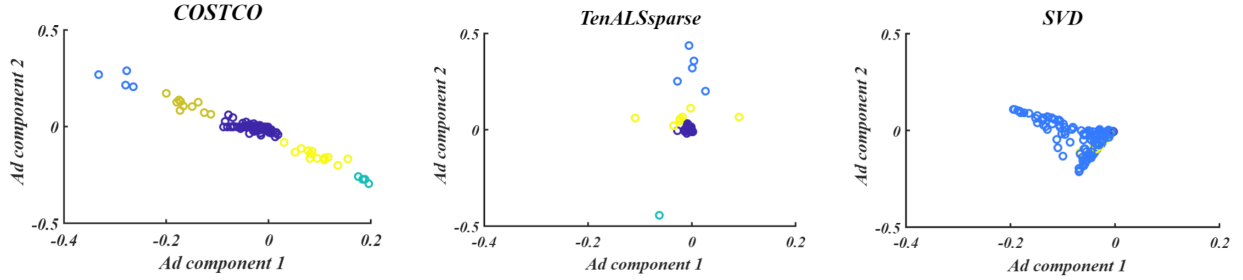
**Table 3.6.** Top ten words for 7 chosen topics. Top words were obtained through LDA.

Topics	Ride	Gaming	Security	Mortgage	Insurance	Online dating	Fashion retail
Top Words	uber	game	vivint	mortgage	get	single	buy
	pay	controller	home	apr	insurance	pic	sale
	car	experience	front	payment	less	man	gilt
	people	gameplay	security	free	see	profile	zulily
	weekly	accessory	smart	new	month	click	lulus
	fare	ebay	call	arm	drive	meet	charlotterusse
	ride	level	control	quotes	day	browse	neimanmarcus
	give	time	camera	calculate	miles	look	maurices
	work	joystick	adt	easy	low	free	lastcall
	drive	wide	look	process	qualify	pay	spring

We first evaluate the tensor recovery error by randomly splitting the observed tensor entries into 80% training and 20% testing. Let  $\hat{\mathcal{T}}$  indicate the recovered tensor from the training set. We use  $\hat{\mathcal{T}}$  for training and compute the recovery error on the testing set. The metrics used to access the recovery error of the tensor is defined as  $\|P_{\Omega_{Test}}(\mathcal{T} - \hat{\mathcal{T}})\|_F / \|P_{\Omega_{Test}}(\mathcal{T})\|_F$ , where  $P_{\Omega_{Test}}(\mathcal{T}) = \Omega_{Test} * \mathcal{T}$  with  $\Omega_{Test}$  being a binary tensor of the same size as  $\mathcal{T}$  that has ones on the test entries and zeros elsewhere. The tensor recovery error for **COSTCO** is 0.825, leading to 23% accuracy improvement over the baseline **tenALSsparse** whose error is 1.083. This again highlights the benefit of fusing the ad content matrix to the ad CTR tensor. The OPT algorithm was not used for comparison as the algorithm optimization package failed with error messages after multiple trials on this data. We conjecture this is due to the unstable performance of the all at once optimization when the missing percentage is very high.













We then compare the ad latent components returned from **COSTCO** and **tenALSsparse** in Figure 3.3. As a comparison, we also include the result of SVD which directly decomposes the ad covariate matrix data. The ad clusters shown in Figure 3.3 are obtained by applying the K-means clustering algorithm to the ad latent component data from each method. As shown in Figure 3.3, the first two columns of the latent components returned from our **COSTCO** show a clear clustering structure with 5 clusters. On the other hand, the ad components extracted from **tenALSsparse** are all clustered around zeros. This is because the ad CTR tensor is highly sparse and the latent components based on decomposing the tensor itself

contain many small values. Therefore, ad clusters generated using `tenALSsparse` tend to have very large and very small clusters.



**Figure 3.3.** Scatter plot of the ad latent components obtained from three methods. Different clusters are represented via different colors.

Finally, Figure 3.4 demonstrates some interesting ad clustering results obtained from our `COSTCO` algorithm which links different ad industries into the same cluster. For example based on cluster 1 from `COSTCO`, ads about male and female online dating are clustered together with ads about women retail stores and man clothing accessories. In cluster 2 from `COSTCO`, ads about weight lost and weight lost surgery are clustered together with ads about gourmet cuisine and restaurant which indicates that users who interact with weight loss ads are also interested in nutrition related ads. Cluster 3 of `COSTCO` contains ads about house mortgage, home security devices, auto, home and auto insurance, house weather control devices which indicates that users that are homeowners tend to be interested in home and auto related things. These interesting clusters are not obtained in the SVD method nor the `tenALSsparse` method. The clusters from SVD are solely related to the topic of each ad as shown in Figure 3.4 and the clusters from `tenALSsparse` are highly unbalanced and do not contain any understandable relationship between ads. These clustering results illustrate the practical value of our `COSTCO` method. By incorporating ad covariate matrix into the completion of the ad CTR tensor, we are able to obtain a more synthetic description of ads and find interesting links between different advertising industries, which directly helps the marketing team to strategize the ad planing procedure accordingly for better ad targeting.

Method	Topics clusters using COSTCO			
COSTCO cluster 1				
COSTCO cluster 2				
COSTCO cluster 3				
Topics clusters using SVD				
SVD cluster1				
SVD cluster 2				

**Figure 3.4.** Result of ad clusters obtained using different methods

### 3.7 Proof of Main Theorems

In this section we provide the proofs of the main theoretical results presented in 3.3.1 and 3.4.1. As elaborated in the discussion paragraphs in section 3.2 proving first the particular case in theorem 3.3.1 allows for a better presentation and explanation for the proof technique used for the general case in theorem 3.4.1. For simplicity, in the following proofs we consider the case where all tensor and matrix modes have the same dimensions  $n$  that is  $n_1 = n_2 = n_3 = n_v = n$ . We also assume that the sparsity parameters for each mode are equal ( $d_1 = d_2 = d_3 = d_v = d$ ). It follows from the two simplification aforementioned that in Algorithm 1 we let  $s_1 = s_2 = s_3 = s_v = s$ . Proving the case, in which the dimensions of the tensor and matrix' modes are allowed to be unequal is a trivial yet notation heavy extension of the technique we use in the proof of Theorem 3.3.1 and Theorem 3.4.1. As defined in equation (3.17), we use the euclidean distance between the component estimates and true components to measure the error for component recovery. We also use the relative absolute difference between estimated and true weights to capture the recovery error for the weights as defined in equation (3.18). Define  $\mathbf{d}_{u_r}$  to be,

$$\mathbf{d}_{u_r} =: \mathbf{u}_r - \mathbf{u}_r^*, \quad \text{and} \quad \|\mathbf{d}_{u_r}\|_2 = \|\mathbf{u}_r - \mathbf{u}_r^*\|_2, \quad (3.17)$$

and

$$\Delta_{\lambda_r} := \left| \frac{\lambda_r - \lambda_r^*}{\lambda_r^*} \right| \quad \text{and} \quad \Delta_{\omega_r} := \left| \frac{\omega_r - \omega_r^*}{\omega_r^*} \right|, \quad (3.18)$$

where  $\mathbf{u}_r$  could be any of  $\mathbf{a}_r, \mathbf{b}_r, \mathbf{c}_r, \mathbf{v}_r, \forall r \in [R]$ .

#### 3.7.1 Proof of Theorem 3.3.1

Theorem 3.3.1 provides the sufficient conditions which guarantee that the shared tensor components  $\mathbf{a}_r$  and non-shared components  $\mathbf{b}_r, \mathbf{c}_r$  recovered in Algorithm 1 converge to the truth  $\mathbf{a}_r^*$  and  $\mathbf{b}_r^*, \mathbf{c}_r^*$  respectively with the assumption that the tensor and matrix are dense and their decomposition weights are equal in each mode i.e  $\lambda_r^* = \omega_r^* \quad \forall r \in [R]$ . The theorem also provides the explicit convergence rates for the tensor components in Algorithm 1 and highlights the difference in rates between the shared and non-shared components.

Our proof consists of three steps. In Step 1 we use Lemma 1 to derive the close form for the optimization problem presented in equation (3.5). This step is only specific to the dense tensor and equal weights case as it makes it possible to derive a close form solution to the optimization formula presented in equation (3.5). In Step 2, we derive a general bound for the share and non-shared tensor estimates by proving Lemmas 2 and 3 given that the components obtained from the initialization method satisfy a specific error constraint. In Step 3, we simplify the error bound obtained in Lemma 2 and 3 to ensure that the share and non-shared tensor component estimate contract at a geometric rate in one iteration. Theorem 3.3.1 is then completed by showing that after enough iterations the contraction error vanishes to only leave a statistical error.

**Step 1:** The next lemma accomplishes the first step in proving Theorem 3.3.1. Since the tensor and matrix weights are assumed to be equal, without loss of generality we use  $\lambda_r^*$  and  $\lambda_r \forall r \in [R]$  to represent true and estimated weights respectively for both tensor and matrix.

### 3.7.2 Key Lemmas

**Lemma 1.** *Let  $res_M = \mathbf{M} - \sum_{m \neq r} \lambda_m \mathbf{a}_m \otimes \mathbf{v}_m$  and  $res_T = P_\Omega(\mathcal{T}) - P_\Omega(\sum_{m \neq r} \lambda_m \mathbf{a}_m \otimes \mathbf{b}_m \otimes \mathbf{c}_m)$  be the residual matrix and residual tensor, respectively defined on line (7) of Algorithm 1. In each ALS update of Algorithm 1, the solution to the optimization problem in equation (3.5) for the shared and non-shared components of the tensor and matrix in the  $r^{th}$  iteration of the inner loop are,*

$$\textbf{Share Components: } \mathbf{a}_r = \frac{\lambda_r res_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r) + \omega_r res_M \mathbf{v}_r}{\lambda_r^2 P_\Omega(\mathbf{I}, \mathbf{b}_r^2, \mathbf{c}_r^2) + \omega_r^2}, \quad (3.19)$$

$$\textbf{Tensor non-shared components: } \mathbf{b}_r = \tilde{\mathbf{b}}_r / \|\tilde{\mathbf{b}}_r\|_2, \quad \mathbf{c}_r = \tilde{\mathbf{c}}_r / \|\tilde{\mathbf{c}}_r\|_2, \quad \lambda_r = \|\tilde{\mathbf{c}}_r\|_2, \quad (3.20)$$

$$\textbf{Matrix non-shared components: } \mathbf{v}_r = \tilde{\mathbf{v}}_r / \|\tilde{\mathbf{v}}_r\|_2 \quad \text{and} \quad \omega_r = \|\tilde{\mathbf{v}}_r\|_2, \quad (3.21)$$

where  $\tilde{\mathbf{b}}_r, \tilde{\mathbf{c}}_r, \tilde{\mathbf{v}}_r$  have the following form

$$\tilde{\mathbf{b}}_r = \frac{res_T(\mathbf{a}_r, \mathbf{I}, \mathbf{c}_r)}{P_\Omega(\mathbf{a}_r^2, \mathbf{I}, \mathbf{c}_r^2)} \quad \tilde{\mathbf{c}}_r = \frac{res_T(\mathbf{a}_r, \mathbf{b}_r, \mathbf{I})}{P_\Omega(\mathbf{a}_r^2, \mathbf{b}_r^2, \mathbf{I})} \quad \text{and} \quad \tilde{\mathbf{v}}_r = res_T \mathbf{a}_r. \quad (3.22)$$

Note that the horizontal double lines in the expressions above indicate element-wise fraction and the squares in the denominator represent the element-wise squaring. The proof of Lemma 1 is provided in Section 3.8. It involves deriving the close form of the optimization problem presented in equation (3.5) in the non-sparse tensor case.

**Step 2:** The second step builds the error contraction results in one iteration of Algorithm 1. We achieve step two through Lemmas 2 and 3 which address the non-shared and shared component cases respectively.

**Lemma 2.** Assume Assumption 1 holds and  $p \geq \frac{C\mu^3(1+\gamma/3)\log_2(n^{10})}{n^{3/2}\gamma^2}$  for some positive  $\gamma$ . Also assume estimates  $\mathbf{a}_r$ ,  $\mathbf{b}_r$ ,  $\lambda_r$  of our algorithm with  $s_i = n_i$ ,  $i = 1, 2, 3, v$ , satisfy  $\max\{\|\mathbf{d}_{a_r}\|, \|\mathbf{d}_{b_r}\|, \lambda_r^* \Delta_{\lambda_r}\} \leq \epsilon_T \ \forall r \in [R]$  with  $\mathbf{d}_{a_r}, \mathbf{d}_{b_r}, \Delta_{\lambda_r}$  defined in (3.17). Then, the update for the non-shared tensor component  $\mathbf{c}_r$  satisfies with probability  $1 - 2n^{-9}$ ,

$$\max_{r \in [R]} \|\mathbf{c}_r - \mathbf{c}_r^*\|_2 \leq \frac{16R\lambda_{max}^* \max(c_0/\sqrt{n} + 3\epsilon_T, \gamma) \epsilon_T + (1 + \gamma)\|\mathcal{E}\|}{\lambda_{min}^*(1 - \gamma)}. \quad (3.23)$$

The detailed proof of Lemma 2 is presented in Section 3.8. We later show in step 3 of the proof of Theorem 3.3.1 that the upper bound in (3.23) can be written as the sum of a contracting term and a non contracting statistical error term.

**Lemma 3.** Assume Assumption 1 holds and  $p \geq \frac{C\mu^3(1+\gamma/3)\log_2(n^{10})}{n^{3/2}\gamma^2}$  for some positive  $\gamma$ . In addition, assume estimators  $\mathbf{c}_r$ ,  $\mathbf{b}_r$ ,  $\mathbf{v}_r$ ,  $\lambda_r$ ,  $\omega_r$  of our algorithm with  $s_i = n_i$ ,  $i = 1, 2, 3, v$ , satisfy  $\max\{\|\mathbf{d}_{c_r}\|, \|\mathbf{d}_{b_r}\|, \lambda_r^* \Delta_{\lambda_r}\} \leq \epsilon_T$  and  $\{\|\mathbf{d}_{v_r}\|, \omega_r^* \Delta_{\omega_r}\} \leq \epsilon_M \ \forall r \in [R]$ . Then the update for the shared tensor component  $\mathbf{a}_r$  satisfies with probability  $1 - 2n^{-9}$ ,

$$\max_{r \in [R]} \|\mathbf{a}_r - \mathbf{a}_r^*\|_2 \leq g(p, \epsilon_T, \zeta, R)\epsilon_T + f(\epsilon_M, \zeta, R)\epsilon_M + \frac{1}{\lambda_{min}^*} \frac{p(1 + \gamma)\|\mathcal{E}_T\| + \|\mathcal{E}_M\|}{p(1 - \gamma) + 1} \quad (3.24)$$

with  $g(p, \epsilon_T, \zeta, R) := \frac{16pR\lambda_{max}^*(\zeta+3\epsilon_T, \gamma)}{\lambda_{min}^*(p(1-\gamma)+1)}$ ;  $f(\epsilon_M, \zeta, R) := \frac{6R\lambda_{max}^*(\zeta+3\epsilon_M)\epsilon_M}{\lambda_{min}^*(p(1-\gamma)+1)}$ , and  $\zeta = c_0/\sqrt{n}$ .

The proof of Lemma 2 and Lemma 3 show that each iteration of Algorithm 1 results in an error contraction for the estimates of the non-shared ( $\mathbf{b}_r$  and  $\mathbf{c}_r$ ) and shared ( $\mathbf{a}_r$ ) tensor components respectively. Such results imply that after a sufficient number of iterations, Algorithm 1 can yield good estimates for these components. The detailed proof of Lemma 3

is discussed in Section 3.8.

**Step 3:** To complete the proof of the theorem, we carefully employ the assumptions on the initialization in order to guarantee that expressions (3.23) and (3.24) in Lemmas 2 and 3 can be written in the form  $\epsilon_R + q\epsilon_0$  with  $q \leq \frac{1}{2}$ . This entails showing that for  $f(\epsilon_M, \zeta, R)$  and  $g(\epsilon_M, \zeta, R)$  in the Lemma 3 adds up to less than  $\frac{1}{2}$  given the assumptions in Theorem 3.3.1. Denote  $\epsilon_0 := \max\{\epsilon_{T_0}, \epsilon_{M_0}\}$ ,  $q_1 := \frac{16R\lambda_{max}^*(\zeta+3\epsilon_0)(p+\frac{6}{16})}{\lambda_{min}^*(p(1-\gamma)+1)}$  and  $q_2 := \frac{16R\lambda_{max}^*(p\gamma+\frac{6}{16})(\zeta+3\epsilon_0)}{\lambda_{min}^*(p(1-\gamma)+1)}$ . Set  $\gamma := \frac{\lambda_{min}^*}{64R\lambda_{max}^*}$ . According to Assumption 3, we get that  $q_1 \leq \frac{p+6/16}{2p+2} \leq \frac{1}{2}$ . Also  $q_2 \leq \frac{p}{4(\frac{63}{64}p+1)} + \frac{3}{16} \leq \frac{1}{4} + \frac{3}{16} < \frac{1}{2}$  since  $p \leq 1$ . This implies that  $q := \max\{q_1, q_2\} \leq 1/2$ . Finally, we bound the error term of  $\max_{r \in [R]} \|\mathbf{a}_r - \mathbf{a}_r^*\|_2$  by showing that it can be written as a sum of a contracting term and a constant non-contracting term. Specifically, according to (3.24) in each iteration we have,

$$\begin{aligned} \max_{r \in [R]} \|\mathbf{a}_r - \mathbf{a}_r^*\|_2 &\leq g(p, \epsilon_{T_0}, \zeta, R)\epsilon_{T_0} + f(\epsilon_{M_0}, \zeta, R)\epsilon_{M_0} + \frac{1}{\lambda_{min}^*} \frac{p(1+\gamma)\|\mathcal{E}_T\| + \|\mathcal{E}_M\|}{p(1-\gamma)+1} \\ &\leq \max\{q_1, q_2\}\epsilon_0 + \frac{1}{\lambda_{min}^*} \frac{p(65/64)\|\mathcal{E}_T\| + \|\mathcal{E}_M\|}{p(63/64)+1} \\ &\leq q\epsilon_0 + \frac{1}{\lambda_{min}^*} \frac{p(65/64)\|\mathcal{E}_T\| + \|\mathcal{E}_M\|}{p(63/64)+1}, \end{aligned} \quad (3.25)$$

where  $q\epsilon_0$  is a contracting term and the term after it is non contracting. By iteratively applying the above inequality, after  $\tau = \Omega\left(\log_2\left(\frac{(p(63/64)+1)\epsilon_0}{(65/64)p\|\mathcal{E}_T\|+\|\mathcal{E}_M\|}\right)\right)$ , we get

$$\max_{r \in [R]} \|\mathbf{a}_r - \mathbf{a}_r^*\|_2 \leq \mathcal{O}\left(\frac{1}{\lambda_{min}^*} \frac{(65/64)p\|\mathcal{E}_T\| + \|\mathcal{E}_M\|}{p(63/64)+1}\right).$$

Similar derivation can be applied on the upper bound of  $\max_{r \in [R]} \|\mathbf{c}_r - \mathbf{c}_r^*\|_2$  in (3.23) to get a contracting and non contracting term. Then taking the maximum over all non-shared components and tensor weights lead to getting after running  $\tau = \Omega\left(\log_2\left(\frac{(63/64)\lambda_{min}^*\epsilon_0}{(65/64)\|\mathcal{E}_T\|}\right)\right)$  iterations of Algorithm 1,

$$\max_{r \in [R]} \left( \|\mathbf{b}_r - \mathbf{b}_r^*\|_2, \|\mathbf{c}_r - \mathbf{c}_r^*\|_2, \frac{|\lambda_r - \lambda_r^*|}{\lambda_r^*} \right) \leq \mathcal{O}\left(\frac{(65/64)\|\mathcal{E}_T\|}{(63/64)\lambda_{min}^*}\right),$$



which completes the proof of Theorem 3.3.1.  $\square$

### 3.7.3 Proof of Theorem 3.4.1

In this section we establish the results for the analysis of Theorem 3.4.1 which is the general and sparse case where the matrix and tensor weights are not assumed to be equal. In order to prove the general case we make use of some of the intermediate results derived in the analysis of Theorem 3.3.1. Namely, we follow the 3 three steps analysis approach introduced in the analysis of Theorem 3.3.1 and highlight the key difference which makes the analysis of Theorem 3.4.1 non trivial in comparison. As presented in the formulation of the optimization problem in (3.5) we use the  $\ell^0$  norm regularization as a mean to introduce sparsity in the model. However, deriving a close form solution to this sparse optimization problem becomes very difficult with this choice of regularization function. In step 1 of the analysis, we circumvent this issue by using a greedy truncation method defined on lines (9) and (11) of Algorithm 1 to approximate the sparse solution to the optimization problem in (3.5). We show that using the truncation method to only preserve the  $s$  largest entries of the components with the condition that  $s \geq d$  is suitable for accurate components recovery. In practice for Algorithm 1 the parameter  $s$  can be tuned in a data-driven manner following the sequential tuning schema presented in Algorithm 3.1.4. In step 2 of the analysis, we derive a general bound for the shared tensor component through Lemma 4. In step 3 we simplify the general bound derived in step 2 to show that one iteration of the algorithm results in a geometric error contraction. Theorem 3.4.1 is then completed by showing that after enough iterations the contraction error vanished to only leave a statistical error.

**Lemma 4.** *Assume Assumptions 4, 5 and 6 hold, and that  $s \geq d$ . In addition, assume estimators  $\mathbf{c}_r$ ,  $\mathbf{b}_r$ ,  $\mathbf{v}_r$ ,  $\lambda_r$ ,  $\omega_r$  of our algorithm satisfy  $\max\{\|\mathbf{d}_{c_r}\|, \|\mathbf{d}_{b_r}\|, \lambda_r^* \Delta_{\lambda_r}\} \leq \epsilon_T$  and*

$\{\|\mathbf{d}_{v_r}\|, \omega_r^* \Delta_{\omega_r}\} \leq \epsilon_M \forall r \in [R]$  and  $s_i \geq d_i$  for  $i = 1, 2, 3, v$ . Then the update for the shared tensor component  $\mathbf{a}_r$  satisfies with probability  $1 - 2n^{-9}$ ,

$$\begin{aligned} \max_{r \in [R]} \|\mathbf{a}_r - \mathbf{a}_r^*\|_2 &\leq g(p, \epsilon_T, \zeta, R) \epsilon_T + f(\epsilon_M, \zeta, R) \epsilon_M \\ &\quad + \frac{\lambda_{max}^* p(1 + \gamma) \|\mathcal{E}_T\|_{<d+s>} + \omega_{max}^* \|\mathcal{E}_M\|_{<d+s>}}{\lambda_{min}^{*2} p(1 - \gamma) + \omega_{min}^{*2}}, \end{aligned} \quad (3.26)$$

with  $g(p, \epsilon_T, \zeta, R) \leq \frac{24pR\lambda_{max}^{*2} \max(\zeta + 3\epsilon_T, \gamma)}{\lambda_{min}^{*2} p(1 - \gamma) + \omega_{min}^{*2}}$ ;  $f(\epsilon_M, \zeta, R) \leq \frac{9R\omega_{max}^{*2}(\zeta + 3\epsilon_M)}{\lambda_{min}^{*2} p(1 - \gamma) + \omega_{min}^{*2}}$  and where  $\zeta = c_0/\sqrt{d}$ .

The detailed proof of Lemma 4 is discussed in Section 3.8.

**Step 3:** The last step in the proof of Theorem 3.4.1, consists in using the assumptions on the initialization error in order to guarantee that expression (3.26) in Lemmas 4 can be written in the form  $\epsilon_R + q\epsilon_0$  with  $q \leq \frac{1}{2}$ . Just like was the case in the proof of Theorem 3.3.1, this entails showing that for  $f(\epsilon_M, \zeta, R)$  and  $g(\epsilon_M, \zeta, R)$  adds up to less than  $\frac{1}{2}$  given the assumptions in Theorem 3.4.1.

Given the initialization condition in Assumption 6 we get

$$g(p, \epsilon_T, \zeta, R) \leq \frac{24pR\lambda_{max}^{*2} \max(\zeta + 3\epsilon_T, \gamma)}{\lambda_{min}^{*2} p(1 - \gamma) + \omega_{min}^{*2}}; \quad f(\epsilon_M, \zeta, R) \leq \frac{9R\omega_{max}^{*2}(\zeta + 3\epsilon_M)}{\lambda_{min}^{*2} p(1 - \gamma) + \omega_{min}^{*2}}$$

Denote  $\epsilon_0 := \max\{\epsilon_{T_0}, \epsilon_{M_0}\}$ ,  $q_1 := \frac{24R(\zeta + 3\epsilon_0)(\lambda_{max}^{*2} p + \frac{9}{24}\omega_{max}^{*2})}{\lambda_{min}^{*2} p(1 - \gamma) + \omega_{min}^{*2}}$  and  $q_2 := \frac{24R(\lambda_{max}^{*2} p\gamma + \frac{9}{24}\omega_{max}^{*2}(\zeta + 3\epsilon_0))}{\lambda_{min}^{*2} p(1 - \gamma) + \omega_{min}^{*2}}$ .

We choose  $\gamma = \frac{1/2\lambda_{min} + 1/2\omega_{min}}{96R\lambda_{max}}$ . According to Assumption 6 we get that  $q_1 \leq \frac{p\lambda_{max}^{*2} + 3/8\omega_{max}^{*2}}{2(p\lambda_{max}^{*2} + \omega_{max}^{*2})} \leq \frac{1}{2}$ .

Also  $q_2 \leq \frac{p \min\{\lambda_{min}^{*2}, \omega_{min}^{*2}\}}{4(\lambda_{min}^{*2} p \frac{95}{96} + \omega_{min}^{*2})} + \frac{3\omega_{max}^{*2}}{16(p\lambda_{max}^{*2} + \omega_{max}^{*2})} \leq \frac{p}{4(p \frac{95}{96} + 1)} + \frac{3}{16}$ . Hence  $q_2 \leq \frac{1}{4} + \frac{3}{16} < \frac{1}{2}$  since  $p \leq 1$ .

This implies that  $q := \max\{q_1, q_2\} \leq 1/2$ .

Finally, we bound the error term of  $\max_{r \in [R]} \|\mathbf{a}_r - \mathbf{a}_r^*\|_2$  by showing that it can be written as a

sum of a contracting term and a constant non-contracting term. Specifically, according to (3.24) in each iteration we have,

$$\begin{aligned}
\max_{r \in [R]} \|\mathbf{a}_r - \mathbf{a}_r^*\|_2 &\leq g(p, \epsilon_{T_0}, \zeta, R) \epsilon_{T_0} + f(\epsilon_{M_0}, \zeta, R) \epsilon_{M_0} \\
&\quad + \frac{(\lambda_{max}^* + \epsilon_T) p (1 + \gamma) \|\mathcal{E}_T\|_{<d+s>} + (\omega_{max}^* + \epsilon_T) \|\mathcal{E}_M\|_{<d+s>}}{(\lambda_{min}^* + \epsilon_T)^2 p (1 - \gamma) + (\omega_{min}^* + \epsilon_M)^2} \\
&\leq \max\{q_1, q_2\} \epsilon_0 + \frac{(97/96) p \lambda_{max}^* \|\mathcal{E}_T\|_{<d+s>} + \omega_{max}^* \|\mathcal{E}_M\|_{<d+s>}}{\frac{95}{96} p \lambda_{min}^{*2} + \omega_{min}^{*2}} \\
&\leq q \epsilon_0 + \frac{(97/96) p \lambda_{max}^* \|\mathcal{E}_T\|_{<d+s>} + \omega_{max}^* \|\mathcal{E}_M\|_{<d+s>}}{\frac{95}{96} p \lambda_{min}^{*2} + \omega_{min}^{*2}}, \tag{3.27}
\end{aligned}$$

where  $q \epsilon_0$  is a contracting term. By iteratively applying the above inequality, after the number of iterations stated in Theorem 3.4.1, we get

$$\max_{r \in [R]} \|\mathbf{a}_r - \mathbf{a}_r^*\|_2 \leq \mathcal{O} \left( \frac{(97/96) p \lambda_{max}^* \|\mathcal{E}_T\|_{<d+s>} + \omega_{max}^* \|\mathcal{E}_M\|_{<d+s>}}{(95/96) p \lambda_{min}^{*2} + \omega_{min}^{*2}} \right),$$

The proof for the non-shared component in Theorem 3.4.1 is very similar to that of the non-share component in Theorem 3.3.1 we therefore leave it out. This completes the proof of Theorem 3.4.1.  $\square$

### 3.8 Additional Results

In this section we provide details of the derivation for the proofs of Lemmas 1-4.

#### 3.8.1 Proof of Lemma 1

The dense version of the optimization problem in (3.5) can be formulated as follows:

**Optimization:** Non-Sparse formulation

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{V}} \left\{ \|P_\Omega(\mathcal{T}) - P_\Omega\left(\sum_{r \in [R]} \lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r\right)\|_F^2 + \|\mathbf{M} - \sum_{r \in [R]} \omega_r \mathbf{a}_r \otimes \mathbf{v}_r\|_F^2 \right\} \tag{3.28}$$

subject to  $\omega_r, \lambda_r \in \mathbb{R}^+$ ,

Given  $\text{res}_M = \mathbf{M} - \sum_{m \neq r} \omega_m \mathbf{a}_m \otimes \mathbf{v}_m$  and  $\text{res}_T = P_\Omega(\mathcal{T}) - P_\Omega(\sum_{m \neq r} \lambda_r \mathbf{a}_m \otimes \mathbf{b}_m \otimes \mathbf{c}_m)$  the residual matrix and residual tensor, respectively. In each ALS update of Algorithm 1 we need to solve the following least-squares optimizations problem.

$$\min_{\mathbf{a}_r} \left\{ \|\text{res}_M - \omega_r \mathbf{a}_r \otimes \mathbf{v}_r\|_F^2 + \|\text{res}_T - P_\Omega(\lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r)\|_F^2 \right\}. \quad (3.29)$$

The optimization problem in (3.29) is convex in  $\mathbf{a}_r$ . Therefore, we can find  $\mathbf{a}_r$  by taking its derivative and setting it to zero. In order to do this we first derive the equivalent of the optimization function in (3.29) explicitly in terms of the entries of the tensor and matrix components:

$$\min_{\mathbf{a}_r} \left\{ \sum_{i,j} \left( \text{res}_{M_{i,l}} - \omega_r \mathbf{a}_r(i) \times \mathbf{v}_r(l) \right)^2 + \sum_{\{i,j,k\} \in \Omega} \left( \text{res}_{T_{i,j,k}} - \lambda_r \mathbf{a}_r(i) \times \mathbf{b}_r(j) \times \mathbf{c}_r(k) \right)^2 \right\}, \quad (3.30)$$

where  $\text{res}_{T_{i,j,k}}$  is the  $(i, j, k)^{\text{th}}$  entry of  $\text{res}_T$  and  $\text{res}_{M_{i,l}}$  is the  $(i, l)^{\text{th}}$  entry of  $\text{res}_M$ . The notation  $\{i, j, k\} \in \Omega$  with  $\Omega$  defines in (3.1), guarantees that the summation only applies on the observed entries of tensor  $\text{res}_T$ ;  $\mathbf{a}_r(i)$  is the  $i^{\text{th}}$  component of  $\mathbf{a}_r$  where  $i \in [n]$ .

Taking the derivative of (3.30) with respect to  $\mathbf{a}_r(i)$  for all  $i \in [n]$  and setting it to zero we get:

$$\mathbf{a}_r(i) = \frac{\lambda_r \sum_{j,k} (\text{res}_{T_{i,j,k}} \mathbf{b}_r(j) \mathbf{c}_r(k)) + \omega_r \sum_j \text{res}_{M_{i,l}} \mathbf{v}_r(l)}{\lambda_r^2 \sum_{j,k} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k) + \omega_r^2 \sum_l \mathbf{v}_r^2(l)} \quad (3.31)$$

for all  $i \in [n]$ . The first summation in the numerator of equation (3.31) is the definition of the modes 2 and 3 tensor matrix product of  $\text{res}_T$  with the matrix obtained from  $\mathbf{b}_r \otimes \mathbf{c}_r$ . Following the notation provided in section 2.2 this product can be rewritten as:

$$\text{res}_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r) = \text{res}_T \times_2 \mathbf{b}_r \times_3 \mathbf{c}_r, \quad (3.32)$$

for all  $i \in [n]$ , where  $\mathbf{I}$  is the identity matrix. It is worth noting that the vector tensor product in (3.32) is a vector of length  $n$ . We can write the second term in the numerator as a matrix vector left multiplication. The vector  $\mathbf{a}_r$  can therefore be written as:

$$\mathbf{a}_r = \frac{\lambda_r \text{res}_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r) + \omega_r \text{res}_M \mathbf{v}_r}{\lambda_r^2 P_\Omega(\mathbf{I}, \mathbf{b}_r^2, \mathbf{c}_r^2) + \omega_r^2}, \quad (3.33)$$

where the double line fraction indicates element-wise division and  $(\cdot)^2$  denotes elements-wise power.

In order to solve the optimization problem for components other than the first component that are not shared with the matrix we proceed similarly. We start from:

$$\min_{\mathbf{b}_r} \left\{ \|\text{res}_T - P_\Omega(\lambda_r \mathbf{a}_r \otimes \mathbf{b}_r)\|_F^2 \right\}, \quad (3.34)$$

which is equivalent to

$$\sum_{\{i,j,k\} \in \Omega} \left( \text{res}_{T_{i,j,k}} - \lambda_r \mathbf{a}_r(i) \times \mathbf{b}_r(j) \times \mathbf{c}_r(k) \right)^2. \quad (3.35)$$

Taking the derivative of (3.35) with respect to  $\mathbf{b}_r(j)$  or  $\mathbf{c}_r(k)$  then setting to them to zero and solving for  $\mathbf{b}_r(j)$  or  $\mathbf{c}_r(k)$  we get the following update:

$$\tilde{\mathbf{b}}_r(j) := \lambda_r \mathbf{b}_r(j) = \frac{\sum_{\{i,.,k\} \in \Omega} (\text{res}_{T_{i,j,k}} \mathbf{a}_r(i) \mathbf{c}_r(k))}{\sum_{\{i,.,k\} \in \Omega} \mathbf{a}_r^2(i) \mathbf{c}_r^2(k)}, \quad \tilde{\mathbf{c}}_r(k) := \lambda_r \mathbf{c}_r(k) = \frac{\sum_{\{i,j,.\} \in \Omega} (\text{res}_{T_{i,j,k}} \mathbf{a}_r(i) \mathbf{b}_r(j))}{\sum_{\{i,j,.\} \in \Omega} \mathbf{a}_r^2(i) \mathbf{b}_r^2(j)}, \quad (3.36)$$

respectively. In vector form this is written as,

$$\tilde{\mathbf{b}}_r = \frac{\text{res}_T(\mathbf{a}_r, \mathbf{I}, \mathbf{c}_r)}{P_\Omega(\mathbf{a}_r^2, \mathbf{I}, \mathbf{c}_r^2)} \quad \text{and} \quad \tilde{\mathbf{c}}_r = \frac{\text{res}_T(\mathbf{a}_r, \mathbf{b}_r, \mathbf{I})}{P_\Omega(\mathbf{a}_r^2, \mathbf{b}_r^2, \mathbf{I})}. \quad (3.37)$$

These are the un-normalized updates in line 10 of Algorithm 1. Since by definition  $\mathbf{b}_r$  and  $\mathbf{c}_r$  are unit vectors then  $\|\tilde{\mathbf{c}}_r\|_2 = \|\lambda_r \mathbf{c}_r\|_2 = |\lambda_r|$  as defined in line 12 of Algorithm 1 and  $\mathbf{c}_r = \tilde{\mathbf{c}}_r / \|\tilde{\mathbf{c}}_r\|_2$  as in line 13 of the main algorithm. The update for  $\mathbf{b}$  is obtained in a similar manner. The above derivation corresponds to the non-sparse scenario, i.e., Algorithm 1

without the truncation steps on lines 9 and 11. However for the sparse case, to incorporate sparsity in the resulting update equations, we use the truncation scheme proposed in Sun, Lu, Liu, *et al.* [49]. We get the estimate of the matrix component  $\mathbf{v}_r$ , using a similar derivation and get,

$$\tilde{\mathbf{v}}_r := \omega_r \mathbf{v}_r = \text{res}_T \mathbf{a}_r, \quad (3.38)$$

and since  $\mathbf{v}_r$  is a unit vector we get  $\omega_r = \|\tilde{\mathbf{v}}_r\|_2$  and  $\mathbf{v}_r = \tilde{\mathbf{v}}_r / \|\tilde{\mathbf{v}}_r\|_2$  as in lines 12 and 13 of Algorithm 1. This complete the proof of Lemma 1.  $\square$

### 3.8.2 Proof of Lemma 2

The main challenge in the proof of Lemma 2 lies in finding a tight upper bound for the error of  $c_r$ . In the following derivation only provide the analysis for the non-shared tensor components  $\mathbf{c}_r$  since the proof of the other non-shared component  $\mathbf{b}_r$  is very similar.

In (3.20) we derived the close form formula for the update  $\mathbf{c}_r$  to be  $\tilde{\mathbf{c}}_r / \|\tilde{\mathbf{c}}_r\|_2$ . To bound the expression  $\|\mathbf{c}_r - \mathbf{c}_r^*\|_2$ , we make use of the intermediate estimate  $\tilde{\mathbf{c}}_r$  which is define in (3.22) as,

$$\tilde{\mathbf{c}}_r = \frac{\text{res}_T(\mathbf{a}_r, \mathbf{b}_r, \mathbf{I})}{P_\Omega(\mathbf{a}_r^2, \mathbf{b}_r^2, \mathbf{I})}. \quad (3.39)$$

From Lemma 1, notice that  $\tilde{\mathbf{c}}_r$  can be written as  $\lambda_r \mathbf{c}_r$ . That is,  $\tilde{\mathbf{c}}_r$  can be thought of as the un-normalized version of the estimate  $\mathbf{c}_r$ . Proving Lemma 2 therefore consists in deriving an error bound for  $\|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2$ , followed by using Lemma 10 which shows that  $\|\mathbf{c}_r - \mathbf{c}_r^*\|_2 \leq \frac{2}{\lambda_r^*} \|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2$ .

Let  $\mathbf{D}$ ,  $\mathbf{E}$ ,  $\mathbf{F}$ ,  $\mathbf{G}$ , be  $n \times n$  diagonal matrices with the following diagonal elements,

$$\begin{aligned} \mathbf{D}_{kk} &= \sum_{i,j} \delta_{ijk} \mathbf{a}_r^2(i) \mathbf{b}_r^2(j) ; \quad \mathbf{E}_{kk} = \sum_{i,j} \delta_{ijk} \mathbf{a}_r^*(i) \mathbf{b}_r^*(j) \mathbf{a}_r(i) \mathbf{b}_r(j); \\ \mathbf{F}_{kk} &= \sum_{i,j} \delta_{ijk} \mathbf{a}_m^*(i) \mathbf{b}_m^*(j) \mathbf{a}_r(i) \mathbf{b}_r(j) ; \quad \mathbf{G}_{kk} = \sum_{i,j} \delta_{ijk} \mathbf{a}_m(i) \mathbf{b}_m(j) \mathbf{a}_r(i) \mathbf{b}_r(j), \end{aligned}$$

where  $\delta_{ijk}$  is a Bernoulli random variable with success probability  $p$  and indicates whether the  $ijk_{th}$  tensor entry is observed or not. Then the vector  $\tilde{\mathbf{c}}_r$  obtained after one pass of the inner loop of Algorithm 1 can be written as

$$\tilde{\mathbf{c}}_r = \mathbf{D}^{-1} \left( \lambda_r^* \mathbf{E} \mathbf{c}_r^* + \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \mathbf{c}_m^* - \lambda_m \mathbf{G} \mathbf{c}_m) + \mathcal{E}_T(\mathbf{a}_r, \mathbf{b}_r, \mathbf{I}) \right). \quad (3.40)$$

We make use of the fact that  $\|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2 = \|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{D}^{-1} \mathbf{D} \mathbf{c}_r^*\|_2$ , to yield,

$$\|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2 = \left\| \underbrace{\lambda_r^* \mathbf{D}^{-1} (\mathbf{E} - \mathbf{D}) \mathbf{c}_r^*}_{err_1} + \underbrace{\mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \mathbf{c}_m^* - \lambda_m \mathbf{G} \mathbf{c}_m)}_{err_2} + \underbrace{\mathbf{D}^{-1} \mathcal{E}_T(\mathbf{a}_r, \mathbf{b}_r, \mathbf{I})}_{err_3} \right\|_2$$

Applying the triangle inequality to the above expression is very convenient as it breaks its into the three different error terms shown below, each characterizing different sources of error affecting the non-shared component update,

$$\|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2 \leq \|err_1\|_2 + \|err_2\|_2 + \|err_3\|_2, \quad (3.41)$$

where  $err_1 = \lambda_r^* \mathbf{D}^{-1} (\mathbf{E} - \mathbf{D}) \mathbf{c}_r^*$  can be characterized as the error due to the power method. This error is well understood and does not require meticulous bound control in order to yield the desire result. Also if  $\mathcal{T}^*$  was a rank 1 and noiseless tensor, the proof of Lemma 2 would reduce to bounding this error term.

Unlike  $err_1$  discussed above, bounding  $err_2 = \mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \mathbf{c}_m^* - \lambda_m \mathbf{G} \mathbf{c}_m)$  represents the main challenge in the proof. It is worth noting that  $err_2$  is the error due to the deflation method applied in Algorithm 1. Two issues arises with bounding this error, the first resides in the non-orthogonality of the tensor  $\mathcal{T}^*$ . If the tensor  $\mathcal{T}^*$  was orthogonal then a deflation algorithm would have little to no difficulty differentiating between the ranks of the tensor. However with the non-orthogonality assumption we are left with a non disappearing residual due to fact that for example two component vectors of the tensor  $\mathbf{c}_r$  and  $\mathbf{c}_j$  could be close to parallel making it difficult for the algorithm to differentiate between the two. Moreover  $err_2$  exposes the relationship that exists between recovering a component  $\mathbf{c}_r$  and the error

for the other mode components  $\mathbf{a}_j, \mathbf{b}_j$  and with  $j \neq r$ . If not carefully controlled,  $err_2$  could cause the estimate  $\mathbf{c}_r$  to diverge from  $\mathbf{c}_r^*$ . Assumption (1.iii) is therefore used and required to control the magnitude of  $err_2$ .

The third error term  $err_3 = \mathbf{D}^{-1}\mathcal{E}_T(\mathbf{a}_r, \mathbf{b}_r, \mathbf{I})$  is simply the error due to the noise of the tensor and can be easily bounded after standard assumptions are made about the spectral norm of  $\mathcal{E}_T$ . Another challenge in bounding the error of the  $\mathbf{c}_r$  update comes from the fact that the tensor has missing entries. As represented in equation (3.39) the operations involved in computing the update  $\mathbf{c}_r$  is only carried on the observed entries of the tensor. This computation caveat forces the use of concentration inequalities in the analysis of the error bound of the component. Choosing the right concentration inequality becomes therefore very important in order to guarantee a given convergence rate while allowing some reasonable constraints on the tensor entry reveal probability to  $p$ . The rest of the proof consists in finding a bound for each of the three errors discussed above. We start with bounding the first error term. Using the fact that  $\|\mathbf{c}_r^*\|_2 = 1$  and since  $\mathbf{D}^{-1}(\mathbf{E} - \mathbf{D})$  is a diagonal matrix its spectral norm is the maximum absolute value of its diagonal elements, we get

$$\begin{aligned} \|err_1\|_2 &\leq \|\lambda_r^* \mathbf{D}^{-1}(\mathbf{E} - \mathbf{D})\|_2 \\ &= \lambda_r^* \max_k |\mathbf{D}^{-1}(\mathbf{E} - \mathbf{D})|_{kk} \\ &\leq \lambda_r^* \max_k |\mathbf{D}^{-1}|_{kk} \max_k |(\mathbf{E} - \mathbf{D})|_{kk}. \end{aligned}$$

Next is finding an upper bound for the maximum of each of the random elements in the equation above with high probability. To do that we first get an upper bound for each of the diagonal elements with high probability and make use of the union bound method. This is derived as:

$$\begin{aligned} |(\mathbf{E} - \mathbf{D})_{kk}| &= \left| \sum_{ij} \delta_{ijk} \mathbf{a}_r^*(i) \mathbf{b}_r^*(j) \mathbf{a}_r(i) \mathbf{b}_r(j) - \sum_{jk} \delta_{ijk} \mathbf{a}_r^2(i) \mathbf{b}_r^2(j) \right| \\ &= \left| \sum_{ij} \delta_{ijk} \mathbf{a}_r^*(i) \mathbf{d}_{b_r}(j) \mathbf{a}_r(i) \mathbf{b}_r(j) - \sum_{ij} \delta_{ijk} \mathbf{d}_{a_r}(i) \mathbf{b}_r^*(j) \mathbf{a}_r(i) \mathbf{b}_r(j) \right. \\ &\quad \left. - \sum_{ij} \delta_{ijk} \mathbf{d}_{a_r}(i) \mathbf{d}_{b_r}(j) \mathbf{a}_r(i) \mathbf{b}_r(j) \right|. \end{aligned}$$



The expression on the right side of the equality are obtained from the fact that  $\mathbf{a}_r(i) = \mathbf{a}_r^*(i) + \mathbf{d}_{a_r}(i)$  and  $\mathbf{b}_r(j) = \mathbf{b}_r^*(j) + \mathbf{d}_{b_r}(j)$ . Next Lemma 7 is used to bound the three random elements inside the absolute value. Combined with the triangle inequality and the fact that  $|\langle \mathbf{d}_{a_r}, \mathbf{a}_r^* \rangle| = \frac{1}{2} \|\mathbf{d}_{a_r}\|_2^2$  (Lemma 12) yields the following,

$$\begin{aligned}
|(\mathbf{E} - \mathbf{D})_{kk}| &\leq p (|\langle \mathbf{a}_r^*, \mathbf{a}_r \rangle \langle \mathbf{d}_{b_r}, \mathbf{b}_r \rangle| + |\langle \mathbf{d}_{a_r}, \mathbf{a}_r \rangle \langle \mathbf{b}_r^*, \mathbf{b}_r \rangle| + |\langle \mathbf{d}_{a_r}, \mathbf{a}_r \rangle \langle \mathbf{d}_{b_r}, \mathbf{b}_r \rangle|) \\
&\quad + p\gamma (\|\mathbf{d}_{a_r}\|_2 + \|\mathbf{d}_{b_r}\|_2 + \|\mathbf{d}_{a_r}\|_2 \|\mathbf{d}_{b_r}\|_2) \\
&\leq 6p \left( \max_{\mathbf{u}_r \in \{a_r, b_r\}} \left\{ \sqrt{1 - \frac{\|d_{u_r}\|_2^2}{2}} \|\mathbf{d}_{u_r}\|_2^2, \|\mathbf{d}_{u_r}\|_2^4, \gamma \|\mathbf{d}_{u_r}\|_2 \right\} \right) \\
&= 6p \max_{\mathbf{u}_r \in \{a_r, b_r\}} \left( \sqrt{1 - \frac{\|d_{u_r}\|_2^2}{2}} \|\mathbf{d}_{u_r}\|_2, \|\mathbf{d}_{u_r}\|_2^3, \gamma \right) \|\mathbf{d}_{u_r}\|_2. \tag{3.42}
\end{aligned}$$

The above inequality holds with probability  $1 - 2n^{-10}$  provided the reveal probability  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(n^{10})}{n^{3/2}\gamma^2}$ . Using (3.42) and the bound from Lemma 6, we get

$$\|err_1\|_2 \leq \frac{6p\lambda_r^* \max_{\mathbf{u}_r \in \{a_r, b_r\}} \left( \sqrt{1 - \frac{\|d_{u_r}\|_2^2}{2}} \|\mathbf{d}_{u_r}\|_2, \|\mathbf{d}_{u_r}\|_2^3, \gamma \right) \|\mathbf{d}_{u_r}\|_2}{p(1 - \gamma)}, \tag{3.43}$$

with probability  $1 - 2n^{-9}$ .

Next we work on bounding  $err_2$ . Note that

$$\begin{aligned}
\|err_2\| &= \|\mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \mathbf{c}_m^* - \lambda_m \mathbf{G} \mathbf{c}_m)\|_2 \\
&\leq \max_{kk} |\mathbf{D}^{-1}|_{kk} \sum_{m \in [R] \setminus r} \|(\lambda_m^* \mathbf{F} \mathbf{c}_m^* - \lambda_m \mathbf{G} \mathbf{c}_m)\|_2 \\
&= \lambda_m^* \max_{kk} |\mathbf{D}^{-1}|_{kk} \sum_{m \in [R] \setminus r} \|\mathbf{F} \mathbf{c}_m^* - \mathbf{G} \mathbf{c}_m + \Delta_{\lambda_m} \mathbf{G} \mathbf{c}_m\|_2 \\
&\leq \lambda_m^* \max_{kk} |\mathbf{D}^{-1}|_{kk} \sum_{m \in [R] \setminus r} \|(\mathbf{F} - \mathbf{G}) \mathbf{c}_m^*\|_2 + \|\mathbf{G} \mathbf{d}_{c_m}\|_2 + \|\Delta_{\lambda_m} \mathbf{G} \mathbf{c}_m\|_2. \tag{3.44}
\end{aligned}$$

We focus on bounding each of the four components in the last inequality above as

$$\begin{aligned}\|\mathbf{F}\mathbf{c}_m^* - \mathbf{G}\mathbf{c}_m\|_2 &\leq \|(\mathbf{F} - \mathbf{G})\mathbf{c}_m^*\| + \|\mathbf{G}\mathbf{d}_{\mathbf{c}_m}\|_2 \\ &= \max_i |\mathbf{F}_{ii} - \mathbf{G}_{kk}| \|\mathbf{c}_m^*\|_2 + \max_i |\mathbf{G}_{kk}| \|\mathbf{d}_{\mathbf{c}_m}\|_2.\end{aligned}\quad (3.45)$$

Just like we did for  $err_1$  we bound each element  $|\mathbf{F}_{kk} - \mathbf{G}_{kk}|$  then apply the union bound to get the bound its maximum,

$$\begin{aligned}|\mathbf{F}_{kk} - \mathbf{G}_{kk}| &= \left| \sum_{jk} \delta_{ijk} \mathbf{a}_m^*(i) \mathbf{b}_m^*(j) \mathbf{a}_r(i) \mathbf{b}_r(j) - \sum_{jk} \delta_{ijk} \mathbf{a}_m(i) \mathbf{b}_m(j) \mathbf{a}_r(i) \mathbf{b}_r(j) \right| \\ &\leq \left| \sum_{jk} \delta_{ijk} \mathbf{d}_{a_m}(i) \mathbf{b}_m^*(j) \mathbf{a}_r(i) \mathbf{b}_r(j) \right| + \left| \sum_{jk} \delta_{ijk} \mathbf{a}_m^*(i) \mathbf{d}_{b_m}(j) \mathbf{a}_r(i) \mathbf{b}_r(j) \right| \\ &\quad + \left| \sum_{jk} \delta_{ijk} \mathbf{d}_{a_m}(i) \mathbf{d}_{b_m}(j) \mathbf{a}_r(i) \mathbf{b}_r(j) \right| \\ &\leq p (|\langle \mathbf{d}_{a_m}, \mathbf{a}_r \rangle \langle \mathbf{b}_m^*, \mathbf{b}_r \rangle| + |\langle \mathbf{a}_m^*, \mathbf{a}_r \rangle \langle \mathbf{d}_{b_m}, \mathbf{b}_r \rangle| + |\langle \mathbf{d}_{a_m}, \mathbf{a}_r \rangle \langle \mathbf{d}_{b_m}, \mathbf{b}_r \rangle|) \\ &\quad + \gamma (\|\mathbf{d}_{a_m}\|_2 + \|\mathbf{d}_{b_m}\|_2 + \|\mathbf{d}_{a_m}\|_2 \|\mathbf{d}_{b_m}\|_2) \\ &\leq 6p \max_{\mathbf{u} \in \{\mathbf{a}_m, \mathbf{b}_m, \mathbf{a}_r, \mathbf{b}_r\}} \left( \left( \frac{c_0}{\sqrt{(n)}} + \|\mathbf{d}_{\mathbf{u}}\|_2 \right) \|\mathbf{d}_{\mathbf{u}}\|_2, \gamma \|\mathbf{d}_{\mathbf{u}}\|_2 \right).\end{aligned}$$

The last inequality above holds with probability  $1 - 2n^{-10}$  provided the reveal probability  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(n^{10})}{n^{3/2}\gamma^2}$ . The second inequality is obtained by using Lemma 8 and the last inequality is obtained using the incoherence assumption (1.iii) to get that  $\max\{|\langle \mathbf{a}_m^*, \mathbf{a}_r \rangle|, |\langle \mathbf{b}_m^*, \mathbf{b}_r \rangle|\} \leq \frac{c_0}{\sqrt{(n)}} + \max\{\|\mathbf{d}_{\mathbf{a}_r}\|_2, \|\mathbf{d}_{\mathbf{b}_r}\|_2\}$ . Using the union bound we get that

$$\max_k |\mathbf{F}_{kk} - \mathbf{G}_{kk}| \leq 6p \max_{\mathbf{u} \in \{\mathbf{a}_m, \mathbf{b}_m, \mathbf{a}_r, \mathbf{b}_r\}} \left( \left( \frac{c_0}{\sqrt{(n)}} + \|\mathbf{d}_{\mathbf{u}}\|_2 \right), \gamma \right) \|\mathbf{d}_{\mathbf{u}}\|_2, \quad (3.46)$$

with probability  $1 - 2n^{-9}$ .

Similarly using Lemma 8, and applying the union bound and the fact that,

$$|\langle \mathbf{a}_m, \mathbf{a}_r \rangle \langle \mathbf{b}_m, \mathbf{b}_r \rangle| \leq \max\{\langle \mathbf{a}_m, \mathbf{a}_r \rangle^2, \langle \mathbf{b}_m, \mathbf{b}_r \rangle^2\} \quad (3.47)$$

$$\leq \left( \frac{c_0}{\sqrt{(n)}} + \max_{\mathbf{u}_r \in \{\mathbf{a}_r, \mathbf{b}_r\}} 3\|\mathbf{d}_{u_r}\|_2 \right)^2, \quad (3.48)$$

yields the following inequality,

$$\max_k |\mathbf{G}_{kk}| \leq p \max_{\mathbf{u}_r \in \{\mathbf{a}_r, \mathbf{b}_r, \mathbf{a}_m, \mathbf{b}_m\}} \left( \left( \frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_{u_r}\|_2 \right)^2, \gamma \right), \quad (3.49)$$

with probability  $1 - 2n^{-9}$ .

Putting equations (3.44), (3.45), (3.49) and using Lemma 6 to bound  $\mathbf{D}^{-1}$  yields,

$$\|err_2\|_2 \leq \frac{8p \sum_{m \in [R] \setminus r} \lambda_m^* \max_{\mathbf{u} \in \{\mathbf{a}_m, \mathbf{b}_m, \mathbf{a}_r, \mathbf{b}_r\}} \left( \left( \frac{c_0}{\sqrt{(n)}} + \|\mathbf{d}_{\mathbf{u}}\|_2 \right), \left( \frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_{\mathbf{u}}\|_2 \right)^2, \gamma \right) \|\mathbf{d}_{\mathbf{u}}\|_2}{p(1 - \gamma)}, \quad (3.50)$$

with probability  $1 - 2n^{-9}$  provided  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(n^{10})}{n^{3/2}\gamma^2}$ .

Next we use Lemma 11, combined with Lemma 6 and apply the union bound to get the bound on the tensor noise  $\|err_3\|_2$  as

$$\|err_3\|_2 \leq \frac{p(1 + \gamma)\|\mathcal{E}_T\|}{p(1 - \gamma)}, \quad (3.51)$$

with probability  $1 - 2n^{-9}$  provided  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(n^{10})}{n^{3/2}\gamma^2}$ . Combining the error bounds results of  $\|err_1\|_2$ ,  $\|err_2\|_2$ ,  $\|err_3\|_2$  in equations (3.43), (3.50) and (3.51) respectively, yields

$$\begin{aligned} & \|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2 \\ & \leq \frac{8pR\lambda_{max}^* \max_{\mathbf{u} \in \{\mathbf{a}_m, \mathbf{b}_m, \mathbf{a}_r, \mathbf{b}_r\}} \left( \sqrt{1 - \frac{\|\mathbf{d}_{\mathbf{u}}\|_2^2}{2}} \|\mathbf{d}_{\mathbf{u}}\|_2, \left( \frac{c_0}{\sqrt{(n)}} + \|\mathbf{d}_{\mathbf{u}}\|_2 \right), \left( \frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_{\mathbf{u}}\|_2 \right)^2, \|\mathbf{d}_{\mathbf{u}}\|_2^3, \gamma \right) \|\mathbf{d}_{\mathbf{u}}\|_2}{p(1 - \gamma)} \\ & + \frac{p(1 + \gamma)\|\mathcal{E}\|}{p(1 - \gamma)}, \end{aligned} \quad (3.52)$$

with probability  $1 - 2n^{-9}$ . The proof of Lemma 2 is then completed by applying the results of Lemma 10 which shows that  $\|\mathbf{c}_r - \mathbf{c}_r^*\|_2 \leq \frac{2}{\lambda_r^*} \|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2$  and Lemma 9 ( $|\lambda_r - \lambda_r^*| \leq \|\tilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2$ ) and by letting  $\max\{\|\mathbf{d}_u\|_2\} = \epsilon_T$ .  $\square$

### 3.8.3 Proof of Lemma 3

We now prove the contraction result in one iteration of Algorithm 1 for the shared components of the tensor and matrix  $\mathbf{a}_r$  in the special case where the tensor and matrix weights are equal and both tensor and matrix are dense. When the tensor and matrix weight are assumed to be equal, the close form solution for the update of the shared tensor component derived in Lemma 1 simplifies to  $\mathbf{a}_r = \frac{(\text{res}_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r) + \text{res}_M \mathbf{v}_r)}{\lambda_r(P_\Omega(\mathbf{I}, (\mathbf{b}_r)^2, (\mathbf{c}_r)^2) + 1)}$ . In this special case we can still employ the same technique used in bounding the non-shared components by using the intermediate step of bounding the expression  $\|\tilde{\mathbf{a}}_r - \lambda_r^* \mathbf{a}_r^*\|_2$  where  $\tilde{\mathbf{a}}_r = \frac{(\text{res}_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r) + \text{res}_M \mathbf{v}_r)}{P_\Omega(\mathbf{I}, (\mathbf{b}_r)^2, (\mathbf{c}_r)^2) + 1}$ . This is the main advantage of restricting the problem to the equal tensor matrix weight case as it allows the proof technique derived for the non-shared component to be easily extended to the case of the shared component. As we will show in the analysis of Lemma 4 this advantage disappears when the weight of the tensor and matrix are allowed to be different.

Let  $\mathbf{D}, \mathbf{E}, \mathbf{F}, \mathbf{G}, \mathbf{H}, \mathbf{J}, \mathbf{P}$  be  $n \times n$  diagonal matrices with diagonal elements,

$$\begin{aligned} \mathbf{D}_{ii} &= \sum_{j,k} \delta_{ijk} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k) + 1 ; \quad \mathbf{E}_{ii} = \sum_{j,k} \delta_{ijk} \mathbf{b}_r^*(j) \mathbf{c}_r^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k); \\ \mathbf{F}_{ii} &= \sum_{j,k} \delta_{ijk} \mathbf{b}_m^*(j) \mathbf{c}_m^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k) ; \quad \mathbf{G}_{ii} = \sum_{j,k} \delta_{ijk} \mathbf{b}_m(j) \mathbf{c}_m(k) \mathbf{b}_r(j) \mathbf{c}_r(k); \\ \mathbf{H}_{ii} &= \sum_l \mathbf{v}_r^*(l) \mathbf{v}_r(l) ; \quad \mathbf{J}_{ii} = \sum_l \mathbf{v}_m^*(l) \mathbf{v}_r(l) ; \quad \mathbf{P}_{ii} = \sum_l \mathbf{v}_m(l) \mathbf{v}_r(l). \end{aligned}$$

Then the vector  $\tilde{\mathbf{a}}_r$  obtained after one pass of the inner loop of Algorithm 1 can be written as

$$\begin{aligned} \tilde{\mathbf{a}}_r &= \mathbf{D}^{-1} \left( \lambda_r^* \mathbf{E} \mathbf{a}_r^* + \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \mathbf{a}_m^* - \lambda_m \mathbf{G} \mathbf{a}_m) + P_\Omega(\mathcal{E}_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r)) \right) \\ &\quad + \mathbf{D}^{-1} \left( \lambda_r^* \mathbf{H} \mathbf{a}_r^* + \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{J} \mathbf{a}_m^* - \lambda_m \mathbf{P} \mathbf{a}_m) + \mathcal{E}_M \mathbf{v}_r \right). \end{aligned} \quad (3.53)$$

In the next steps we bound

$$\begin{aligned}
\|\tilde{\mathbf{a}}_r - \lambda_r^* \mathbf{a}_r^*\|_2 &\leq \underbrace{\|\lambda_r^* \mathbf{D}^{-1} (\mathbf{E} + \mathbf{H} - \mathbf{D}) \mathbf{a}_r^*\|_2}_{err_1} + \underbrace{\|\mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \mathbf{a}_m^* - \lambda_m \mathbf{G} \mathbf{a}_m)\|_2}_{err_2} \\
&\quad + \underbrace{\|\mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{J} \mathbf{a}_m^* - \lambda_m \mathbf{P} \mathbf{a}_m)\|_2}_{err_3} + \underbrace{\|\mathbf{D}^{-1} (P_\Omega(\mathcal{E}_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r)) + \mathcal{E}_M \mathbf{v}_r)\|_2}_{err_4}.
\end{aligned} \tag{3.54}$$

In the shared component case, the right hand side of equation (3.54) can be characterized as the sum of 4 sources of errors, where  $err_1 = \lambda_r^* \mathbf{D}^{-1} (\mathbf{E} + \mathbf{H} - \mathbf{D}) \mathbf{a}_r^*$  can be characterized as the error due to the power method applied to both the tensor and matrix. This error is similar to  $err_1$  discussed in the proof of Lemma 2 with the exception that it factors in the contribution of the matrix. Again, if  $\mathcal{T}^*$  was a rank 1, noiseless tensor, then proving Lemma 3 would reduce to bounding this term. The second and third sources of error  $err_2 = \mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \mathbf{a}_m^* - \lambda_m \mathbf{G} \mathbf{a}_m)$  and  $err_3 = \mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{J} \mathbf{a}_m^* - \lambda_m \mathbf{P} \mathbf{a}_m)$  again represents the main challenge in the proof. The challenge in bounding these two errors are very similar to those exposed for  $err_2$  in the analysis of Lemma 2 in addition to the fact that we have an extra residual due to the matrix. If both the tensor and matrix components were orthogonal this error would be non existent. We therefore partly control these errors magnitude through the bound imposed on the components vector inner product namely Assumption (1.iii) the incoherence assumption. The fourth error term  $err_4 = \mathbf{D}^{-1} (P_\Omega(\mathcal{E}_T(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r)) + \mathcal{E}_M \mathbf{v}_r)$  is simply the error due to the noise of the tensor and the matrix and can be easily bounded after standard Assumptions are made about the spectral norms of  $\mathcal{E}_T$  and  $\mathcal{E}_M$ . At first glance it might seem that right hand-side of the inequalities in equation (3.54) is larger than that found in equation (3.41) making therefore the bound on the shared component larger than that of the that of the non-shared component. However as we demonstrate in the proof below, the component  $\mathbf{D}^{-1}$  plays the role of a weight which averages the tensor and matrix sources of error in equation (3.54).

We start with bounding the first error term,

$$\begin{aligned}
\|err_1\|_2 &= \|\lambda_r^* \mathbf{D}^{-1} (\mathbf{E} + \mathbf{H} - \mathbf{D}) a_r^*\|_2 \\
&\leq \lambda_r^* \|\mathbf{D}^{-1} (\mathbf{E} + \mathbf{H} - \mathbf{D})\|_2 \|a_r^*\|_2 \\
&\leq \lambda_r^* \max_i |\mathbf{D}_{ii}^{-1}| \|(\mathbf{E} + \mathbf{H} - \mathbf{D})_{ii}\|,
\end{aligned}$$

where last inequality above is obtained by observing that  $\mathbf{D}^{-1} (\mathbf{E} + \mathbf{H} - \mathbf{D})$  is a diagonal matrix whose spectral norm is the maximum absolute value of its diagonal elements and that  $\|a_r^*\|_2 = 1$ . We proceed to getting an upper bound for each of the maximum of each of the random variable elements in the equation above with high probability. To do that we first get an upper bound on each of the diagonal elements with high probability and make use of the union bound method to get a high probability bound on the maximums.

$$\begin{aligned}
|(\mathbf{E} + \mathbf{H} - \mathbf{D})_{ii}| &\leq |\langle \mathbf{v}_r^*, \mathbf{v}_r \rangle - 1| + \left| \sum_{jk} \delta_{ijk} \mathbf{b}_r^*(j) \mathbf{c}_r^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k) - \sum_{jk} \delta_{ijk} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k) \right| \\
&= \frac{1}{2} \|\mathbf{d}_v\|_2^2 + \left| \sum_{ij} \delta_{ijk} \mathbf{a}_r^*(i) \mathbf{d}_{b_r}(j) \mathbf{a}_r(i) \mathbf{b}_r(j) - \sum_{ij} \delta_{ijk} \mathbf{d}_{a_r}(i) \mathbf{b}_r^*(j) \mathbf{a}_r(i) \mathbf{b}_r(j) \right. \\
&\quad \left. - \sum_{ij} \delta_{ijk} \mathbf{d}_{c_r}(i) \mathbf{d}_{b_r}(j) \mathbf{c}_r(i) \mathbf{b}_r(j) \right| \\
&\leq \frac{1}{2} \|\mathbf{d}_v\|_2^2 + p (|\langle \mathbf{c}_r^*, \mathbf{c}_r \rangle \langle \mathbf{d}_{b_r}, \mathbf{b}_r \rangle| + |\langle \mathbf{d}_{c_r}, \mathbf{c}_r \rangle \langle \mathbf{b}_r^*, \mathbf{b}_r \rangle| + |\langle \mathbf{d}_{c_r}, \mathbf{c}_r \rangle \langle \mathbf{d}_{b_r}, \mathbf{b}_r \rangle|) \\
&\quad + p\gamma (\|\mathbf{d}_{c_r}\|_2 + \|\mathbf{d}_{b_r}\|_2 + \|\mathbf{d}_{c_r}\|_2 \|\mathbf{d}_{b_r}\|_2).
\end{aligned}$$

The expression on the right side of the equality is obtained by combining the triangle inequality to the fact that  $\mathbf{c}_r(i) = \mathbf{c}_r^*(i) + \mathbf{d}_{c_r}(i)$   $\mathbf{b}_r(j) = \mathbf{b}_r^*(j) + \mathbf{d}_{b_r}(j)$  and using the results from Lemma 12. We then use Lemma 7 to bound the three random elements inside the absolute value. Hence, provided the reveal probability  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(n^{10})}{n^{3/2}\gamma^2}$  we get,

$$\begin{aligned}
|(\mathbf{E} + \mathbf{H} - \mathbf{D})_{ii}| |(\mathbf{E} + \mathbf{H} - \mathbf{D})_{ii}| &\leq \frac{1}{2} \|\mathbf{d}_v\|_2^2 + 6p \left( \max_{\mathbf{u}_r \in \{c_r, b_r\}} \left\{ \sqrt{1 - \frac{\|d_{u_r}\|_2}{2}} \|\mathbf{d}_{u_r}\|_2^2, \|\mathbf{d}_{u_r}\|_2^4, \gamma \|\mathbf{d}_{u_r}\|_2 \right\} \right) \\
&\leq \frac{1}{2} \|\mathbf{d}_v\|_2^2 + 6p \max_{\mathbf{u}_r \in \{c_r, b_r\}} \left( \sqrt{1 - \frac{\|d_{u_r}\|_2}{3}} \|\mathbf{d}_{u_r}\|_2, \|\mathbf{d}_{u_r}\|_2^3, \gamma \right) \|\mathbf{d}_{u_r}\|_2,
\end{aligned} \tag{3.55}$$

with probability  $1 - 2n^{-10}$ . Using the union bound on the result in equation (3.55) combined with the results of Lemma 6. We get,

$$\|err_1\|_2 \leq \frac{\lambda_r^* \left( 6p \max_{\mathbf{u}_r \in \{a_r, b_r\}} \left( \sqrt{1 - \frac{\|\mathbf{d}_{u_r}\|_2}{2}} \|\mathbf{d}_{u_r}\|_2, \|\mathbf{d}_{u_r}\|_2^3, \gamma \right) \|\mathbf{d}_{u_r}\|_2 + 1/2 \|\mathbf{d}_v\|_2^2 \right)}{p(1 - \gamma) + 1} \quad (3.56)$$

with probability  $1 - 2n^{-9}$ .

Next we proceed to bound  $\|err_3\|_2$  before coming back to  $\|err_2\|_2$ ,

$$\|err_3\|_2 = \|\mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{J} a_m^* - \lambda_m \mathbf{P} a_m)\|_2.$$

We start by bounding the component inside the summation.

$$\begin{aligned} \|\lambda_m^* \mathbf{J} a_m^* - \lambda_m \mathbf{P} a_m\|_2 &= \|\lambda_m^* \langle \mathbf{v}_m^*, \mathbf{v}_r \rangle \mathbf{a}_m^* - \lambda_m \langle \mathbf{v}_m, \mathbf{v}_r \rangle \mathbf{a}_m\|_2 \\ &= \lambda_m^* \|(\langle \mathbf{v}_m^*, \mathbf{v}_r \rangle - \langle \mathbf{v}_m, \mathbf{v}_r \rangle) \mathbf{a}_m^* + \langle \mathbf{v}_m, \mathbf{v}_r \rangle \mathbf{d}_{a_m} + \Delta_{\lambda_m} \langle \mathbf{v}_m, \mathbf{v}_r \rangle \mathbf{a}_m\|_2 \\ &\leq 3\lambda_m^* \max \left( \|\mathbf{d}_{\mathbf{v}_m}\|_2, \frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_{v_r}\|_2 \right) \|\mathbf{d}_{v_r}\|_2, \end{aligned} \quad (3.57)$$

where the last inequality is due to the fact that  $\langle \mathbf{v}_m, \mathbf{v}_r \rangle \leq (\frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_{v_r}\|_2)$ . This, combined with the results of Lemma 6 to bound  $|\mathbf{D}^{-1}|$  yields,

$$\|err_3\|_2 \leq \frac{3 \sum_{m \in [R] \setminus r} \lambda_m^* \max(\|\mathbf{d}_{\mathbf{v}_m}\|_2, \frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_{v_r}\|_2) \|\mathbf{d}_{v_r}\|_2}{p(1 - \gamma) + 1}, \quad (3.58)$$

with probability  $1 - 2n^{-9}$ .

The technique used to bound  $\|err_2\|_2$  in this section is very similar to the one used to bound expression in section. We therefore provide the bound and incite the reader to review the section mention to understand the process involved. The main difference recedes in

substituting the components  $\mathbf{c}$  for  $\mathbf{a}$  and finding a lower bound for  $D^{-1}$  using Lemma 6. This yields,

$$\|err_2\|_2 \leq \frac{8p \sum_{m \in [R] \setminus r} \lambda_m^* \max_{\mathbf{u} \in \{\mathbf{c}_m, \mathbf{b}_m, \mathbf{c}_r, \mathbf{b}_r\}} \left( \left( \frac{c_0}{\sqrt{(n)}} + \|\mathbf{d}_u\|_2 \right), \left( \frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_u\|_2 \right)^2, \gamma \right) \|\mathbf{d}_u\|_2}{p(1 - \gamma) + 1}, \quad (3.59)$$

with probability  $1 - 2n^{-9}$ .

Next  $\|err_4\|_2$  is bounded using Lemma 11, combined with Lemma 6 and the fact that  $\|\mathcal{E}_M \mathbf{v}_r\|_2 \leq \|\mathcal{E}_M\|$  since  $\|\mathbf{v}_r\|_2 = 1$  and by definition  $\|\mathcal{E}_M\| = \sup_{\|\mathbf{u}\|=1} \|\mathcal{E}_M \mathbf{u}\|_2$ . This therefore yields

$$\|err_4\|_2 \leq \frac{p(1 + \gamma)\|\mathcal{E}_T\| + \|\mathcal{E}_M\|}{p(1 - \gamma) + 1} \quad (3.60)$$

with probability  $1 - 2n^{-9}$ .

Combining the error bounds results of  $\|err_1\|_2$ ,  $\|err_3\|_2$ ,  $\|err_2\|_2$ ,  $\|err_4\|_2$  in equations (3.56), (3.59), (3.58) and (3.60) respectively, we get

$$\begin{aligned} & \|\tilde{\mathbf{a}}_r - \lambda_r^* \mathbf{a}_r^*\|_2 \\ & \leq \frac{8pR\lambda_{max}^* \max_{\mathbf{u} \in \{\mathbf{c}_m, \mathbf{b}_m, \mathbf{c}_r, \mathbf{b}_r\}} \left( \sqrt{1 - \frac{\|\mathbf{d}_u\|_2^2}{2}} \|\mathbf{d}_u\|_2, \left( \frac{c_0}{\sqrt{(n)}} + \|\mathbf{d}_u\|_2 \right), \left( \frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_u\|_2 \right)^2, \|\mathbf{d}_u\|_2^3, \gamma \right) \|\mathbf{d}_u\|_2}{p(1 - \gamma) + 1} \\ & + \frac{3R\lambda_{max}^* \max \left( \|\mathbf{d}_{v_r}\|_2, \frac{c_0}{\sqrt{(n)}} + 3\|\mathbf{d}_{v_r}\|_2 \right) \|\mathbf{d}_{v_r}\|_2 + p(1 + \gamma)\|\mathcal{E}_T\| + \|\mathcal{E}_M\|}{p(1 - \gamma) + 1} \end{aligned} \quad (3.61)$$

with probability  $1 - 2n^{-9}$ .

The proof of Lemma 3 is then completed by applying the results of Lemma 10 which shows that  $\|\mathbf{a}_r - \mathbf{a}_r^*\|_2 \leq \frac{2}{\lambda_r^*} \|\tilde{\mathbf{a}}_r - \lambda_r^* \mathbf{a}_r^*\|_2$  and letting  $\max\{\|\mathbf{d}_u\|_2\} = \epsilon_T$  and  $\max\{\|\mathbf{d}_v\|_2\} = \epsilon_M$ .  $\square$

### 3.8.4 Proof of Lemma 4

We now prove Lemma 4 which establishes an error contraction result for the shared tensor components in one iteration of Algorithm 1 when the input tensor and matrix are assumed to be sparse and their respective components weight are allowed to differ. First, we introduce



some notation below in order reveal how we address the sparse components in the analysis . Define  $F_a := \text{supp}(\mathbf{a}_r^*) \cup \text{supp}(\mathbf{a}_r)$ ,  $F_b := \text{supp}(\mathbf{b}_r^*) \cup \text{supp}(\mathbf{b}_r)$ ,  $F_c := \text{supp}(\mathbf{c}_r^*) \cup \text{supp}(\mathbf{c}_r)$  and  $F_v := \text{supp}(\mathbf{v}_r^*) \cup \text{supp}(\mathbf{v}_r)$  where  $\text{supp}(\mathbf{u})$  refers to the set of indices in a vector  $\mathbf{u}$  that are nonzero. Then let  $F$  and  $F$  be compositions of support sets defined as  $F := F_a \circ F_b \circ F_c$  and  $F := F_1 \circ F_v$  respectively. We use the notation  $\mathcal{T}^{\setminus r} := \sum_{m \in [R] \setminus r} \lambda_m \mathbf{a}_m \otimes \mathbf{b}_m \otimes \mathbf{c}_m$  to represent the CP decomposition of the tensor  $\mathcal{T}$  minus its  $r^{\text{th}}$  rank 1 tensor element  $(\lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r)$ . Denote the truncated vectors  $\mathbf{u}_r^*$  and  $\mathbf{u}_r$  to be  $\bar{\mathbf{u}}_r^* = \text{Truncate}(\mathbf{u}_r^*, F_{\mathbf{u}})$  and  $\bar{\mathbf{u}}_r = \text{Truncate}(\mathbf{u}_r, F_{\mathbf{u}})$  with  $\mathbf{u} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{v}\}$  and  $r = 1, \dots, R$ .

Note that in the update of  $a_r$  in our algorithm, we first obtain non-sparse estimator  $\mathbf{a}_r$  in line (8) of algorithm 1 then update it by applying the truncation method and normalization method in (9). We let  $\dot{\mathbf{a}}_r$  be the update on line (8) of algorithm 1 before the truncation and  $\mathbf{a}_r$  be the truncated update on line (9) of the algorithm. That is  $\mathbf{a}_r = \frac{\dot{\mathbf{a}}_r}{\|\dot{\mathbf{a}}_r\|_2}$  with,

$$\dot{\mathbf{a}}_r = \frac{(\lambda_r \text{res}_{T_F}(\mathbf{I}, \mathbf{b}_r, \mathbf{c}_r) + \omega_r \text{res}_{M_F} \mathbf{v}_r)}{(\lambda_r^2 P_{\Omega}(\mathbf{I}, (\mathbf{b}_r)^2, (\mathbf{c}_r)^2) + \omega_r^2)}$$

where  $\text{res}_{T_F}$  denotes the restriction of the residual tensor  $\text{res}_T$  on the three modes indexed by  $F_a$ ,  $F_b$  and  $F_c$  and  $\text{res}_{T_F}$  is the equivalent for the residual matrix  $\text{res}_M$ . That is

$$\begin{aligned} \text{res}_{T_F} &= \sum_{m \in [R]} \lambda_m^* \bar{\mathbf{a}}_m^* \otimes \bar{\mathbf{b}}_m^* \otimes \bar{\mathbf{c}}_m^* - \sum_{m \in [R] \setminus r} \lambda_m \bar{\mathbf{a}}_m \otimes \bar{\mathbf{b}}_m \otimes \bar{\mathbf{c}}_m, \\ \text{res}_{M_F} &= \sum_{m \in [R]} \omega_m^* \bar{\mathbf{a}}_m^* \otimes \bar{\mathbf{v}}_m^* - \sum_{m \in [R] \setminus r} \lambda_m \bar{\mathbf{a}}_m \otimes \bar{\mathbf{v}}_m. \end{aligned}$$

Proving Lemma 4 involves bounding  $\|\mathbf{a}_r - \mathbf{a}_r^*\|_2$  which we do in two steps. First we notice that  $\|\mathbf{a}_r - \mathbf{a}_r^*\|_2 \leq \|\mathbf{a}_r - \dot{\mathbf{a}}_r\|_2 + \|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2$  using the triangle inequality. Then we bound each of the two norms in the expression above. As will be demonstrated in the proof,

$$\|\mathbf{a}_r - \mathbf{a}_r^*\|_2 \leq \|\mathbf{a}_r - \dot{\mathbf{a}}_r\|_2 + \|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2 \leq 2\|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2.$$

While bounding  $\|\mathbf{a}_r - \mathbf{a}_r^*\|_2$  directly is a challenge, getting relatively tight upper bounds for  $\|\mathbf{a}_r - \dot{\mathbf{a}}_r\|_2$  and  $\|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2$  although challenging is feasible.

**Step1:** We begin with bounding  $\|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2$ .

Let  $\mathbf{D}, \mathbf{E}, \mathbf{F}, \mathbf{G}, \mathbf{H}, \mathbf{J}, \mathbf{P}$  be  $n \times n$  diagonal matrices with diagonal elements,

$$\begin{aligned}\mathbf{D}_{ii} &= \lambda_r^2 \sum_{j,k} \delta_{ijk} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k) + \omega_r^2 ; \quad \mathbf{E}_{ii} = \sum_{j,k} \delta_{ijk} \bar{\mathbf{b}}_r^*(j) \bar{\mathbf{c}}_r^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k); \\ \mathbf{F}_{ii} &= \sum_{j,k} \delta_{ijk} \bar{\mathbf{b}}_m^*(j) \bar{\mathbf{c}}_m^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k) ; \quad \mathbf{G}_{ii} = \sum_{j,k} \delta_{ijk} \bar{\mathbf{b}}_m(j) \bar{\mathbf{c}}_m(k) \mathbf{b}_r(j) \mathbf{c}_r(k); \\ \mathbf{H}_{ii} &= \sum_l \bar{\mathbf{v}}_r^*(l) \mathbf{v}_r(l) ; \quad \mathbf{J}_{ii} = \sum_l \bar{\mathbf{v}}_m^*(l) \mathbf{v}_r(l) ; \quad \mathbf{P}_{ii} = \sum_l \bar{\mathbf{v}}_m(l) \mathbf{v}_r(l).\end{aligned}$$

Then the vector  $\mathbf{a}_r$  obtained after one pass of the inner loop of Algorithm 1 and before normalization can be written as

$$\begin{aligned}\dot{\mathbf{a}}_r &= \lambda_r \mathbf{D}^{-1} \left( \lambda_r^* \mathbf{E} \bar{\mathbf{a}}_r^* + \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \bar{\mathbf{a}}_m^* - \lambda_m \mathbf{G} \bar{\mathbf{a}}_m) + \mathcal{E}_{T_F} \times_2 \mathbf{b}_r \times_3 \mathbf{c}_r \right) \\ &+ \omega_r \mathbf{D}^{-1} \left( \omega_r^* \mathbf{H} \bar{\mathbf{a}}_r^* + \sum_{m \in [R] \setminus r} (\omega_m^* \mathbf{J} \bar{\mathbf{a}}_m^* - \omega_m \mathbf{P} \bar{\mathbf{a}}_m) + \mathcal{E}_{M_F} \mathbf{v}_r \right).\end{aligned}\tag{3.62}$$

This means that

$$\begin{aligned}\|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2 &= \underbrace{\|\mathbf{D}^{-1} (\lambda_r \lambda_r^* \mathbf{E} + \omega_r \omega_r^* \mathbf{H} - \mathbf{D} \mathbf{I}) \bar{\mathbf{a}}_r^*\|_2}_{err_1} + \underbrace{\|\lambda_r \mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \bar{\mathbf{a}}_m^* - \lambda_m \mathbf{G} \bar{\mathbf{a}}_m)\|_2}_{err_2} \\ &+ \underbrace{\|\omega_r \mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\omega_m^* \mathbf{J} \bar{\mathbf{a}}_m^* - \omega_m \mathbf{P} \bar{\mathbf{a}}_m)\|_2}_{err_3} + \underbrace{\|\mathbf{D}^{-1} (\lambda_r \mathcal{E}_{T_F} \times_2 \mathbf{b}_r \times_3 \mathbf{c}_r + \omega_r \mathcal{E}_{M_F} \mathbf{v}_r)\|_2}_{err_4}.\end{aligned}\tag{3.63}$$

The right hand side of the inequality above is split into four sources of errors where  $err_2$  and  $err_3$  are due to tensor rank being greater than one,  $err_3$  is the error associated tot the tensor and matrix noise and  $err_1$  is the error from the power iteration used in the algorithm. We notice in the case where the tensor and matrix have different weight expression of  $\mathbf{a}_r$  contains the estimated weights unlike when the tensor weights can be assumed to be equal. This main difference requires careful derivation of the error bound for the update of the shared components.

We start with bounding the first error term

$$\begin{aligned}
\|err_1\|_2 &= \|\mathbf{D}^{-1} (\lambda_r \lambda_r^* \mathbf{E} + \omega_r \omega_r^* \mathbf{H} - \mathbf{DI}) \bar{\mathbf{a}}_r^*\|_2 \\
&\leq \|\mathbf{D}^{-1} (\lambda_r \lambda_r^* \mathbf{E} + \omega_r \omega_r^* \mathbf{H} - \mathbf{DI})\|_2 \|\bar{\mathbf{a}}_r^*\|_2 \\
&\leq \max_i \underbrace{|\mathbf{D}_{ii}^{-1}|}_{err_{11}} \underbrace{|(\lambda_r \lambda_r^* \mathbf{E} + \omega_r \omega_r^* \mathbf{H} - \mathbf{DI})_{ii}|}_{err_{12}},
\end{aligned}$$

where the third inequality is due to the fact that  $\|\bar{\mathbf{a}}_r^*\|_2 \leq \|\mathbf{a}_r^*\|_2 = 1$  and since,  $\mathbf{D}^{-1} (\lambda_r \lambda_r^* \mathbf{E} + \omega_r \omega_r^* \mathbf{H} - \mathbf{DI})$  is a diagonal matrix hence its spectral norm is obtained by taking the maximum absolute value of its diagonal elements. We therefore proceed to getting an upper bound each of the maximum of each of the random variable elements in the equation above with high probability. To do that we first get an upper bound on each of the diagonal elements with high probability and make use of the union bound method to get a high probability bound on the maximums.

$$\begin{aligned}
err_{12} &= |\lambda_r \lambda_r^* \sum_{jk} \delta_{ijk} \bar{\mathbf{b}}_r^*(j) \bar{\mathbf{c}}_r^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k) + \omega_r \omega_r^* \langle \bar{\mathbf{v}}_r^*, \mathbf{v}_r \rangle - (\lambda_r^2 \sum_{jk} \delta_{ijk} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k) + \omega_r^2)| \\
&\leq \underbrace{|\lambda_r \lambda_r^* \sum_{jk} \delta_{ijk} \bar{\mathbf{b}}_r^*(j) \bar{\mathbf{c}}_r^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k) - \lambda_r^2 \sum_{jk} \delta_{ijk} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k)|}_{I_{121}} + \underbrace{|\omega_r \omega_r^* \langle \bar{\mathbf{v}}_r^*, \mathbf{v}_r \rangle - \omega_r^2|}_{I_{122}}.
\end{aligned}$$

We can bound  $I_{121}$  and  $I_{122}$  next

$$\begin{aligned}
I_{122} &= |\omega_r \omega_r^* \langle \bar{\mathbf{v}}_r^*, \mathbf{v}_r \rangle - \omega_r^2| \\
&\leq \omega_r \omega_r^* (|\langle \bar{\mathbf{v}}_r^*, \mathbf{v}_r \rangle - 1| + \Delta_{\omega_r}) \\
&\leq \omega_r \omega_r^* \left( \frac{1}{2} \|d_v\|_2^2 + \Delta_{\omega_r} \right)
\end{aligned} \tag{3.64}$$

where the first inequality is due to using the triangle inequality, the fact that  $\omega_r = \omega_r - \omega_r^* + \omega_r^*$  and Lemma 13 by noting that  $\text{supp}(\mathbf{v}_r) \subseteq F_b$ . The second inequality is obtained from the results of Lemma 12. Next we also bound  $I_{121}$ .

$$\begin{aligned}
I_{121} &= |\lambda_r \lambda_r^* \sum_{jk} \delta_{ijk} \bar{\mathbf{b}}_r^*(j) \bar{\mathbf{c}}_r^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k) - \lambda_r^2 \sum_{jk} \delta_{ijk} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k)| \\
&\leq \lambda_r \lambda_r^* (|\sum_{jk} (\delta_{ijk} \bar{\mathbf{b}}_r^*(j) \bar{\mathbf{c}}_r^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k) - \delta_{ijk} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k))| + \Delta_{\lambda_r} \sum_{jk} \delta_{ijk} \mathbf{b}_r^2(j) \mathbf{c}_r^2(k)) \\
&\leq |\sum_{jk} \delta_{ijk} \mathbf{b}_r^*(j) \mathbf{d}_{c_r}^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k)| + |\sum_{jk} \delta_{ijk} \mathbf{d}_{b_r}^*(j) \mathbf{c}_r^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k)| \\
&\quad + |\sum_{jk} \delta_{ijk} \mathbf{d}_{b_r}^*(j) \mathbf{d}_{c_r}^*(k) \mathbf{b}_r(j) \mathbf{c}_r(k)|,
\end{aligned}$$

where the last inequality is obtained using the triangle inequality and the fact that  $\mathbf{b}_r(j) = \mathbf{b}_r^*(j) + \mathbf{d}_{b_r}(j)$  and  $\mathbf{c}_r(j) = \mathbf{c}_r^*(j) + \mathbf{d}_{c_r}(j)$  combined with the fact that  $F_b = \text{supp}(\mathbf{b}_r^*) \subseteq \text{supp}(\bar{\mathbf{b}}_r^*) = F$  and  $F_c = \text{supp}(\mathbf{c}_r^*) \subseteq \text{supp}(\bar{\mathbf{c}}_r^*) = F$  which means that  $\bar{\mathbf{b}}_r^*(k) - \mathbf{b}_r^*(k) = 0$  and  $\bar{\mathbf{c}}_r^*(k) - \mathbf{c}_r^*(k) = 0$ . Next applying the results of Lemma 6 and Lemma 9, we get

$$\begin{aligned}
I_{121} &\leq \lambda_r^* \lambda_r p (|\langle \mathbf{b}_r^*, \mathbf{b}_r \rangle \langle \mathbf{d}_{c_r}, \mathbf{c}_r \rangle| + |\langle \mathbf{d}_{b_r}, \mathbf{b}_r \rangle \langle \mathbf{c}_r^*, \mathbf{c}_r \rangle| + |\langle \mathbf{d}_{b_r}, \mathbf{b}_r \rangle \langle \mathbf{d}_{c_r}, \mathbf{c}_r \rangle| + \Delta_{\lambda_r}) \\
&\quad + p\gamma (\|\mathbf{d}_{c_r}\|_2 + \|\mathbf{d}_{b_r}\|_2 + \|\mathbf{d}_{c_r}\|_2 \|\mathbf{d}_{b_r}\|_2 + \Delta_{\lambda_r}) \\
&\leq 8\lambda_r^* \lambda_r p \left( \max_{\mathbf{u}_r \in \{c_r, b_r\}} \left\{ \sqrt{1 - \frac{\|\mathbf{d}_{u_r}\|_2}{2}} \|\mathbf{d}_{u_r}\|_2^2, \|\mathbf{d}_{u_r}\|_2^4, \Delta_{\lambda_r}, \gamma \|\mathbf{d}_{u_r}\|_2, \gamma \Delta_{\lambda_r} \right\} \right), \quad (3.65)
\end{aligned}$$

where the last inequality above holds with probability  $1 - 2d^{-10}$  provided the reveal probability  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(d^{10})}{d^{3/2}\gamma^2}$ . Combining equations (3.64) and (3.65) followed by making use of lemma (6) to bound the denominator of  $\|\text{err}_1\|_2$ , we get

$$\|\text{err}_1\|_2 \leq \frac{8\lambda_r^* \lambda_r p \left( \max_{\mathbf{u}_r \in \{c_r, b_r\}} \left\{ \sqrt{1 - \frac{\|\mathbf{d}_{u_r}\|_2}{2}} \|\mathbf{d}_{u_r}\|_2^2, \|\mathbf{d}_{u_r}\|_2^4, \Delta_{\lambda_r}, \gamma \|\mathbf{d}_{u_r}\|_2, \gamma \Delta_{\lambda_r} \right\} \right) + \omega_r \omega_r^* (\frac{1}{2} \|\mathbf{d}_v\|_2^2 + \Delta_{\omega_r})}{\lambda_r^2 p (1 - \gamma) + \omega_r^2}, \quad (3.66)$$

with probability  $1 - 2d^{-9}$ .

We now move on to bounding the expression  $\|err_3\|_2$ .

$$\begin{aligned}
\|err_3\|_2 &\leq \omega_r \|\mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\omega_m^* \mathbf{J} \bar{\mathbf{a}}_m^* - \omega_m \mathbf{P} \bar{\mathbf{a}}_m)\|_2 \\
&\leq \omega_r \max_i |\mathbf{D}_{ii}^{-1}| \sum_{m \in [R] \setminus r} \|\omega_m^* \langle \bar{\mathbf{v}}_m^*, \mathbf{v}_r \rangle \bar{\mathbf{a}}_m^* - \omega_m \langle \bar{\mathbf{v}}_m, \mathbf{v}_r \rangle \bar{\mathbf{a}}_m\|_2 \\
&\leq \omega_r \max_i |\mathbf{D}_{ii}^{-1}| \sum_{m \in [R] \setminus r} \omega_m^* (|\langle \bar{\mathbf{v}}_m^*, \mathbf{v}_r \rangle - \langle \bar{\mathbf{v}}_m, \mathbf{v}_r \rangle| \|\bar{\mathbf{a}}_m^*\|_2 + |\langle \bar{\mathbf{v}}_m, \mathbf{v}_r \rangle| \|\mathbf{d}_{a_m}\|_2) \\
&\quad + \omega_r \max_i |\mathbf{D}_{ii}^{-1}| \sum_{m \in [R] \setminus r} \omega_m^* (\Delta_{\omega_m} |\langle \bar{\mathbf{v}}_m, \mathbf{v}_r \rangle| \|\bar{\mathbf{a}}_m\|_2), \tag{3.67}
\end{aligned}$$

where for inequality three, we use the fact that  $\|\langle \bar{\mathbf{v}}_m^*, \mathbf{v}_r \rangle \bar{\mathbf{a}}_m^*\|_2 \leq \|\langle \mathbf{v}_m^*, \mathbf{v}_r \rangle \mathbf{a}_m^*\|_2$  since  $\|\bar{\mathbf{a}}_m^*\|_2 \leq 1$  and that the truncation process is invariant to scaling. We also used the fact that  $\omega_r = \omega_r - \omega_r^* + \omega_r^*$ . Next, since  $\{\text{supp}(\mathbf{v}_m^*), \text{supp}(\mathbf{v}_m)\} \subseteq F$  it follows that  $\langle \bar{\mathbf{v}}_m^*, \mathbf{v}_r \rangle - \langle \bar{\mathbf{v}}_m, \mathbf{v}_r \rangle = \langle \mathbf{d}_{\mathbf{v}_m}, \mathbf{v}_r \rangle$ . Then noticing that  $\langle \mathbf{v}_m, \mathbf{v}_r \rangle \leq (\frac{c_0}{\sqrt{d}} + 3\|\mathbf{d}_{v_r}\|_2)$  and using the results of Lemma 6 to bound  $\max_i |\mathbf{D}_{ii}^{-1}|$  yields

$$\|err_3\|_2 \leq \frac{\omega_r \sum_{m \in [R] \setminus r} \omega_m^* (\|\mathbf{d}_{\mathbf{v}_m}\|_2 + (\frac{c_0}{\sqrt{d}} + 3\|\mathbf{d}_{v_r}\|_2) \|\mathbf{d}_{a_m}\|_2 + \Delta_{\omega_m} (\frac{c_0}{\sqrt{d}} + 3\|\mathbf{d}_{v_r}\|_2))}{\lambda_r^2 p(1 - \gamma) + \omega_r^2}, \tag{3.68}$$

with probability  $1 - 2d^{-9}$  provided the reveal probability  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(d^{10})}{d^{3/2}\gamma^2}$ .

Next we bound the expression  $\|err_2\|_2$  as

$$\begin{aligned}
\|err_2\|_2 &= \|\lambda_r \mathbf{D}^{-1} \sum_{m \in [R] \setminus r} \lambda_m^* (\mathbf{F} \bar{\mathbf{a}}_m^* - \mathbf{G} \bar{\mathbf{a}}_m + \Delta_{\lambda_m} \mathbf{G} \bar{\mathbf{a}}_m)\|_2 \\
&\leq \lambda_r \|\mathbf{D}^{-1}\|_2 \sum_{m \in [R] \setminus r} \lambda_m^* (\|(\mathbf{F} - \mathbf{G}) \bar{\mathbf{a}}_m^*\|_2 + \|\mathbf{G} \mathbf{d}_{a_m}\|_2 + \|\Delta_{\lambda_m} \mathbf{G} \bar{\mathbf{a}}_m\|_2) \\
&\leq \lambda_r \|\mathbf{D}^{-1}\|_2 \sum_{m \in [R] \setminus r} \lambda_m^* \left( \underbrace{\max_i |(\mathbf{F} - \mathbf{G})_{ii}|}_{I_{21}} + (\|\mathbf{d}_{a_m}\|_2 + \Delta_{\lambda_m}) \underbrace{\max_i |\mathbf{G}_{ii}|}_{I_{22}} \right), \tag{3.69}
\end{aligned}$$

where the second inequality is due to the triangle inequality and the third inequality is due to the fact that  $\|\bar{\mathbf{a}}_m^*\|_2 \leq \|\mathbf{a}_m^*\|_2 = 1$  and  $\|\bar{\mathbf{a}}_m\|_2 \leq \|\mathbf{a}_m\|_2 = 1$  as well as the fact that

the matrices  $\|\mathbf{F} - \mathbf{G}\|_2$  and  $\|\mathbf{G}\|_2$  are diagonal matrices hence their spectral norm is their maximum absolute diagonal value. We focus on bounding  $I_{21}$  and  $I_{22}$  next.

$$\begin{aligned}
I_{21} &= \left| \sum_{jk} \delta_{ijk} \bar{\mathbf{c}}_m^*(k) \bar{\mathbf{b}}_m^*(j) \mathbf{c}_r(k) \mathbf{b}_r(j) - \sum_{jk} \delta_{ijk} \bar{\mathbf{c}}_m(k) \bar{\mathbf{b}}_m(j) \mathbf{c}_r(k) \mathbf{b}_r(j) \right| \\
&\leq \left| \sum_{jk} \delta_{ijk} \mathbf{d}_{c_m}(k) \bar{\mathbf{b}}_m^*(j) \mathbf{c}_r(k) \mathbf{b}_r(j) \right| + \left| \sum_{jk} \delta_{ijk} \bar{\mathbf{c}}_m^*(k) \mathbf{d}_{b_m}(j) \mathbf{c}_r(k) \mathbf{b}_r(j) \right| \\
&\quad + \left| \sum_{jk} \delta_{ijk} \mathbf{d}_{c_m}(k) \mathbf{d}_{b_m}(j) \mathbf{c}_r(k) \mathbf{b}_r(j) \right| \\
&\leq p \left( |\langle \mathbf{d}_{c_m}, \mathbf{c}_r \rangle \langle \bar{\mathbf{b}}_m^*, \mathbf{b}_r \rangle| + |\langle \bar{\mathbf{c}}_m^*, \mathbf{c}_r \rangle \langle \mathbf{d}_{b_m}, \mathbf{b}_r \rangle| + |\langle \mathbf{d}_{c_m}, \mathbf{c}_r \rangle \langle \mathbf{d}_{b_m}, \mathbf{b}_r \rangle| \right) \\
&\quad + \gamma (\|\mathbf{d}_{c_m}\|_2 + \|\mathbf{d}_{b_m}\|_2 + \|\mathbf{d}_{c_m}\|_2 \|\mathbf{d}_{b_m}\|_2) \\
&\leq 6p \max_{\mathbf{u} \in \{\mathbf{c}_m, \mathbf{b}_m, \mathbf{c}_r, \mathbf{b}_r\}} \left( \left( \frac{c_0}{\sqrt{d}} + \|\mathbf{d}_u\|_2 \right), \|\mathbf{d}_u\|_2, \gamma \right) \|\mathbf{d}_u\|_2. \tag{3.70}
\end{aligned}$$

The last inequality above holds with probability  $1 - 2d^{-10}$  provided the reveal probability  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(d^{10})}{d^{3/2}\gamma^2}$ . The third inequality is due to Lemma 7 by noting that since  $\text{supp}(\mathbf{b}_m^*) \subseteq F_b$  then  $\bar{\mathbf{b}}_m^*(j) \leq \frac{\mu}{\sqrt{d}}$ . Similarly using Lemma 8, and applying the union bound and the fact that  $|\langle \mathbf{c}_m, \mathbf{c}_r \rangle \langle \mathbf{b}_m, \mathbf{b}_r \rangle| \leq \max\{\langle \mathbf{c}_m, \mathbf{c}_r \rangle^2, \langle \mathbf{b}_m, \mathbf{b}_r \rangle^2\} \leq \left( \frac{c_0}{\sqrt{d}} + \max_{\mathbf{u} \in \{\mathbf{c}_r, \mathbf{b}_r\}} 3\|\mathbf{d}_u\|_2 \right)^2$  yields the following inequality

$$I_{22} \leq p \max_{\mathbf{u}_r \in \{\mathbf{c}_r, \mathbf{b}_r, \mathbf{c}_m, \mathbf{b}_m\}} \left( \left( \frac{c_0}{\sqrt{d}} + 3\|\mathbf{d}_{u_r}\|_2 \right)^2, \gamma \right), \tag{3.71}$$

with probability  $1 - 2d^{-9}$ .

Putting equations (3.69), (3.70), (3.71), and Lemma 6 together yields

$$\|err_2\|_2 \leq \frac{\lambda_r 8p \sum_{m \in [R] \setminus r} \lambda_m^* \max_{\mathbf{u} \in \{\mathbf{a}_m, \mathbf{b}_m, \mathbf{a}_r, \mathbf{b}_r\}} \left( \left( \frac{c_0}{\sqrt{d}} + \|\mathbf{d}_u\|_2 \right), \left( \frac{c_0}{\sqrt{d}} + 3\|\mathbf{d}_u\|_2 \right)^2, \|\mathbf{d}_u\|_2, \gamma \right) \|\mathbf{d}_u\|_2}{\lambda_r^2 p (1 - \gamma) + \omega_r^2}, \tag{3.72}$$

with probability  $1 - 2d^{-9}$  provided  $p \geq \frac{C\mu^3(1+\gamma/3)\log^2(d^{10})}{d^{3/2}\gamma^2}$ .

Next, we bound the error matrix and error matrix through  $\|err_4\|_2$  which is bounded by

applying Lemma 11, combined with Lemma 6 and the fact that  $\|\mathcal{E}_M \mathbf{v}_r\|_2 \leq \|\mathcal{E}_M\|$  since  $\|\mathbf{v}_r\|_2=1$  and by definition  $\|\mathcal{E}_M\| = \sup_{\|\mathbf{u}\|=1} \|\mathcal{E}_M \mathbf{u}\|_2$  yields,

$$\|\text{err}_4\|_2 \leq \frac{\lambda_r p(1+\gamma) \|\mathcal{E}_T\|_{<d+s>} + \omega_r \|\mathcal{E}_M\|_{<d+s>}}{\lambda_r^2 p(1-\gamma) + \omega_r^2}, \quad (3.73)$$

with probability  $1 - 2d^{-9}$  provided  $p \geq \frac{C\mu^4(1+\gamma/3)\log^2(d^{10})}{d^2\gamma^2}$ . Combining the error bounds results of  $\|\text{err}_1\|_2$ ,  $\|\text{err}_3\|_2$ ,  $\|\text{err}_2\|_2$ ,  $\|\text{err}_4\|_2$  in equations (3.66), (3.72), (3.68) and (3.73), lettings  $\|\mathbf{d}_u\|_2 = \epsilon_T$ , for  $\mathbf{u} \in \{\mathbf{a}_r, \mathbf{b}_r, \mathbf{c}_r\}$ ,  $\|\mathbf{d}_v\|_2 = \epsilon_M$ ,  $\Delta_{\lambda_r} = \frac{\epsilon_T}{\lambda_r^*}$  and  $\Delta_{\omega_r} = \frac{\epsilon_M}{\omega_r^*} \forall r \in [R]$  and using the fact that  $\lambda_r^* - \epsilon_T \leq \lambda_r \leq \lambda_r^* + \epsilon_T$  and  $\omega_r^* - \epsilon_T \leq \omega_r \leq \lambda_r^* + \epsilon_T$  for all  $r \in [R]$ , yields

$$\begin{aligned} & \|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2 \\ & \leq \frac{8pR\lambda_{\max}^*(\lambda_r^* + \epsilon_T) \max_{\mathbf{u} \in \{\mathbf{c}_m, \mathbf{b}_m, \mathbf{c}_r, \mathbf{b}_r\}} \left( \sqrt{1 - \frac{\epsilon_T}{2}} \epsilon_T, \left(\frac{c_0}{\sqrt{d}} + \epsilon_T\right), \left(\frac{c_0}{\sqrt{d}} + 3\epsilon_T\right)^2, \epsilon_T, \gamma, 1/\lambda_{\min}^* \right) \epsilon_T}{(\lambda_{\min}^* - \epsilon_T)^2 p(1-\gamma) + (\omega_{\min}^* - \epsilon_M)^2} \\ & + \frac{3R\omega_{\max}(\omega_r^* + \epsilon_M) \max \left( \epsilon_M, 1/\omega_{\min}^*, \frac{c_0}{\sqrt{d}} + 3\epsilon_M \right) \epsilon_M}{(\lambda_{\min}^* - \epsilon_T)^2 p(1-\gamma) + (\omega_{\min}^* - \epsilon_M)^2} \\ & + \frac{(\lambda_r^* + \epsilon_T)p(1+\gamma) \|\mathcal{E}_T\|_{<d+s>} + (\omega_r^* + \epsilon_T) \|\mathcal{E}_M\|_{<d+s>}}{(\lambda_{\min}^* - \epsilon_T)^2 p(1-\gamma) + (\omega_{\min}^* - \epsilon_M)^2}, \end{aligned} \quad (3.74)$$

with probability  $1 - 2d^{-9}$ . Simplifying the expression completes the proof for step 1 of the Lemma 4.

**Step2:** We now get an upper bound for  $\|\mathbf{a}_r - \dot{\mathbf{a}}_r\|_2$ . Note that

$$\|\mathbf{a}_r - \dot{\mathbf{a}}_r\|_2 = \left\| \frac{\dot{\mathbf{a}}_r}{\|\dot{\mathbf{a}}_r\|_2} - \dot{\mathbf{a}}_r \right\|_2 = \left\| \frac{\dot{\mathbf{a}}_r}{\|\dot{\mathbf{a}}_r\|_2} \right\|_2 |1 - \|\dot{\mathbf{a}}_r\|_2| = |1 - \|\dot{\mathbf{a}}_r\|_2|.$$

Hence bounding  $\|\mathbf{a}_r - \dot{\mathbf{a}}_r\|_2$  simplifies to bounding  $|1 - \|\dot{\mathbf{a}}_r\|_2|$ . Using the expression of  $\dot{\mathbf{a}}_r$  in (3.62) and applying the triangle inequality we get,

$$\begin{aligned} |1 - \|\dot{\mathbf{a}}_r\|_2| & \leq \underbrace{|1 - \|\lambda_r \mathbf{D}^{-1} \lambda_r^* \mathbf{E} \bar{\mathbf{a}}_r^* + \omega_r \mathbf{D}^{-1} \omega_r^* \mathbf{H} \bar{\mathbf{a}}_r^*\|_2|}_I + \underbrace{\|\lambda_r \mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\lambda_m^* \mathbf{F} \bar{\mathbf{a}}_m^* - \lambda_m \mathbf{G} \bar{\mathbf{a}}_m)\|_2}_{II} \\ & + \underbrace{\|\omega_r \mathbf{D}^{-1} \sum_{m \in [R] \setminus r} (\omega_m^* \mathbf{J} \bar{\mathbf{a}}_m^* - \omega_m \mathbf{P} \bar{\mathbf{a}}_m)\|_2}_{III} + \underbrace{\|\mathcal{E}_{T_F} \times_2 \mathbf{b}_r \times_3 \mathbf{c}_r + \mathcal{E}_{M_F} \mathbf{v}_r\|_2}_{IV}. \end{aligned} \quad (3.75)$$

Bounds for elements (II) (III) and (IV) in the equation above are derived in (3.72), (3.68) and (3.73) respectively. Hence we only focus on bounding elements (I).

$$\begin{aligned}
I &= \|\mathbf{a}_r^*\|_2 - \|\lambda_r \mathbf{D}^{-1} \lambda_r^* \mathbf{E} \bar{\mathbf{a}}_r^* + \omega_r \mathbf{D}^{-1} \omega_r^* \mathbf{H} \bar{\mathbf{a}}_r^*\|_2 \\
&\leq \|\mathbf{a}_r^* - \mathbf{D}^{-1} (\lambda_r \lambda_r^* \mathbf{E} + \omega_r \omega_r^* \mathbf{H}) \bar{\mathbf{a}}_r^*\|_2 \\
&= \|\mathbf{D}^{-1} (\lambda_r \lambda_r^* \mathbf{E} + \omega_r \omega_r^* \mathbf{H} - \mathbf{D} \mathbf{I}) \bar{\mathbf{a}}_r^*\|_2 \\
&= \|err_1\|_2,
\end{aligned} \tag{3.76}$$

where  $err_1$  is the error component defined in (3.63) and bounded in (3.66). The first equality is obtained by using the fact that  $\|\mathbf{a}_r^*\|_2 = 1$ , vector norm property is then use to get the first inequality and finally second equality is due to  $\mathbf{a}_r^* = \mathbf{D}^{-1} \mathbf{D} \mathbf{a}_r^*$  and the fact that  $\bar{\mathbf{a}}_r^* = \mathbf{a}_r^*$  since  $F_a = \text{supp}(\mathbf{a}_r^*) \subseteq \text{supp}(\bar{\mathbf{a}}_r^*) = F$ . Hence combining the results in equations (3.76) and (3.75) yields,

$$\begin{aligned}
\|\mathbf{a}_r - \dot{\mathbf{a}}_r\|_2 &\leq I + II + III + III + IV \\
&\leq \|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2,
\end{aligned} \tag{3.77}$$

which ends step 2 of the proof. The proof of Lemma 4 is completed by combining results of step 1 and step 2 which shows that  $\|\mathbf{a}_r - \mathbf{a}_r^*\|_2 \leq 2\|\dot{\mathbf{a}}_r - \mathbf{a}_r^*\|_2$ , and taking the maximum over all  $r$ .  $\square$

### 3.9 Auxillary Lemmas

**Lemma 5.** Fix  $r$  and let  $\mathbf{a}_r^{t+1}$  be obtained by the update on line (9) of Algorithm 1 and . Given the conditions in Theorem 3.4.1 hold, we get with probability  $1 - n^{-9}$  that  $\mathbf{a}^{t+1} \leq 3\mu/\sqrt{d}$ .



**Proof:** Let  $\tilde{\mathbf{a}}_r^{t+1}$  be the update on line (8) of Algorithm 1, then we can decompose its absolute valued in the following way.

$$\begin{aligned}
|\tilde{\mathbf{a}}^{t+1}(\mathbf{i})| \leq & \frac{1}{\mathbf{D}_{\text{ii}}} \left( \underbrace{\lambda_r^* |\mathbf{E}_{\text{ii}}| \frac{\mu}{\sqrt{n}}}_{:=\kappa_1} + \underbrace{\sum_{m \in [R] \setminus r} \lambda_m^* |\mathbf{F}_{\text{ii}}| \frac{\mu}{\sqrt{n}} + \sum_{m \in [R] \setminus r} \lambda_m |\mathbf{G}_{\text{ii}}| \frac{\mu}{\sqrt{n}}}_{:=\kappa_2} \right) + \\
& \frac{1}{\mathbf{D}_{\text{ii}}} \left( \underbrace{\omega_r^* |\mathbf{H}_{\text{ii}}| \frac{\mu}{\sqrt{n}}}_{:=\kappa_3} + \underbrace{\sum_{m \in [R] \setminus r} \omega_m^* |\mathbf{J}_{\text{ii}}| \frac{\mu}{\sqrt{n}} + \sum_{m \in [R] \setminus r} \omega_m |\mathbf{P}_{\text{ii}}| \frac{\mu}{\sqrt{n}}}_{:=\kappa_4} \right) + \\
& \frac{1}{\mathbf{D}_{\text{ii}}} \left( \underbrace{\left| \sum_{\mathbf{i}, \mathbf{j}, k} \mathcal{E}_T \delta_{\mathbf{i}, \mathbf{j}, k} \mathbf{b}_r(\mathbf{j}) \mathbf{c}_r(k) \right|}_{:=\kappa_5} + \underbrace{\left| \sum_{\mathbf{i}, l} \mathcal{E}_M \mathbf{v}_r(l) \right|}_{:=\kappa_6} \right), \tag{3.78}
\end{aligned}$$

Using the results of Lemmas 20-22 along with the decomposition of  $\mathbf{d}_b = \mathbf{b} - \mathbf{b}^*$  and  $\mathbf{d}_b = \mathbf{c} - \mathbf{c}^*$  and the incoherence condition on the tensor components we get the following bounds:

$$\begin{aligned}
\kappa_1 & \leq \lambda_r^* p \left( \frac{\mu}{\sqrt{n}} + \gamma \right); \quad \kappa_2 \leq (r-1) p \lambda_{\max}^* \left( 2 \frac{c_0^2}{n} + 19 \|d\|_{\max} + 2\gamma \right) \frac{\mu}{\sqrt{n}} \\
\kappa_3 & \leq \omega_r^* \frac{\mu}{\sqrt{n}}; \quad \kappa_4 \leq (r-1) \omega_{\max}^* \left( 2 \frac{c_0}{\sqrt{n}} + 4 \|d\|_{\max} + \right) \frac{\mu}{\sqrt{n}} \\
\kappa_5 & \leq n^2 \frac{\mu}{\sqrt{n}} |p \max(\mathcal{E}_T) + \gamma| \frac{\mu}{\sqrt{n}}; \quad \kappa_6 \leq n |\max(\mathcal{E}_M)| \frac{\mu}{\sqrt{n}}.
\end{aligned}$$

Using the initialization condition presented in Assumption 6 and letting  $d = \epsilon_0$  we get that

$$\frac{1}{\mathbf{D}_{\text{ii}}} \sum_r \kappa_r \leq \frac{\lambda_r^* p (1 + \gamma) + \omega_r^* \frac{\mu}{\sqrt{n}}}{p(1 - \gamma) + 1} \frac{\mu}{\sqrt{n}},$$

where  $\gamma = o(1)$ . It now remains to show that  $\kappa_5$  and  $\kappa_6$  are constants. To do so we use impose the following condition

$$\frac{|\max(\mathcal{E}_T)|}{\lambda_{\min}^*} \leq \frac{p(1 - \gamma) + 1}{n^{3/2} \mu} \quad \text{and} \quad \frac{|\max(\mathcal{E}_M)|}{\omega_{\min}^*} \leq \frac{p(1 - \gamma) + 1}{n}$$

Using the condition above and the expression of  $|\mathbf{a}^{t+1}|$  in (3.78) and noting that  $|\tilde{\mathbf{a}}^{t+1}(\mathbf{i})|\lambda_r^* \geq |\mathbf{a}^{t+1}(\mathbf{i})|$ ; which completes the proof of the lemma.  $\square$

**Lemma 6.** *Let  $\mathbf{u}$  and  $\mathbf{w}$  be unit vectors in  $\mathbb{R}^n$  such that  $|\mathbf{u}(\mathbf{i})| \leq \frac{\mu}{\sqrt{d}}$  and  $|\mathbf{w}(\mathbf{j})| \leq \frac{\beta}{\sqrt{d}}$ . Also let  $\delta_{\mathbf{i},\mathbf{j},k}$  be i.i.d. Bernoulli random variables with  $P(\delta_{\mathbf{i},\mathbf{j},k} = 1) = p$  and  $1 \leq \mathbf{i} \leq n$ ,  $1 \leq \mathbf{j} \leq n$ ,  $1 \leq k \leq n$ .*

*Then provided  $p \geq \frac{C\mu^2\beta^2(1+\gamma/3)\log(d^{10})}{d^2\gamma^2}$  we have*

$$\left| \sum_{\mathbf{j},k} \delta_{\mathbf{i},\mathbf{j},k} \mathbf{u}_r^2(\mathbf{j}) \mathbf{w}_r^2(k) \right| \leq p \langle \mathbf{u}, \mathbf{u} \rangle \langle \mathbf{w}, \mathbf{w} \rangle - p\gamma,$$

*with probability  $1 - d^{-10}$ .*

Proof: Let  $X_{jk} = \frac{1}{p} (\delta_{\mathbf{i},\mathbf{j},k} \mathbf{u}^2(\mathbf{j}) \mathbf{w}^2(k) - E(\delta_{\mathbf{i},\mathbf{j},k} \mathbf{u}^2(\mathbf{j}) \mathbf{w}^2(k)))$ . Using the bound on the elements of  $\mathbf{u}$  and  $\mathbf{w}$ , we have  $|X_{jk}| = |\frac{1}{p} (\delta_{\mathbf{i},\mathbf{j},k} - p) \mathbf{u}^2(\mathbf{j}) \mathbf{w}^2(k)| \leq \frac{\mu^2 \beta^2}{pd^2}$ . Also

$$\sum_{\mathbf{j},k} E[X_{jk}^2] = \frac{1}{p} (1-p) \sum_{\mathbf{j},k} \mathbf{u}_r^4(\mathbf{j}) \mathbf{w}_r^4(k) \leq \frac{\mu^2 \beta^2}{pd^2}.$$

Applying Bernstein tail bound inequality we get:

$$P \left( \left| \sum_{\mathbf{j},k} \delta_{\mathbf{i},\mathbf{j},k} \mathbf{u}_r^2(\mathbf{j}) \mathbf{w}_r^2(k) - p \langle \mathbf{u}, \mathbf{u} \rangle \langle \mathbf{w}, \mathbf{w} \rangle \right| \geq pt \right) \leq \exp \left( \frac{-d^2 pt^2 / 2}{\mu^2 \beta^2 (1 + \frac{1}{3}t)} \right).$$

Setting the right side of the inequality to be less than  $q$  yields:

$$P \left( \left| \sum_{\mathbf{j},k} \delta_{\mathbf{i},\mathbf{j},k} \mathbf{u}_r^2(\mathbf{j}) \mathbf{w}_r^2(k) \right| \leq p \langle \mathbf{u}, \mathbf{u} \rangle \langle \mathbf{w}, \mathbf{w} \rangle - p\gamma \right) \geq 1 - q,$$

for  $p \geq \frac{\mu^2 \beta^2 (1+\gamma/3) \log(1/q)}{d^2 \gamma^2}$ . Choosing  $q \leq d^{-10}$  completes the proof of Lemma 6.  $\square$

**Lemma 7.** *Let  $\mathbf{u}^*$ ,  $\mathbf{u}$  and  $\mathbf{w}$  be unit vectors in  $\mathbb{R}^n$  such that  $|\mathbf{u}_1^*| \leq \frac{\mu}{\sqrt{d}}$ ,  $|\mathbf{u}|$  and  $|\mathbf{w}| \leq \frac{\beta}{\sqrt{d}}$ . Let  $\mathbf{d}$  be another vector with  $\|\mathbf{d}\|_2 \leq 1$ . Also let  $\delta_{\mathbf{i},\mathbf{j},k}$  be i.i.d. Bernoulli random variables with*

$P(\delta_{ijk} = 1) = p$  and  $1 \leq i \leq n, 1 \leq j \leq n, 1 \leq k \leq n$ . Provided  $p \geq \frac{C\mu\beta^2(1+\gamma/3)\log^2(\frac{1}{2}d^{10})}{d^{3/2}\gamma^2}$ , with probability greater than  $1 - 2d^{-10}$ , we have

$$\left| \sum_{j,k} \delta_{ijk} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - p \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{d}, \mathbf{w} \rangle \right| \leq p\gamma \|\mathbf{d}\|_2.$$

Proof: Let  $X_{jk} = \frac{1}{p} (\delta_{ijk} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - E(\delta_{ijk} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k)))$ . Then we have That is  $|X_{jk}| = \frac{1}{p} (\delta_{ijk} - p) \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) \leq \frac{1}{p} (1 - p) \frac{\mu\beta^2}{d^{3/2}} \|\mathbf{d}\|_2$ . Also,

$$\sum_{j,k} E[X_{jk}^2] = \frac{1}{p} \sum_{j,k} (\mathbf{u}(j)^2 \mathbf{d}(k)^2 \mathbf{u}(j)^2 \mathbf{w}(k)^2) \leq \frac{\mu\beta^2 \|\mathbf{d}\|_2^2}{pd^{3/2}}.$$

Applying Bernstein tail bound inequality we get:

$$P \left( \left| \sum_{j,k} \delta_{ijk} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - p \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{d}, \mathbf{w} \rangle \right| \geq pt \right) \leq 2 \exp \left( \frac{-d^{3/2} p t^2}{\mu\beta^2 \|\mathbf{d}\|_2 (\|\mathbf{d}\|_2 + \frac{1}{3}t)} \right). \quad (3.79)$$

Setting the right side of the inequality to be less than  $q$  and choosing  $t \leq \gamma \|\mathbf{d}\|_2$  then solving for  $p$  yields:

$$P \left( \left| \sum_{j,k} \delta_{ijk} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - p \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{d}, \mathbf{w} \rangle \right| \leq p\gamma \|\mathbf{d}\|_2 \right) \geq 1 - 2q,$$

for  $p \geq \frac{\mu\beta^2(1+\gamma/3)\log(\frac{1}{q})}{d^{3/2}\gamma^2}$ . Choosing  $q \leq d^{-10}$  completes the proof of Lemma 7.  $\square$

**Lemma 8.** Let  $\mathbf{u}^*, \mathbf{w}^*, \mathbf{u}$  and  $\mathbf{w}$  be unit vectors in  $\mathbb{R}^n$  such that  $|\mathbf{u}^*(i)|$  and  $|\mathbf{w}^*(j)| \leq \frac{\mu}{\sqrt{d}}$ ,  $|\mathbf{u}_i|$  and  $|\mathbf{w}_i| \leq \frac{\beta}{\sqrt{d}}$ . Let  $\delta_{i,j,k}$  be i.i.d. Bernoulli random variables with  $P(\delta_{ijk} = 1) = p$  and  $1 \leq i, j, k \leq n$ . Provided  $p \geq \frac{C\mu^2\beta^2(1+\gamma/3)\log(\frac{1}{2}d^{10})}{d^2\gamma^2}$ , with probability greater than  $1 - 2d^{-10}$ , we have

$$\left| \sum_{j,k} \delta_{ijk} \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k) \right| \leq p |\langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{w}^*, \mathbf{w} \rangle| + p\gamma.$$

Proof: Let  $X_{jk} = \frac{1}{p} (\delta_{ijk} \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k) - E(\delta_{ijk} \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k)))$ . Then we have  $|X_{jk}| = \frac{1}{p} (\delta_{ijk} - p) \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k) \leq \frac{1}{p} (1 - p) \frac{\mu^2 \beta^2}{d^2}$ . Also

$$\sum_{j,k} E[X_{jk}^2] = \frac{1}{p} (1 - p) \sum_{j,k} (\mathbf{u}(j)^2 \mathbf{w}(k)^2) \leq \frac{1}{p} (1 - p) \frac{\mu^2 \beta^2}{d^2}.$$

Applying Bernstein tail bound inequality we get:

$$P \left( \left| \sum_{j,k} \delta_{ijk} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - p \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{d}, \mathbf{w} \rangle \right| \geq pt \right) \leq 2 \exp \left( \frac{-d^2 p t^2}{\mu^2 \beta^2 (1 - p) (1 + \frac{1}{3} t)} \right).$$

Setting the right side of the inequality to be less than  $q$  and choosing  $t \leq \gamma$  then solving for  $p$  yields:

$$P \left( \left| \sum_{j,k} \delta_{ijk} \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k) - \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{w}^*, \mathbf{w} \rangle \right| \leq p \gamma \right) \geq 1 - 2q,$$

and  $p \geq \frac{\mu^2 \beta^2 (1 + \gamma/3) \log(\frac{1}{q})}{d^2 \gamma^2}$ . Letting  $q \leq d^{-10}$  completes the proof of Lemma 8.  $\square$

**Lemma 9.** Let  $\lambda_r$  be the update of the  $r^{th}$  weight of the tensor after one iteration of Algorithm 1 and let  $\lambda_r^*$  be the true  $r^{th}$  weight of the tensor decomposition in the dense tensor and dense matrix case. Let  $\widetilde{\mathbf{c}}$  be as defined in (3.22) and  $\mathbf{c}$  as defined in (3.20) then with probability greater than  $1 - 2n^{-9}$  we have

$$|\lambda_r - \lambda_r^*| \leq \|\widetilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2.$$

Proof: We know that  $\|\mathbf{c}_r^*\|_2 = \|\mathbf{c}_r\|_2 = 1$  hence we can write,

$$\begin{aligned} |\lambda_r - \lambda_r^*| &= \left| \|\lambda_r \mathbf{c}_r\|_2 - \|\lambda_r^* \mathbf{c}_r^*\|_2 \right| \\ &\leq \|\lambda_r \mathbf{c}_r - \lambda_r^* \mathbf{c}_r^*\|_2 \\ &= \|\widetilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2 \end{aligned}$$

The last equality above is obtained by observing that  $\widetilde{\mathbf{c}}_r = \lambda_r \mathbf{c}_r$  as shown in the proof of Lemma 1. This complete the proof of the Lemma. Notice that the above Lemma can also be

applied on  $\omega_r$  to obtain  $|\omega_r - \omega_r^*| \leq \|\widetilde{\mathbf{v}}_r - \omega_r^* \mathbf{v}_r^*\|_2$ .  $\square$

**Lemma 10.** *Let  $\widetilde{\mathbf{c}}$  be as defined in (3.22) and  $\mathbf{c}$  as defined in (3.20). Also let  $\lambda_r$  be the update of the  $r^{\text{th}}$  weight of the tensor after one iteration of Algorithm 1 and let  $\lambda_r^*$  be the true  $r^{\text{th}}$  weight of the tensor decomposition in the dense tensor and dense matrix case. Then with probability greater than  $1 - 2n^{-9}$  we have*

$$\begin{aligned} \|\mathbf{c}_r - \mathbf{c}_r^*\|_2 &\leq \frac{2}{\lambda_r^*} \|\widetilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2, \\ \|\mathbf{c}_r - \mathbf{c}_r^*\|_2 + \Delta_{\lambda_r} &\leq \frac{3}{\lambda_r^*} \|\widetilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2, \end{aligned}$$

where  $\Delta_{\lambda_r}$  is as defined in (3.18).

Proof:

$$\begin{aligned} \lambda_r^* \|\mathbf{c}_r - \mathbf{c}_r^*\|_2 &= \|\lambda_r^* \mathbf{c}_r - \lambda_r^* \mathbf{c}_r^*\|_2 \\ &= \|\lambda_r \mathbf{c}_r - \lambda_r^* \mathbf{c}_r^* - \epsilon_{\lambda_r} \mathbf{c}_r\|_2 \leq \|\lambda_r \mathbf{c}_r - \lambda_r^* \mathbf{c}_r^*\|_2 + \|\epsilon_{\lambda_r} \mathbf{c}_r\|_2 \\ &= \|\widetilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2 + |\lambda_r - \lambda_r^*| \\ &\leq 2\|\widetilde{\mathbf{c}}_r - \lambda_r^* \mathbf{c}_r^*\|_2, \end{aligned} \tag{3.80}$$

which proves the first inequality of the Lemma. The proof of the second inequality in the lemma is obtained by combining (3.80) with the results of Lemma 9.  $\square$

**Lemma 11.** *For any tensor  $\mathcal{E}_T \in \mathbb{R}^{n \times n \times n}$  and any vectors  $\mathbf{u}$  and  $\mathbf{v} \in \mathbb{R}^n$  with*

$\|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = 1$ , *we have*

$$\|\mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v}\|_2 \leq \|\mathcal{E}_T\|,$$

where  $\|\mathcal{E}_T\|$  represents the spectral norm of the tensor defined in (2.1).

Proof:

$$\begin{aligned}
\|\mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v}\|_2 &= \frac{\|\mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v}\|_2^2}{\|\mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v}\|_2} \\
&= \left| \mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v} \times_3 \left( \frac{\mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v}}{\|\mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v}\|_2} \right) \right| \\
&\geq \sup_{\|\mathbf{u}\|=\|\mathbf{v}\|=\|\mathbf{w}\|=1} \left| \mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v} \times_3 \mathbf{w} \right| \\
&= \|\mathcal{E}_T\|.
\end{aligned}$$

The first inequality is due to  $\|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = 1$  and the fact that  $\frac{\mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v}}{\|\mathcal{E}_T \times_1 \mathbf{u} \times_2 \mathbf{v}\|_2} = 1$ . The last equality is obtained by applying the definition of the tensor spectral norm provided in (2.1).  $\square$

**Lemma 12.** *Let  $\mathbf{u}$  and  $\mathbf{w}$  be unit vectors and let  $\mathbf{d}$  be a vector such that  $\mathbf{d} = \mathbf{u} - \mathbf{w}$  then*

$$|\langle \mathbf{w}, \mathbf{d} \rangle| = \frac{1}{2} \|\mathbf{d}\|_2^2.$$

Proof: Note that  $\|\mathbf{u}\|_2^2 = \sum (\mathbf{w}(i) + \mathbf{d}(i))^2$ . Hence given that  $\mathbf{u}$  is a unit vector we get

$$\begin{aligned}
\sum \mathbf{w}(i)^2 + 2 \sum \mathbf{w}(i) \mathbf{d}(i) + \sum \mathbf{d}(i)^2 &= 1 \\
2 \sum \mathbf{w}(i) \mathbf{d}(i) + \sum \mathbf{d}(i)^2 &= 0 \\
2 \sum \mathbf{w}(i) \mathbf{d}(i) &= - \sum \mathbf{d}(i)^2 \\
|\langle \mathbf{w}, \mathbf{d} \rangle| &= \frac{1}{2} \|\mathbf{d}\|_2^2,
\end{aligned}$$

Which completes the proof of the lemma.  $\square$

**Lemma 13.** *Let  $\mathbf{u}$  and  $\mathbf{w}$  be unit vectors define  $F_1 := \text{supp}(\mathbf{u})$ ,  $F_2 := \text{supp}(\mathbf{w})$  be the support sets for  $\mathbf{u}$  and  $\mathbf{w}$  respectively with  $F_i \subseteq \{1, \dots, d\}$  and  $F := F_u \cup F_w$  be the union of the two vectors' support sets. Let  $\bar{\mathbf{u}} := \text{Truncate}(\mathbf{u}, F)$  then it follows that*

$$\langle \bar{\mathbf{u}}, \mathbf{w} \rangle = \langle \mathbf{u}, \mathbf{w} \rangle.$$

Proof: Since by definition,  $\bar{\mathbf{u}} := \text{Truncate}(\mathbf{u}, F)$ , then we can write  $\langle \bar{\mathbf{u}}, \mathbf{w} \rangle$  explicitly as  $\langle \bar{\mathbf{u}}, \mathbf{w} \rangle = \sum_{i \in [d]} \bar{\mathbf{u}}(i) \mathbf{w}(i)$ . Since  $\bar{\mathbf{u}}(i) \neq 0$  only when  $i \in F_1$  and  $i \in F_2$ , we get  $\sum_{i \in [d]} \bar{\mathbf{u}}(i) \mathbf{w}(i) = \sum_{i \in F} \mathbf{u}(i) \mathbf{w}(i)$ . However, we know that  $\text{supp}(\mathbf{w}) = F_2 \subseteq F$  hence we get

$$\langle \bar{\mathbf{u}}, \mathbf{w} \rangle = \sum_{i \in F} \mathbf{u}(i) \mathbf{w}(i) = \sum_{i \in [d]} \mathbf{u}(i) \mathbf{w}(i) = \langle \mathbf{u}, \mathbf{w} \rangle.$$

□

## 4. UNCERTAINTY QUANTIFICATION IN COVARIATE ASSISTED TENSOR COMPLETION

In Chapter 3 we proposed **COSTCO**, an algorithm which aims to complete a sparse and highly-missing tensor in the presence of covariate information along at least one tensor mode. Using a low-rank assumption on both the tensor and a single covariate matrix, the methodology in Chapter 3 assumes that the latent components corresponding to the coupled modes are shared by both the tensor and matrix’s decomposition therefore leveraging the additional covariate information to improve the accuracy of the recovered tensor. The parameter estimation in **COSTCO** is formulated as a non-convex optimization with sparsity constraints, and employs a sparse alternating least-squares approach to recover tensor components. It was shown that **COSTCO** allows for a relaxation in the bound of the tensor entries reveal probability and enjoys better recovery accuracy compared to stand alone tensor completion methods.

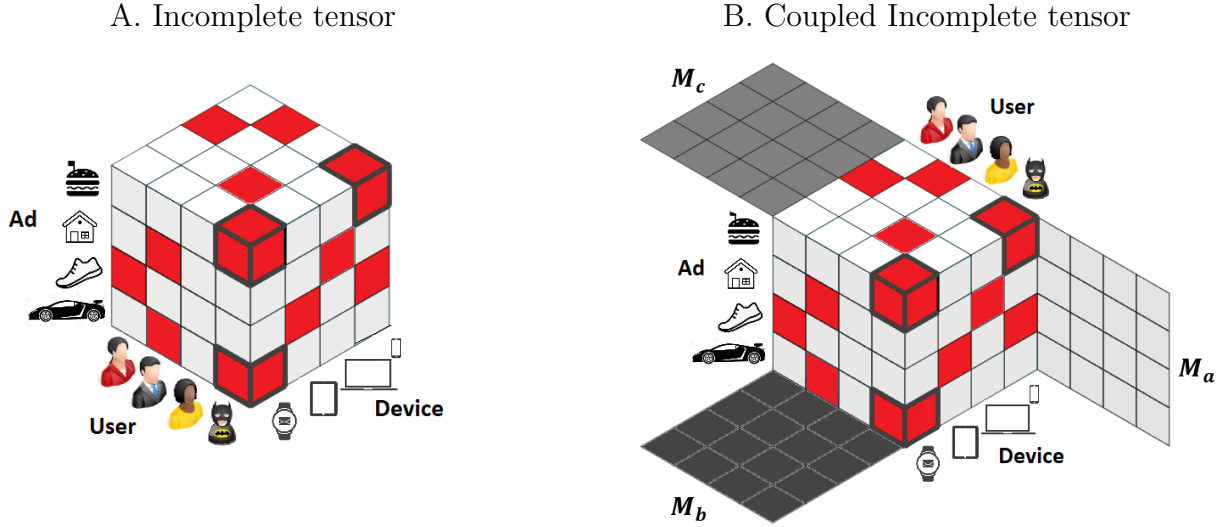
In this chapter we focus on building tools to facilitate inference for the recovered tensor components. In fact, recovery of the tensor components alone does not suffice to enable practical use of the recovered tensor in real problems. A need to assess the trustworthiness of these predictions arises if completion methods are to be used in practice. Due to the non convexity of most tensor completion problems, this constitutes a challenge, with very few methods providing theoretical guarantees. We therefore aims to: (1) characterize the distributional property of the recovered tensor components, (2) propose a reliable risk assessment method for the recovered components through the building of confidence intervals and (3) demonstrate that the inclusion of side information in the tensor completion model leads to shorter confidence intervals compared to those obtained from stand alone tensor completion methods.

In Chapter 3 the methodology for **COSTCO** is formulated for an arbitrary rank, sparse tensor and matrix. In this chapter, we focus on the rank one 3-way tensor case with covariate matrices coupled alongside all three modes of the tensor (see Figure 4.1). Although this work involves a much simplified tensor structure than that provided in Chapter 3, to our knowledge this is the first work that aims to characterize the distribution and uncertainty quantification



for tensor factors recovered using a tensor completion method which incorporates covariate information.

The rest of the chapter is organized as follows. In Section 4.1 we revisit the model and the optimization problem of **COSTCO** and present it in the special case of the rank one non-sparse tensor with matrices coupled along all three modes of the tensor. Section 4.2 presents the main theoretical results. Section 4.3 contains a series of simulation studies. All proof details are provided in Section 4.4.



**Figure 4.1.** A. Order-3 (user  $\times$  ad  $\times$  device) tensor with missing entries; B. Order-3 tensor with missing entries coupled with matrices of ad, user and device covariates  $M_a$ ,  $M_b$ ,  $M_c$  respectively. The red cells represent missing entries; grey and white cells represent non-zero entries.

## 4.1 Methodology

In this section we revisit the non-convex optimization method for parameter estimation in **COSTCO** for the case of the rank one non-sparse tensor with all three modes coupled to covariate matrices. For ease of notation we also assume that the tensor and matrix components are of equal dimension.

#### 4.1.1 Model and Algorithm

Let  $\mathcal{T} \in \mathbb{R}^{n \times n \times n}$  and  $\mathbf{M}_a \in \mathbb{R}^{n \times n}$ ,  $\mathbf{M}_b \in \mathbb{R}^{n \times n}$ ,  $\mathbf{M}_c \in \mathbb{R}^{n \times n}$  be the observed third-order tensor and covariate matrices corresponding to the feature information along the three modes of the tensor  $\mathcal{T}$ . Let  $\Omega$  be the subset of indexes of the tensor  $\mathcal{T}$  for which entries are not missing. Throughout this chapter, we shall let

$$\delta_{i,j,k} := \mathbb{I}\{(i, j, k) \in \Omega\}, \text{ for } 1 \leq i, j, k \leq n.$$

We assume a noisy observation model, where the observed tensor and matrices are noisy versions of their true counterparts. That is,

$$P_\Omega(\mathcal{T}) = P_\Omega(\mathcal{T}^* + \mathcal{E}_T); \quad \mathbf{M}_a = \mathbf{M}_a^* + \mathcal{E}_{Ma}; \quad \mathbf{M}_b = \mathbf{M}_b^* + \mathcal{E}_{Mb}; \quad \mathbf{M}_c = \mathbf{M}_c^* + \mathcal{E}_{Mc}, \quad (4.1)$$

where  $\mathcal{E}_T$ ,  $\mathcal{E}_{Ma}$ ,  $\mathcal{E}_{Mb}$  and  $\mathcal{E}_{Mc}$  are the error tensor and the error matrices respectively;  $\mathcal{T}^*$ ,  $\mathbf{M}_a^*$ ,  $\mathbf{M}_b^*$  and  $\mathbf{M}_c^*$  are the true tensor and the true matrices, which are assumed to have each a rank one CP decomposition structure [29] represented as,

$$\mathcal{T}^* = \lambda^* \mathbf{a}^* \otimes \mathbf{b}^* \otimes \mathbf{c}^*; \quad \mathbf{M}_a^* = \omega_a^* \mathbf{a}^* \otimes \mathbf{v}_a^* \quad (4.2)$$

$$\mathbf{M}_b^* = \omega_b^* \mathbf{b}^* \otimes \mathbf{v}_b^*; \quad \mathbf{M}_c^* = \omega_c^* \mathbf{c}^* \otimes \mathbf{v}_c^*, \quad (4.3)$$

where  $\lambda^*, \omega_a^*, \omega_b^*$  and  $\omega_c^* \in \mathbb{R}^+$ , and  $\mathbf{a}^* \in \mathbb{R}^n$ ,  $\mathbf{b}^* \in \mathbb{R}^n$ ,  $\mathbf{c}^* \in \mathbb{R}^n$ ,  $\mathbf{v}_a^* \in \mathbb{R}^n$ ,  $\mathbf{v}_b^* \in \mathbb{R}^n$  and  $\mathbf{v}_c^* \in \mathbb{R}^n$  with  $\|\mathbf{a}^*\|_2 = \|\mathbf{b}^*\|_2 = \|\mathbf{c}^*\|_2 = \|\mathbf{v}_a^*\|_2 = \|\mathbf{v}_b^*\|_2 = \|\mathbf{v}_c^*\|_2 = 1$ .

Given a rank one tensor  $\mathcal{T}$  with missing entries and covariate matrices  $\mathbf{M}_a$ ,  $\mathbf{M}_b$  and  $\mathbf{M}_c$  COSTCO recovers the true tensor  $\mathcal{T}^*$  and its latent components using the following model formulation.

$$\begin{aligned} \min_{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{v}_a, \mathbf{v}_b, \mathbf{v}_c} & \left\{ \|P_\Omega(\mathcal{T}) - P_\Omega(\lambda \mathbf{a} \otimes \mathbf{b} \otimes \mathbf{c})\|_F^2 + \|\mathbf{M}_a - \omega_a \mathbf{a} \otimes \mathbf{v}_a\|_F^2 \right\} + \\ & \left\{ \|\mathbf{M}_b - \omega_b \mathbf{b} \otimes \mathbf{v}_b\|_F^2 + \|\mathbf{M}_c - \omega_c \mathbf{c} \otimes \mathbf{v}_c\|_F^2 \right\} \\ \text{subject to } & \|\mathbf{a}\|_2 = \|\mathbf{b}\|_2 = \|\mathbf{c}\|_2 = \|\mathbf{v}_a\|_2 = \|\mathbf{v}_b\|_2 = \|\mathbf{v}_c\|_2 = 1. \end{aligned} \quad (4.4)$$

As shown in Chapter 3, the problem in (4.4) is a non-convex optimization when considering all parameters at once. However the objective function is convex in each parameter, given the other parameters are kept fixed.

The algorithm **COSTCO** is an alternative minimization procedure, which at each step fixes all but one of the tensor components to be estimated. In its general form, **COSTCO** enforces sparsity in the model via a truncation step. However, because in this chapter we only focus on the dense case, we present the **COSTCO** algorithm without that truncation step in Algorithm 2. We also adjust the Algorithm 1 to reflect the fact that all three modes of the tensor are coupled instead of one mode as presented in Chapter 3.

---

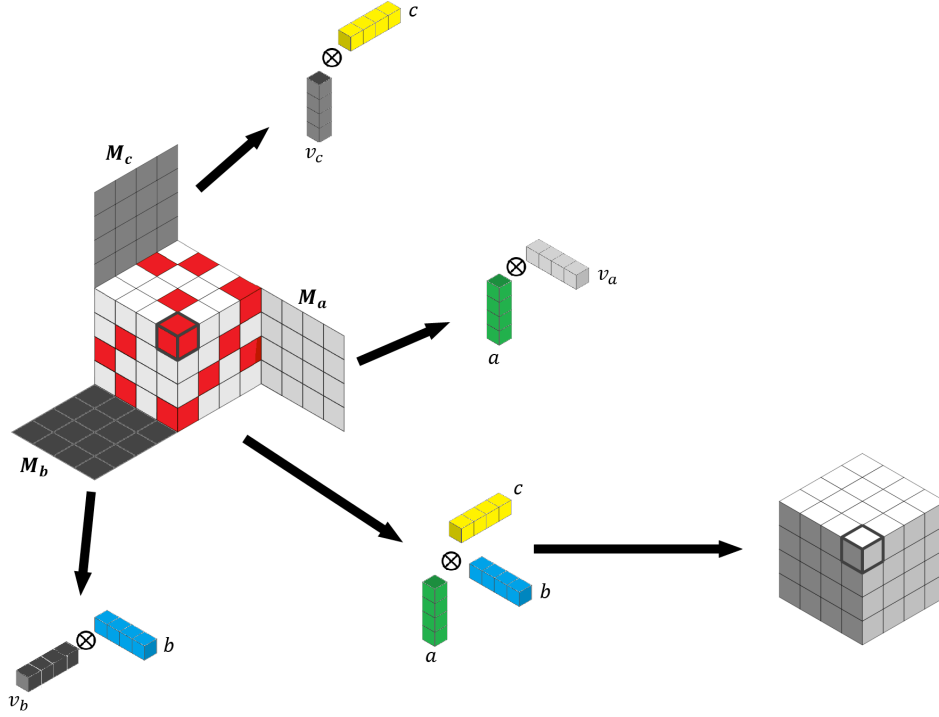
**Algorithm 2** **COSTCO: Covariate-assisted Tensor Completion (No truncation)**

---

- 1: **Input:** Observed tensor  $P_\Omega(\mathcal{T}) \in \mathbb{R}^{n \times n \times n}$ , observed matrix  $\mathbf{M}_a \in \mathbb{R}^{n \times n_{va}}$ ,  $\mathbf{M}_b \in \mathbb{R}^{n \times n_{vb}}$ ,  $\mathbf{M}_c \in \mathbb{R}^{n \times n_{vc}}$  maximal number of iterations  $\tau$ , tolerance  $tol$ .
  - 2: Initialize  $\lambda, (\mathbf{a}, \mathbf{b}, \mathbf{c}), (\omega_a, \omega_b, \omega_c), (\mathbf{v}_a, \mathbf{v}_b, \mathbf{v}_c)$ .
  - 3: **While**  $t \leq \tau$  and  $\left( \frac{\|\mathbf{a}_{old} - \mathbf{a}\|_2}{\|\mathbf{a}_{old}\|_2} + \frac{\|\mathbf{b}_{old} - \mathbf{b}\|_2}{\|\mathbf{b}_{old}\|_2} + \frac{\|\mathbf{c}_{old} - \mathbf{c}\|_2}{\|\mathbf{c}_{old}\|_2} \right) \geq tol$ ,
  - 4:    $\mathbf{a}_{old} \leftarrow \mathbf{a}, \quad \mathbf{b}_{old} \leftarrow \mathbf{b}, \quad \mathbf{c}_{old} \leftarrow \mathbf{c}, \quad (\mathbf{v}_a)_{old} \leftarrow \mathbf{v}_a, \quad (\mathbf{v}_b)_{old} \leftarrow \mathbf{v}_b, \quad (\mathbf{v}_c)_{old} \leftarrow \mathbf{v}_c$
  - 5:    $\mathbf{a} \leftarrow \frac{\lambda P_\Omega(\mathcal{T})(\mathbf{I}, \mathbf{b}, \mathbf{c}) + \omega_a \mathbf{M}_a \mathbf{v}_a}{\lambda^2 P_\Omega(\mathbf{I}, \mathbf{b}^2, \mathbf{c}^2) + \omega_a^2}, \quad \mathbf{a} \leftarrow \mathbf{a} / \|\mathbf{a}\|_2$
  - 6:    $\mathbf{b} \leftarrow \frac{\lambda P_\Omega(\mathcal{T})(\mathbf{a}, \mathbf{I}, \mathbf{c}) + \omega_b \mathbf{M}_b \mathbf{v}_b}{\lambda^2 P_\Omega(\mathbf{a}^2, \mathbf{I}, \mathbf{c}^2) + \omega_b^2}, \quad \mathbf{b} \leftarrow \mathbf{b} / \|\mathbf{b}\|_2$
  - 7:    $\mathbf{c} \leftarrow \frac{\lambda P_\Omega(\mathcal{T})(\mathbf{a}, \mathbf{b}, \mathbf{I}) + \omega_c \mathbf{M}_c \mathbf{v}_c}{\lambda^2 P_\Omega(\mathbf{a}^2, \mathbf{b}^2, \mathbf{I}) + \omega_c^2}, \quad \mathbf{c} \leftarrow \mathbf{c} / \|\mathbf{c}\|_2$
  - 8:    $\mathbf{v}_a \leftarrow \mathbf{M}_a^\top \mathbf{a}, \quad \mathbf{v}_b \leftarrow \mathbf{M}_b^\top \mathbf{b}, \quad \mathbf{v}_c \leftarrow \mathbf{M}_c^\top \mathbf{c}$
  - 9:    $\lambda \leftarrow \frac{P_\Omega(\mathcal{T})(\mathbf{a}, \mathbf{b}, \mathbf{c})}{P_\Omega(\mathbf{a}^2, \mathbf{b}^2, \mathbf{c}^2)}, \quad \omega_a \leftarrow \|\mathbf{v}_a\|_2, \quad \omega_b \leftarrow \|\mathbf{v}_b\|_2, \quad \omega_c \leftarrow \|\mathbf{v}_c\|_2$
  - 10:    $\mathbf{v}_a \leftarrow \mathbf{v}_a / \|\mathbf{v}_a\|_2, \quad \mathbf{v}_b \leftarrow \mathbf{v}_b / \|\mathbf{v}_b\|_2, \quad \mathbf{v}_c \leftarrow \mathbf{v}_c / \|\mathbf{v}_c\|_2$
  - 11: **End While**
- 

**Initialization:** In Chapter 3 we argued that there could be multiple local optima due to the non-convexity of the optimization problem and propose a procedure to initialize the tensor and matrix components. The procedure uses SVD decomposition as the initialization method for the shared tensor components and the robust tensor power method proposed in Anandkumar, Ge, Hsu, *et al.* [57] for the non-shared tensor components. Since we assume in this chapter that all three modes of the tensor are coupled to a covariate matrix, we apply the SVD initialization on all tensor and matrix components. This new initialization procedure forces us to redefine the formula for updating the tensor weight  $\lambda$ . In fact, in Algorithm 1,  $\lambda$  was defined as the 2-norm of the un-normalized and non-coupled tensor

component **b**. It is worth noting that the norm of the coupled component would not only contain the tensor weight but also the matrix weight, therefore making it an inadequate estimate for the tensor weight. Since we now focus on the case in which all three modes of the tensor are coupled, we redefine the update of  $\lambda$  to the formula provided on line 9 of the Algorithm 2. This new update method produces an equivalent error bound for  $\lambda$  compared to that obtained when defining  $\lambda$  as the 2-norm of the un-normalized non-coupled tensor component.



**Figure 4.2.** Illustration of COSTCO showing recovery procedure for missing entries through joint decomposition of a rank 1 tensor and rank 1 matrices; red cells represent missing entries. The tensor and matrices  $M_a$ ,  $M_b$  and  $M_c$  are coupled along mode 1, mode 2 and mode 3 respectively. The components **a**, **b** and **c** are shared by the tensor and matrices  $M_a$ ,  $M_b$  and  $M_c$  decomposition respectively.

Figure 4.2 illustrates COSTCO with a rank one decomposition and when all three modes of the tensor are coupled each to a covariate matrix. It reveals how COSTCO, leverages the additional latent information coming from the matrices of covariates on the shared modes. In

the general rank and sparse case discussed in Chapter 3, a tuning procedure with BIC-type criterion is employed in order to estimate the rank and sparsity parameters required in the initialization of the algorithm. However, since we focus on the non-sparse and rank one case, there is no more need for such tuning procedures in Algorithm 2.

## 4.2 Theoretical Analysis

In this section, we present the distributional theory for the recovered estimates of the tensor factors using `COSTCO`. We also demonstrate how to conduct data-driven uncertainty quantification for the affordmentioned estimates through the construction of confidence intervals. We start with presenting a set of assumptions required for the main theoretical results.

### 4.2.1 Assumptions

The theoretical analysis is built on the following assumptions:

**Assumption 1** (Tensor and matrix structure):

Assume  $\mathcal{T}^*$ ,  $\mathbf{M}_a^*$ ,  $\mathbf{M}_b^*$ ,  $\mathbf{M}_c^*$  are rank one tensors and matrices respectively coupled along the modes of the tensor and the entries of their decomposed components respect the  $\mu$ -mass condition,

$$\max\{\|\mathbf{a}^*\|_\infty, \|\mathbf{b}^*\|_\infty, \|\mathbf{c}^*\|_\infty, \|\mathbf{v}_a^*\|_\infty, \|\mathbf{v}_b^*\|_\infty, \|\mathbf{v}_c^*\|_\infty\} \leq \frac{\mu}{\sqrt{n}},$$

where  $\mu$  is a fixed constant.

Assumption 1 here is a special case of Assumption 1 from Chapter 3. Notice, the fact that the incoherence assumption, which was necessary to guarantee soft orthogonality of the tensor and matrix factors is no longer required since we are working on the special case of the rank one tensor and matrices. The  $\mu$ -mass condition is still required for this special case as there is still a need to ensure that the mass of the tensor and the matrices are evenly distributed and not centered around just a couple of entries.

**Assumption 2:** (Tensor and matrix noise)

Assume  $\{\mathcal{E}_{i,j,k}\}_{1 \leq i,j,k \leq n}$ ,  $\{(\mathcal{E}_{Ma})_{i,l}\}_{1 \leq i,l \leq n}$

$\{(\mathcal{E}_{Mb})_{j,l}\}_{1 \leq j,l \leq n}$ ,  $\{(\mathcal{E}_c)_{k,l}\}_{1 \leq k,l \leq n}$  are independent sub-Gaussian random variables satisfying:

$$\begin{aligned} \mathbb{E}[\mathcal{E}_{i,j,k}] &= 0, \quad \text{Var}(\mathcal{E}_{i,j,k}) = (\sigma_T^2)_{i,j,k}; & \mathbb{E}[(\mathcal{E}_{Ma})_{i,l}] &= 0, \quad \text{Var}((\mathcal{E}_{Ma})_{i,l}) = (\sigma_{Ma}^2)_{i,l}; \\ \mathbb{E}[(\mathcal{E}_{Mb})_{j,l}] &= 0, \quad \text{Var}((\mathcal{E}_{Mb})_{j,l}) = (\sigma_{Mb}^2)_{j,l}; & \mathbb{E}[(\mathcal{E}_{Mc})_{k,l}] &= 0, \quad \text{Var}((\mathcal{E}_{Mc})_{k,l}) = (\sigma_{Mc}^2)_{k,l}. \end{aligned}$$

Denote the maximum and minimum variance for the noise tensor and matrices as follows:

$$\begin{aligned} \sigma_{max}^2 &:= \max_{1 \leq i,j,k \leq n} (\sigma_T^2)_{i,j,k} \quad \text{and} \quad \sigma_{min}^2 := \min_{1 \leq i,j,k \leq n} \sigma_{i,j,k}^2, \quad \text{also let} \\ (\sigma_{Mu}^2)_{max} &:= \max_{1 \leq i,l \leq n} (\sigma_{Mu}^2)_{i,l} \quad \text{and} \quad (\sigma_{Mu}^2)_{min} := \min_{1 \leq i,l \leq n} (\sigma_{Mu}^2)_{i,l} \quad \text{for } \mathbf{u} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}. \end{aligned}$$

Then we also assume that  $\sigma_{max}^2/\sigma_{min}^2 = O(1)$  and  $(\sigma_{Mu}^2)_{max}/(\sigma_{Mu}^2)_{min} = O(1)$ , for  $\mathbf{u} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$ .

Assumption 2 provides specification for the noise tensor and noise matrices. The key elements for this assumption are the independence condition coupled with the possibility for non-equal variance. In fact, the noise tensor and noise matrix entries are required to be independent not only to each other but also to the true tensor entries. This specification is important as it will be exploited later for decoupling the true tensor components from the associated noise tensor and matrix noise entries.

Another important feature of Assumption 2 is the the fact that it allows for heteroskedasticity in both the noise tensor and matrices. That is, the variance of each entry in the noise tensor and noise matrices are allowed to be different. This represents a key advantage for the theory derived in this chapter since, in practice, data often exhibit non equal variance properties. Lastly the assumption that the largest variance and smallest variance be of the same order is added to ease the presentation of the theoretical results. More general results can be obtained without this specification however quite convoluted in form.

**Assumption 3:** (Reveal probability)

We assume that each tensor entry  $(i, j, k)$  for all  $i \in [n]$ ,  $j \in [n]$  and  $k \in [n]$  is observed with independent and equal probability  $p$  which satisfies,

$$p \geq \frac{C\mu^3\lambda^{*2}\log^2(d)}{(\lambda^* + \min_{u \in \{a,b,c\}} \sigma_{Mu}^*)^2 d^{3/2}}, \quad (4.5)$$

where  $C$  is a constant.

**Assumptions 4:** (Initialization error)

Define the initialization errors for the tensor components as  $\epsilon_{0_T} := \max_{r \in [R]} \{\|\mathbf{a}^0 - \mathbf{a}^*\|_2, \|\mathbf{b}^0 - \mathbf{b}^*\|_2, \|\mathbf{c}^0 - \mathbf{c}^*\|_2, \frac{|\lambda^0 - \lambda^*|}{\lambda^*}\}$  and let the initialization error for the matrix components be defined as  $\epsilon_{0_M} := \max_{u \in \{a,b,c\}} \{\|\mathbf{v}_u^0 - \mathbf{v}_u^*\|_2, \frac{|\omega_u^0 - \omega_u^*|}{\omega_u^*}\}$ . Assume that

$$\epsilon_0 := \max\{\epsilon_{0_T}, \epsilon_{0_M}\} = O(1). \quad (4.6)$$

Assumptions 3 and 4 are special cases of the corresponding assumptions provided in Chapter 3 with the rank of the tensor and matrix being set to one.

**Assumption 5** (Signal-to-noise condition)

We assume that tensor and matrices weights satisfy the following condition,

$$\sigma_{max} \preceq \frac{\lambda_{min}^{*2}p + \omega_{min}^{*2}}{\lambda_{max}^*p\sqrt{pn^2\log^2(n)}} \quad \text{and} \quad (\sigma_{M_u})_{max} \preceq \frac{\lambda_{min}^{*2}p + \omega_{min}^{*2}}{\omega_{max}^*\sqrt{n\log(n)}} \quad \text{with} \quad \mathbf{u} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}, \quad (4.7)$$

where  $\preceq$  is used to represent asymptotic inequality.

Assumption 5 is a special case of the signal to noise ratio condition which general form was introduced in Chapter 3. In the special case in which the noise of the matrix and tensor are assumed to be sub-Gaussian, as specified in Assumption 2. The signal-to-noise condition takes the form presented above in Assumption 5. This derivation is obtained as a result of concentration inequalities of sub-Gaussian random variables coupled with the fact that

the spectral norm sub-Gaussian tensor scales as  $O(\sigma kn \log(k))$  where  $k$  is the number of dimensions of the tensor [65].

#### 4.2.2 Distributional Guarantee of Tensor Factors

In this section we present the results of the distributional analysis for the tensor factors recovered through Algorithm 2. We present the results for the factor estimate using the first component  $\mathbf{a}^*$  as reference since the distributional guarantees for the other two factors  $\mathbf{b}^*$  and  $\mathbf{c}^*$  are identical. Also for conciseness, we introduce the following additional vectors and matrices notations. Let  $\mathbf{A} \in \mathbb{R}^{n^2}$  be a vector and  $\mathbf{Q}_i^* \in \mathbb{R}^{n^2 \times n^2}$  and  $(\mathbf{Q}_M^*)_i \in \mathbb{R}^{n \times n}$  be diagonal matrices with diagonal elements defined respectively as follows,

$$\mathbf{A}^* := \text{vec}(\mathbf{b}^* \otimes \mathbf{c}^*),$$

$$(\mathbf{Q}_i^*)_{(j,k),(j,k)} := (\sigma_T^{*2})_{i,j,k}, \quad \text{for } 1 \leq j, k \leq n; \quad ((\mathbf{Q}_M^*)_i)_{l,l} := (\sigma_M^{*2})_{i,l}, \quad \text{for } 1 \leq l \leq n,$$

where we use  $(j, k)$  to denote  $(j-1)n + k$ .

**Theorem 4.2.1** (Distributional guarantees for tensor factor estimates under Gaussian noise). *Provided Assumptions 1, 2, 3, 4 and 5 are met and given that  $\{\mathcal{E}_{T_{i,j,k}}\}_{1 \leq i,j,k \leq n}$ ,  $\{\mathcal{E}_{Ma_{i,l}}\}_{1 \leq i,l \leq n}$ ,  $\{\mathcal{E}_{Mb_{j,l}}\}_{1 \leq j,l \leq n}$  and  $\{\mathcal{E}_{Mc_{k,l}}\}_{1 \leq k,l \leq n}$  are Gaussian, after running the require number of iterations mentioned in Theorem 3.4.1, with high probability the following holds,*

$$\mathbf{u}_i - \mathbf{u}_i^* = \mathbf{Y} + \mathbf{W},$$

where  $\mathbf{u} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$ ,  $\|\mathbf{W}\|_\infty = o\left(\frac{\lambda^* p \sigma_{T_{max}} + \omega^* (\sigma_M)_{max}}{\lambda^{*2} p + \omega^{*2}}\right)$ ,  $Y_i \sim N(0, \Sigma_i)$  and

$$\Sigma_i = \frac{\lambda^{*2} p \mathbf{A}^{*\top} \mathbf{Q}_i^* \mathbf{A}^* + \omega^{*2} \mathbf{v}^{*\top} (\mathbf{Q}_M^*)_i \mathbf{v}^*}{(\lambda^{*2} p + \omega^{*2})^2} \quad \text{for } 1 \leq i \leq n. \quad (4.8)$$

Theorem 4.2.1 reveals that the estimation error of the recovered tensor factor  $\mathbf{a}$  can be decomposed into a Gaussian component  $\mathbf{Y}$  and a residual term  $\mathbf{W}$  with the residual term being dominated by the Gaussian term and hence can be neglected. Theorem 4.2.1 therefore



reveals that the estimates of the tensor components  $\mathbf{a}$ ,  $\mathbf{b}$  and  $\mathbf{c}$  are nearly un-biased estimates of the true tensor factors  $\mathbf{a}^*$ ,  $\mathbf{b}^*$  and  $\mathbf{c}^*$ , with estimation errors being approximately Gaussian. Moreover, it is worth noting that the variance for the tensor factor errors derived in (4.8) is always smaller than that derived in Cai, Poor, and Chen [47] when the noise of the matrix is smaller or equivalent to that of the tensor noise. This property becomes more apparent in the case where the reveal probability  $p$  gets smaller. We see in Chapter 4.2.3 and later in the simulation results that this translate into tighter confidence intervals for tensor factors compared to those derived through the use of a standalone tensor completion methods in the high missing data regime.

Next, in Theorem 4.2.2 we show that the distributional properties presented in Theorem 4.2.1 can be extended to accommodate a broader family of noise beyond Gaussian noise.

**Theorem 4.2.2** (Distributional guarantees for tensor factor estimates under general noise). *Assume that  $\{\mathcal{E}_{T_{i,j,k}}\}_{1 \leq i,j,k \leq n}$ ,  $\{\mathcal{E}_{Ma_{i,l}}\}_{1 \leq i,l \leq n}$ ,  $\{\mathcal{E}_{Mb_{j,l}}\}_{1 \leq j,l \leq n}$  and  $\{\mathcal{E}_{Mc_{k,l}}\}_{1 \leq k,l \leq n}$  are not necessarily Gaussian but still satisfy Assumption 2, then the statement in Theorem 4.2.1 still holds, with the exception that  $\mathbf{Y}$  is no longer necessarily Gaussian but instead satisfies the following condition*

$$|P\{\mathbf{Y}_i \in \mathcal{A}\} - P\{g_i \in \mathcal{A}\}| \leq o(1), \quad (4.9)$$

for any convex set  $\mathcal{A} \subset \mathbb{R}$ , where  $g_i \sim N(0, \Sigma_i^*)$  with variance  $\Sigma_i^*$  defined as in (4.8).

### 4.2.3 Confidence Interval for Tensor Factors

The distributional guarantees highlighted in Theorem 4.2.1 and 4.2.2, gives us the tools to tackle the problem of uncertainty quantification for the tensor factors. To do that, it remains to be shown how to compute the variances  $\Sigma_i$  in since its expression in (4.8) depends on the true tensor and associated matrices components and weights  $\mathbf{a}^*$ ,  $\mathbf{b}^*$ ,  $\mathbf{c}^*$ ,  $\mathbf{v}^*$ ,  $\lambda^*$  and  $\omega^*$  as well as the true variance  $(\sigma_T^2)_{i,j,k}$  and  $(\sigma_M^2)_{i,l}$  of the noise tensor and noise matrices respectively which in practice are all unknown. We show that despite the lack of knowledge of these parameters we can still use a data driven approach to reliably estimate them using a simple plug-in procedure.

**Variance Estimation:** Although we do not have the true parameters mentioned above, we

still get reliable estimates of the variance parameter  $\Sigma_i$ . The overall construction relies on a plug-in estimation technique. Rather than trying to estimate the noise tensor and noise matrix entry variances  $(\sigma_T^2)_{i,j,k}$  and  $(\sigma_M^2)_{i,j,k}$ , we rely on estimating the noise tensor and noise matrix entries directly. These are then plugged into the formula of the  $\Sigma_i^*$  provided in (4.8) to produce an estimate of the factor error variance which we refer to as  $\hat{\Sigma}$ . The following steps are taken for computing the variance of the estimated tensor factor errors.

1. Estimate the noise tensor and noise matrices  $\{\mathcal{E}_{T_{i,j,k}}\}_{1 \leq i,j,k \leq n}$ ,  $\{\mathcal{E}_{Ma_{i,l}}\}_{1 \leq i,l \leq n}$ ,  $\{\mathcal{E}_{Mb_{j,l}}\}_{1 \leq j,l \leq n}$  and  $\{\mathcal{E}_{Mc_{k,l}}\}_{1 \leq k,l \leq n}$  as:

$$\begin{aligned}\hat{\mathcal{E}}_{T_{i,j,k}} &= \frac{1}{p}(\mathcal{T}_{i,j,k}^{obs} - \mathcal{T}_{i,j,k}) \text{ with } (i,j,k) \in \Omega; & (\hat{\mathcal{E}}_{Ma})_{i,l} &= (\mathbf{M}_a^{obs})_{i,j,k} - (\mathbf{M}_a)_{i,j,k}, \\ (\hat{\mathcal{E}}_{Mb})_{i,l} &= (\mathbf{M}_b^{obs})_{i,j,k} - (\mathbf{M}_b)_{i,j,k}; & (\hat{\mathcal{E}}_{Mc})_{i,l} &= (\mathbf{M}_c^{obs})_{i,j,k} - (\mathbf{M}_c)_{i,j,k}.\end{aligned}\quad (4.10)$$

We then construct the estimates diagonal matrices  $\mathbf{Q}^*$  and  $\mathbf{Q}_M^*$  as :

$$\begin{aligned}(\widehat{\mathbf{Q}}_i)_{(j,k),(j,k)} &:= \hat{\mathcal{E}}_{i,j,k}^2 \mathbb{I}_{\{(i,j,k) \in \Omega\}}, & ((\widehat{\mathbf{Q}}_{Ma})_i)_{l,l} &:= (\hat{\mathcal{E}}_{Ma})_{i,l}^2, \\ ((\widehat{\mathbf{Q}}_{Mb})_i)_{l,l} &:= (\hat{\mathcal{E}}_{Mb})_{j,l}^2 & ((\widehat{\mathbf{Q}}_{Mc})_i)_{l,l} &:= (\hat{\mathcal{E}}_{Mc})_{k,l}^2,\end{aligned}$$

where we use  $(j, k)$  to denote  $(j-1)n + k$ .

2. We estimate  $\mathbf{A}^*$ , using the plug-in estimator:  $\mathbf{A} = \mathbf{b} \otimes \mathbf{c}$ , where  $\mathbf{b}$  and  $\mathbf{c}$  are estimated using COSTCO. Similarly we use  $\mathbf{v}$ ,  $\lambda$  and  $\omega$  to be the estimates of  $\mathbf{v}^*$ ,  $\lambda^*$  and  $\omega^*$  obtained from Algorithm 2 respectively.
3. We then use the above estimators to substitute into expression (4.8) to get an estimator for  $\Sigma_i^*$ ,

$$\hat{\Sigma}_i = \frac{\lambda^2 p \mathbf{A}^\top \widehat{\mathbf{Q}}_i \mathbf{A} + \omega^2 \mathbf{v}^\top (\widehat{\mathbf{Q}}_M)_i \mathbf{v}}{(\lambda^2 p + \omega^2)^2} \quad \text{for } 1 \leq i \leq n. \quad (4.11)$$

**Confidence Interval:** Given the important parameters plug-in estimates derived above we construct an entry wise confidence interval for the tensor factors which serves as a procedure for uncertainty quantification for the unknown tensor.

For each  $1 \leq i, j, k \leq n$ , we construct a  $(1 - \alpha)$ -confidence interval for the  $i$ th entry of the tensor component  $\mathbf{u}^*$  as:

$$\mathbf{CI}_{\mathbf{u}_i^*}^{1-\alpha} := \left[ \mathbf{u}_i \pm \sqrt{\Sigma_i} \Phi^{-1}(1 - \alpha/2) \right], \quad (4.12)$$

where  $\mathbf{u} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$ ,  $\Phi^{-1}$  is the inverse CDF of a standard Gaussian and  $\Sigma_i$  is as constructed in (4.11). Next we prove the validity of the constructed confidence intervals in Theorem 4.2.3.

**Theorem 4.2.3** (Validity of tensor factors confidence interval). *Assuming all assumptions required for Theorem 4.2.2 holds. For any  $0 < \alpha < 1$  the confidence interval constructed in (4.12) obeys:*

$$\mathbb{P}\{\mathbf{u}_i^* \in \mathbf{CI}_{\mathbf{u}_i^*}^{1-\alpha}\} = 1 - \alpha + o(1), \quad \forall 1 \leq i \leq n \quad \text{and} \quad \mathbf{u} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}.$$

Theorem 4.2.3 demonstrates the validity of the uncertainty quantification procedure we proposed. In the following paragraph we highlight the properties of the proposed procedure for constructing the confidence interval for tensor factors.

- i. *Entrywise uncertainty quantification:* Our results enables valid uncertainty quantification for each entry of the tensor factor. This allows us to assess the risk of the estimation process at a factor entry level. To the best of our knowledge, this is the first work that provides uncertainty quantification at an entry level for tensor completion in the presence of covariate information.
- ii. *Heteroscedasticity compliance:* The proposed procedure adapts to heterogeneous and unknown noise distribution. The confidence intervals do not require knowledge of the tensor or covariate matrices noise distribution and hence is a distributional-free procedure which is highly desirable in practice.
- iii. *Tighter Confidence intervals:* As mentioned in the discussion of Theorem 4.2.1, the variance obtained in (4.8) is smaller than that produced using a standalone tensor completion method which translates to tighter confidence intervals as illustrated in Figure 4.5. Combined with the improvement in factors recovery accuracy proven in

Chapter 3, tighter confidence intervals for the factor entries means overall a more trustworthy and useful uncertainty quantification for the unknown tensor.

### 4.3 Simulations

In each simulation, we generate a third-order rank one tensor  $\mathcal{T}^* \in \mathbb{R}^{50 \times 50 \times 50}$  and rank one matrices  $\mathbf{M}_a^* \in \mathbb{R}^{50 \times 50}$ ,  $\mathbf{M}_b^* \in \mathbb{R}^{50 \times 50}$ ,  $\mathbf{M}_c^* \in \mathbb{R}^{50 \times 50}$ . We assume that the matrices  $\mathbf{M}_a^*$ ,  $\mathbf{M}_b^*$  and  $\mathbf{M}_c^*$  share components across the first, second and third mode of the tensor respectively just as is the case in the theory section. In order to form tensor  $\mathcal{T}^*$  and the matrices, we draw each entry of  $\mathbf{a}^* \in \mathbb{R}^{50 \times 1}$ ,  $\mathbf{b}^* \in \mathbb{R}^{50 \times 1}$ ,  $\mathbf{c}^* \in \mathbb{R}^{50 \times 1}$  and  $\mathbf{v}_a^* \in \mathbb{R}^{50 \times 1}$ ,  $\mathbf{v}_b^* \in \mathbb{R}^{50 \times 1}$ ,  $\mathbf{v}_c^* \in \mathbb{R}^{50 \times 1}$  from the iid standard normal distribution. We define  $\lambda^* = \|\mathbf{a}^* \times \mathbf{b}^* \times \mathbf{c}^*\|_2$  and  $\omega_a^* = \|\mathbf{a}^* \times \mathbf{v}_a^*\|_2$ , same goes for  $\omega_b^*$  and  $\omega_c^*$ . We normalize each of the vectors of  $\mathbf{a}^*$ ,  $\mathbf{b}^*$ ,  $\mathbf{c}^*$ ,  $\mathbf{v}_a^*$ ,  $\mathbf{v}_b^*$ ,  $\mathbf{v}_c^*$  to unit norm. The tensor  $\mathcal{T}^*$  and matrices are then formed as  $\mathcal{T}^* = \lambda^* \mathbf{a}^* \otimes \mathbf{b}^* \otimes \mathbf{c}^*$  and  $\mathbf{M}_a^* = \omega_a^* \mathbf{a}_1^* \otimes \mathbf{v}_a^*$ , same goes for  $\mathbf{M}_b^*$  and  $\mathbf{M}_c^*$ . We then add noise to the tensor and matrices using the following setup  $\mathcal{T} = \mathcal{T}^* + \sigma_T \mathcal{N}_T$  and  $\mathbf{M} = \mathbf{M}^* + \sigma_M \mathcal{N}_M$ , where  $\mathcal{N}_T$  and  $\mathcal{N}_M$  are a tensor and a matrix of the same size as  $\mathcal{T}^*$  and  $\mathbf{M}^*$  respectively, whose entries are generated from the standard normal distribution. We simulate the uniformly missing at random pattern in the tensor data by generating entries of the reveal tensor  $\mathbf{\Omega} \in \mathbb{R}^{50 \times 50 \times 50}$  from the binomial distribution with reveal probability  $p$ . The noisy tensor  $P_\Omega(\mathcal{T})$  with missing data is finally obtained as  $P_\Omega(\mathcal{T}) = \mathcal{T} * \mathbf{\Omega}$ , where  $*$  is the element-wise multiplication.

#### 4.3.1 Empirical Distribution

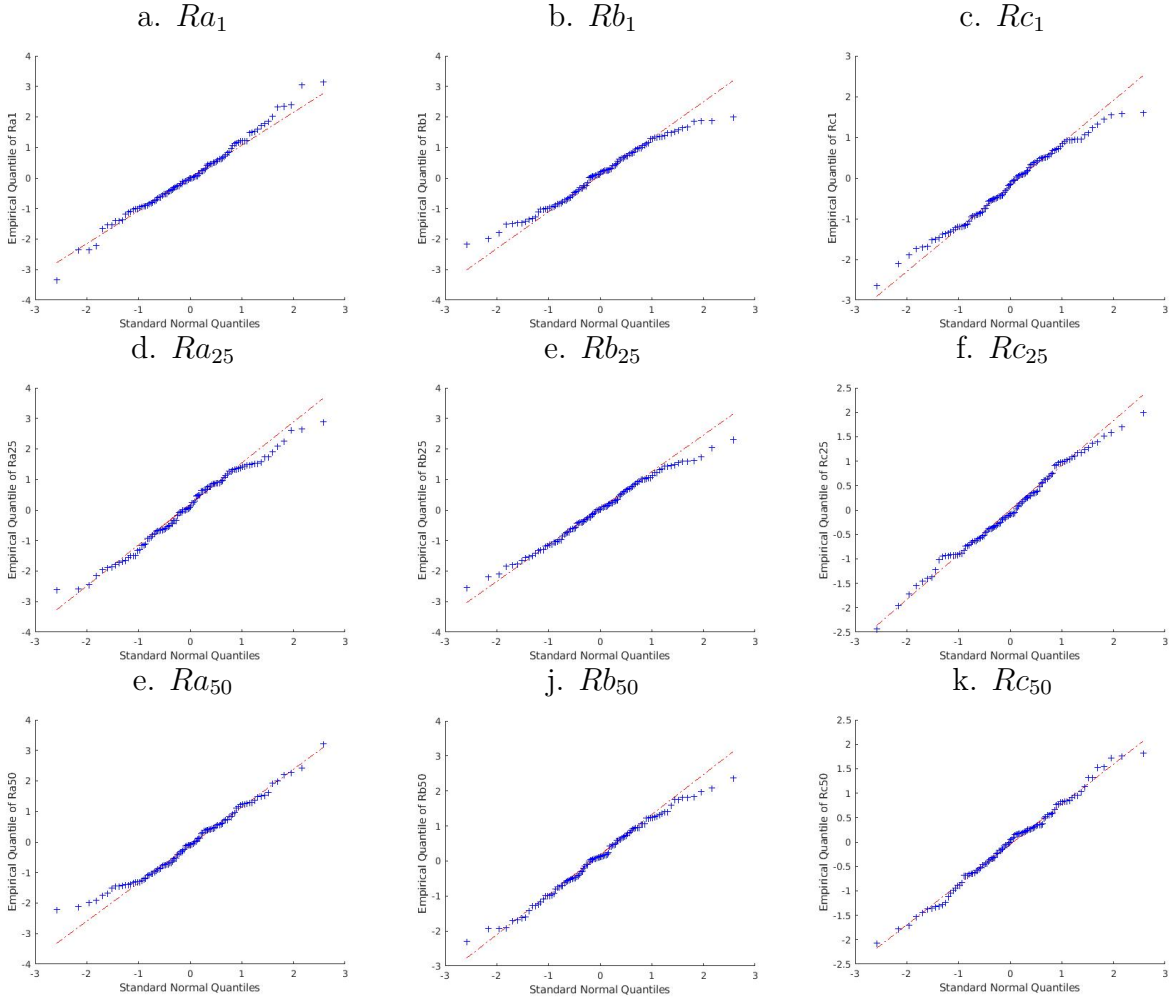
Recall that in Chapter 4.2.2, we characterized the distribution of the tensor components error as being Gaussian with variance parameter  $\Sigma^*$ . In this section, we investigate through simulation experiments whether this claim holds when the variance parameter  $\Sigma^*$  is replaced with the estimate  $\hat{\Sigma}$  constructed in (4.11).

Define the normalized estimation error for each tensor component entry as follows

$$Ra_i := \frac{\mathbf{a}_i^* - \mathbf{a}_i}{\hat{\Sigma}_a(i)}; \quad Rb_j := \frac{\mathbf{b}_j^* - \mathbf{b}_j}{\hat{\Sigma}_b(j)}; \quad Rc_k := \frac{\mathbf{c}_k^* - \mathbf{c}_k}{\hat{\Sigma}_c(k)} \quad \text{for } 1 \leq i, j, k \leq n,$$

where  $\hat{\Sigma}_u(i)$  with  $\mathbf{u} \in \{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$  are constructed using the procedure proposed in (4.11).

Figure 4.3 represents the Q-Q plots of  $Ra_i$ ,  $Rb_j$  and  $Rc_k$  where  $i, j, k \in \{1, 25, 50\}$  plotted against that of a standard Gaussian random variable (red line). Data for the figure were obtained over 100 trials of estimating the tensor and matrix components using `COSTCO` followed by computing  $\hat{\Sigma}$  using (4.11). The reveal probability in this experiment was set to  $p = 0.1$  and the parameter for the tensor and matrix noise were  $\sigma_T = \sigma_M = 0.01$ . We notice in Figure 4.3 that as demonstrated in the theory, the empirical distributions of the normalized estimation error for tensor factor entries are all well approximated by a standard Gaussian distribution.



**Figure 4.3.** Q-Q (quantile-quantile) plots of normalized factor entry error with  $p = 0.1$  and  $\sigma_T = \sigma_M = 0.01$ .

### 4.3.2 Empirical Coverage Rate of Confidence Intervals

We have shown in Section 4.2 that the proposed construction of confidence intervals for tensor factor entries are valid and we have also discussed the fact that under certain condition, the variance for the tensor factors the estimation error  $\Sigma^*$  using **COSTCO** is smaller than that achieved using a standalone tensor completion method. We explore these properties through a series of experiments where we compared the performance of **COSTCO** to that of a standalone tensor completion method. We start by exploring the effects of tensor noise and the reveal probability  $p$  on the coverage rate of the constructed confidence intervals.

We let  $CD_i$  denote the empirical coverage rate for the 95% confidence interval for  $\mathbf{a}_i$  over 100 independent trials. That is, we compute the percentage of time the estimate  $Ra_i$  fall in the interval  $[-1.96, 1.96]$ . We use the notation  $\text{Mean}(CD)$  and  $\text{Sd}(CD)$  to refer to the mean and standard deviation of the coverage rate over all 100 replicates and over  $1 \leq i \leq 50$ . Table 4.1 and Table 4.2 provide a side by side comparison of the coverage rate of the constructed confidence intervals using **COSTCO** versus using a standalone tensor completion method. In the case of **COSTCO**, both tensor and matrix factors were initialized using SVD decomposition. For the standalone tensor completion method we use Jain and Oh's completion algorithm **tenALS** [24] where the RTPM method was used to initialize the tensor factors (as recommended by the authors). In order to facilitate comparison of the results from both algorithms, the number of iterations for both algorithms was set to 200 or iteration stopped when the tolerance condition from Algorithm 2  $tol \leq 10^{-5}$  was reached. In Table 4.1 the reveal probability was set to 0.1 and  $\sigma_M = 0.01$ , while the tensor noise parameter  $\sigma_T$  was varied between  $\{0.001, 0.01, 0.1, 1, 2\}$ . Note that for sake of comparison between the two algorithms we could not set a lower reveal probability as the **tenALS** algorithm would fail to converge when the reveal probability was lower than 0.1 unlike **COSTCO**.

Table 4.1 shows that the empirical coverage rate for confidence interval constructed using **COSTCO** are mostly around 95% however, the coverage obtained using **tenALS** underestimates the true coverage rate as the tensor noise level increases. These results highlight again the need for including side information in the process of the tensor completion since the standalone tensor completion method are very sensitive to noise level when the reveal probability is

very low as is the case in this experiment. We then investigated the effect of the reveal

**Table 4.1.** Empirical coverage rate for 95% confidence interval of **COSTCO** versus **tenALS** with varying tensor noise parameter  $\sigma$ .

Tensor Noise level $\sigma$	<b>COSTCO</b>		<b>tenALS</b>	
	Mean(CD)	Sd(CD)	Mean(CD)	Sd(CD)
0.001	0.9576	0.0194	0.9370	0.0596
0.01	0.9542	0.0209	0.9310	0.0654
0.1	0.9558	0.0222	0.9370	0.0788
1	0.9538	0.0239	0.9350	0.0797
2	0.9440	0.0218	0.8980	0.0953

probability  $p$  on the coverage rate in a second experiment. In Table 4.2 the noise parameter for both tensor and matrix were set to  $\sigma_T = \sigma_M = 0.01$  and the reveal probability  $p$  was varied from  $\{0.01, 0.05, 0.1, 0.2\}$  which corresponds to  $\{99, 95, 90, 80, 50\}$  percent missing data respectively. Again we notice that the coverage rate for **COSTCO** are all close to the true 95% making it robust to the missing probability level. Yet the coverage rate for the standalone tensor completion method **tenALS** is largely affected by the reveal probability. The results in Table 4.2 highlights the benefit of including side information in the model of the tensor completion problem. Indeed as the reveal probability decreases estimates obtained using **COSTCO** remain reliable as proven in Chapter 3 and therefore provide a better estimate for the estimation error variance used in the construction of the confidence intervals. Whereas standalone tensor completion method do not have such an advantage, resulting in poor estimation results leading to very bias variance estimates and poor coverage rates.

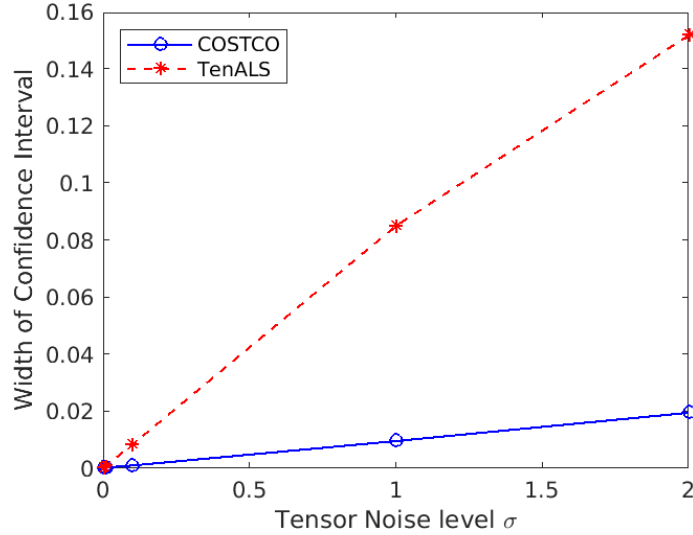
**Table 4.2.** Empirical coverage rate for 95% confidence interval of **COSTCO** versus **tenALS** with varying tensor reveal probabilities  $p$ .

Tensor reveal probability $p$	<b>COSTCO</b>		<b>tenALS</b>	
	Mean(CD)	Sd(CD)	Mean(CD)	Sd(CD)
0.5	0.9520	0.0200	0.9560	0.0459
0.2	0.9562	0.0230	0.9520	0.0474
0.1	0.9572	0.0193	0.9380	0.0540
0.05	0.9572	0.0205	0.8310	0.0920
0.01	0.9588	0.0172	0.7520	0.1924

### 4.3.3 Tightness of Confidence Intervals

In Chapter 4.2.2 we discussed the fact that provided the noise of the matrix does not dominate that of the tensor the variance  $\Sigma^*$  is smaller than what would be obtained using a standalone tensor completion method. Leading to relatively tighter confidence interval for the recovered tensor factor estimates. In the series of experiments to follow, we investigate the effect of the tensor noise and tensor reveal probability on the length of the confidence intervals obtained using the construction proposed in Subsection 4.2.3 for **COSTCO** and the confidence interval obtained by using a standalone tensor completion method **tenALS**. The set up for the experiments which results are presented in Figure 4.4 and Figure 4.5 are similar to the set up for those provided in Table 4.1 and Table 4.2 respectively.

Figure 4.4 shows the effects of tensor noise on the length of the confidence interval for



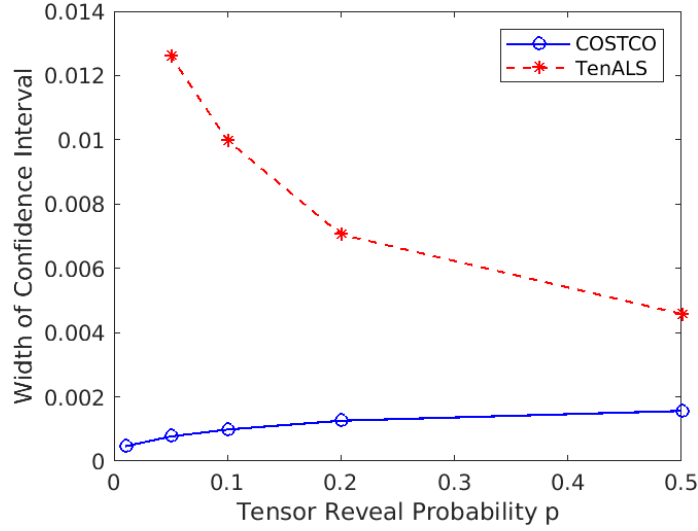
**Figure 4.4.** Width of constructed confidence interval with varying tensor noise level for **COSTCO** versus **tenALS**

the recovered tensor factor entries. We notice that for both algorithms, the width of the confidence interval increases with an increasing noise tensor variance. However, the rate of increase of the confidence interval is drastically different between **COSTCO** and **tenALS** with the later exhibiting a much faster increase in the width of the confidence interval. Moreover, we notice that at each value of the tensor noise, the confidence interval width for **COSTCO** is much smaller than that of the standalone tensor completion method. This phenomenon is



due to the addition of the matrices information in the the tensor completion problem. Since the matrix noise was set to be smaller or equal to that of the tensor in this experiment, the overall variance estimate in (4.8) which can be seen as a weighted average between matrix and tensor noise becomes smaller than the variance obtained through a standalone completion method.

Figure 4.5 showcases the relationship between the reveal probability  $p$  and the width of the



**Figure 4.5.** Width of constructed confidence interval with varying tensor reveal probability for **COSTCO** versus **tenALS**

confidence interval for the recovered tensor factors. Notice that the trends in the curves for the two algorithms are very different, with the curve for **COSTCO** exhibiting a positive slope while the curve for **tenALS** shows a negative slope. Although downward sloping, the width of the confidence intervals for **tenALS** is always above that of **COSTCO** for every value of  $p$ . Both curves then converge to an asymptote with the asymptote of the red line being greater than that of the blue line. This shows that the width of the confidence interval using **COSTCO** when the variance of the noise tensor and noise matrix are similar leads to tighter confidence intervals than that obtained from a standalone completion method.

#### 4.4 Proof of Main Theorem

In this section we provide the proofs of the main theoretical results presented in Theorems 4.2.1, 4.2.2 and 4.2.3. For simplicity, in the following proofs we drop the subscript which serves to differentiate the three covariate matrices  $\mathbf{M}_a$ ,  $\mathbf{M}_b$  and  $\mathbf{M}_c$  and only refer to each as  $\mathbf{M}$ . Hence  $\mathcal{E}_{Ma}$  and  $n_{va}$  become  $\mathcal{E}_M$  and  $n_v$  respectively. We also consider the case where all tensor and matrices modes have the same dimensions  $n$  that is  $n = n = n = n_v = n$ . Recall the definition of  $\mathbf{d}_u$  from Chapter 3,

$$\mathbf{d}_u =: \mathbf{u} - \mathbf{u}^*, \quad \text{and} \quad \|\mathbf{d}_u\|_2 = \|\mathbf{u} - \mathbf{u}^*\|_2, \quad (4.13)$$

and

$$\Delta_\lambda := \left| \frac{\lambda - \lambda^*}{\lambda^*} \right| \quad \text{and} \quad \Delta_\omega := \left| \frac{\omega - \omega^*}{\omega} \right|, \quad (4.14)$$

where  $\mathbf{u}$  could be any of  $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{v}$ .

##### 4.4.1 Proof of Theorem 4.2.1: Gaussian Noise

The proof of Theorem 4.2.1 consists in showing that the estimate of tensor component recovered at the end of Algorithm 2 can be written as a sum of a Gaussian random variable and a vanishing residual. We provide the proof for the tensor component  $\mathbf{a}$  since the proof for the two other components  $\mathbf{b}$  and  $\mathbf{c}$  follow a similar analysis.

We start by recalling the expression for the update for  $\mathbf{a}$  in Algorithm 2.

$$\textbf{Tensor Components: } \mathbf{a} = \frac{\lambda P_\Omega(\mathcal{T})(\mathbf{I}, \mathbf{b}, \mathbf{c}) + \omega_a \mathbf{M}_a \mathbf{v}_a}{\lambda^2 P_\Omega(\mathbf{I}, \mathbf{b}^2, \mathbf{c}^2) + \omega_a^2}. \quad (4.15)$$

Note that the horizontal double lines in the expressions above indicates element-wise fraction and the squares in the denominator represent the element-wise squaring. The expression of  $\mathbf{a}$  in (4.15) can be decomposed in the following way.

$$\mathbf{a} = \mathbf{D}^{-1}(\text{unfold}(P_\Omega(\lambda \mathcal{E}_T))\mathbf{A} + \omega \mathcal{E}_M \mathbf{v}) + \mathbf{D}^{-1}(\text{unfold}(P_\Omega(\lambda \mathcal{T}^*))\mathbf{A} + \omega \mathbf{M}^* \mathbf{v}) \quad (4.16)$$

where  $\mathbf{D}$  is defined as  $\mathbf{D} := (\lambda^2 \mathbf{A}^\top \mathbf{A} + \omega^2 \mathbf{v}^\top \mathbf{v})$ ; that is  $\mathbf{D}$  is a  $n \times n$  diagonal matrix with diagonal elements  $\mathbf{D}_{ii} = \lambda^2 \sum_{j,k} \delta_{i,j,k} \mathbf{b}^2(j) \mathbf{c}^2(k) + \omega^2$  and  $\mathbf{A}$  is as defined in (4.16).

Let  $\mathbf{a}$  be the estimate obtained after the required number of iterations recommended in Theorem 3.4.1 for estimating the tensor factors. We can express the error between the estimate  $\mathbf{a}$  and the true component  $\mathbf{a}^*$  as,

$$\mathbf{a} - \mathbf{a}^* = \underbrace{\mathbf{D}^{*-1} \left( \text{unfold}(P_\Omega(\lambda^* \mathcal{E}_T)) \mathbf{A}^* + \omega^* \mathcal{E}_M \mathbf{v}^* \right)}_{:= \mathbf{Y}} + \sum_{i=1}^3 \mathbf{W}_i, \quad (4.17)$$

with  $\mathbf{W}_i$  taking the following forms,

$$\mathbf{W}_2 = \text{unfold}(P_\Omega(\mathcal{E}_T)) \left( \lambda^* \mathbf{D}^{*-1} \mathbf{A}^* - \lambda \mathbf{D}^{-1} \mathbf{A} \right) \quad (4.18)$$

$$\mathbf{W}_3 = \mathcal{E}_M \left( \omega^* \mathbf{D}^{*-1} \mathbf{v}^* - \omega \mathbf{D}^{-1} \mathbf{v} \right) \quad (4.19)$$

$$\mathbf{W}_1 = \mathbf{a}^* - \mathbf{D}^{-1} \left( \text{unfold}(P_\Omega(\lambda \mathcal{T}^*)) \mathbf{A} + \omega \mathbf{M}^* \mathbf{v} \right), \quad (4.20)$$

where for the tensor  $\mathcal{T} \in \mathbb{R}^{n \times n \times n}$ , the notation  $\text{unfold}(\mathcal{T})$  represent a  $\mathbb{R}^{n \times n^2}$  matrix obtained from the mode 1 matricization of tensor  $\mathcal{T}$  as defined in Kolda and Bader [29]. Given the decomposition in (4.17), proving Theorem 4.2.1 then boils down to proving that  $\mathbf{Y}$  is approximately Gaussian and that  $\sum_{i=1}^3 \mathbf{W}_i$  can be bounded by a term which is dominated by  $\mathbf{Y}$ . We achieve that by stating and proving the following lemmas:

- Lemma 14 which reveals that under Gaussian noise, each entry of  $\mathbf{Y}$  is approximately a Gaussian random variable.
- Lemmas 15-17 which deliver upper bounds for the  $l_\infty$  norm of the residual quantities  $W_1$ ,  $W_2$  and  $W_3$  respectively and show that these quantities are negligible compared to a typical entry of  $\mathbf{Y}$ .

Theorems 4.2.1 then follows immediately by combining the results of Lemma 14- Lemma 17. Next we state the key Lemmas mentioned above.

**Lemma 14.** *Given the assumptions in Theorem 4.2.1 holds, with high probability the expression  $\mathbf{Y}$  in (4.17) can be written as  $\mathbf{Y} = \mathbf{Z} + \mathbf{X} + \mathbf{W}_0$  such that for any  $1 \leq i \leq n$ ,  $\mathbf{Z}_i + \mathbf{X}_i \sim N(0, \Sigma_i^*)$  with variance matrix  $\Sigma_i^*$  defined as in (4.8) and*

$$\|\mathbf{W}_0\|_\infty \leq \frac{\lambda \sigma_{max}}{\lambda^{*2}p + \omega^{*2}} \left( \frac{\mu^2 \log(n)}{n} \right). \quad (4.21)$$

Lemma 14 shows that  $\mathbf{Y}$  is approximately Gaussian where the Gaussian approximation residual is characterized by  $\mathbf{W}_0$ . The proof of the lemma involves bounding the infinite norm of  $\mathbf{W}_0$  and showing that the variance of  $\mathbf{Z}_i + \mathbf{X}_i$  is well approximated by  $\Sigma_i^*$ . This is done by using various concentration inequality results as well as applying properties of sub-Gaussian and sub-exponential random variables. Next we state the three other lemmas used to bound the expressions of  $\mathbf{W}_1, \mathbf{W}_2$  and  $\mathbf{W}_3$ .

**Lemma 15.** *Given the assumptions of Theorem 4.2.1 hold we have,*

$$\|\mathbf{W}_2\|_\infty \leq \frac{\lambda^* \sigma_{max} \|\mathbf{d}\|_2 \sqrt{np \log(n)}}{\lambda^{*2}p + \omega^{*2}}, \quad (4.22)$$

with probability  $1 - n^{-9}$ .

**Lemma 16.** *Given the assumptions of Theorem 4.2.1 hold we have,*

$$\|\mathbf{W}_3\|_\infty \leq \frac{\omega^* (\sigma_M)_{max} \|\mathbf{d}_v\|_2 \sqrt{n \log(n)}}{\lambda^{*2}p + \omega^{*2}}, \quad (4.23)$$

with probability  $1 - n^{-9}$

**Lemma 17.** *Given the assumptions of Theorem 4.2.1 after the number of interactions recommended for Algorithm 2 in Theorem 3.4.1, we have with probability  $1 - n^{-9}$ ,*

$$\|\mathbf{W}_1\| \leq \frac{4\lambda^{*2}p \|\mathbf{d}\|_2 \left( \|\mathbf{d}\|_2 + \sqrt{\frac{\mu^3 \log(n)}{pn^{1.5}}} \right) + \omega^{*2} \left( \frac{1}{2} \|\mathbf{d}\|_2^2 + \Delta_\omega \right)}{\lambda^{*2}p + \omega^{*2}}. \quad (4.24)$$

Notice that given the condition of  $p$  in Assumption 3, the bounds in (4.22), (4.23) and (4.24) are dominated by that of (4.21). Using this knowledge and recalling the fact that

$\mathbf{a} - \mathbf{a}^* := \mathbf{Y} + \sum_{i=1}^4 \mathbf{W}_i$  completes the proof of the theorem.  $\square$

#### 4.4.2 Proof of Theorem 4.2.2: General Noise

Theorem 4.2.2 extends the normality result obtained in Theorem 4.2.1 to the case with a tensor and matrices with non-Gaussian noise. Given Theorem 4.2.1 the proof of Theorem 4.2.2 consists in applying the results of Lemma 18 then showing that the bound derived in (4.25) has a rate of  $o(1)$ .

**Lemma 18.** *Given the assumptions in Theorem 4.2.2, with probability at least  $1 - n^{-9}$  Lemma 14 still holds excepts that  $\mathbf{Y}$  obeys,*

$$|P\{\mathbf{Y}_i \in \mathcal{A}\} - P\{g_i \in \mathcal{A}\}| \leq \frac{2^{5/2}}{\sqrt{\pi}} \left( \frac{\sqrt{p}(\sigma_T^3)_{\max}\mu^2/n}{(\sigma_T^3)_{\min}} + \frac{(\sigma_M^3)_{\max}\mu/\sqrt{n}}{(\sigma_M^3)_{\min}} \right), \quad (4.25)$$

for any convex set  $\mathcal{A} \subset \mathbb{R}^d$ , where  $g_i \sim N(0, \Sigma_i^*)$  where the variance  $\Sigma_i^*$  defined as in (4.8).

Notice that when  $(\sigma_T)_{\max} \asymp (\sigma_T)_{\min}$  and  $(\sigma_M)_{\max} \asymp (\sigma_M)_{\min}$  right side of the inequality in (4.25) can be simplified to reveal a rate equal to  $o(n^{-1})$ . The proof of Lemma 18 is provided in section 4.5. This result shows that the estimation error for tensor factor is approximately Gaussian even when the noise distribution is not necessarily Gaussian which completes the proof of the theorem.

#### 4.4.3 Proof of Theorem 4.2.3: Confidence Intervals for Tensor Factors

Given the results in Theorem 4.2.2 and using the continuous mapping theorem, it is true that,

$$\sup_{\tau \in \mathbb{R}} |\mathbb{P}\{\mathbf{a} - \mathbf{a}^* \leq \tau \sqrt{\Sigma_i^*}\} - \Phi(\tau)| = o(1). \quad (4.26)$$

The proof of the theorem consists in showing that equation (4.26) still holds even after substituting  $\Sigma_i^*$  with the plug-in estimate  $\Sigma_i$  proposed in (4.11).

To do that, notice that the standardized error for each entry of  $\mathbf{a}$  can be written as follows,

$$\frac{\mathbf{a}_i - \mathbf{a}_i^*}{\sqrt{\Sigma_i}} = \frac{\mathbf{a}_i - \mathbf{a}_i^*}{\sqrt{\Sigma_i^*}} + \underbrace{\frac{\mathbf{a}_i - \mathbf{a}_i^*}{\sqrt{\Sigma_i}} - \frac{\mathbf{a}_i - \mathbf{a}_i^*}{\sqrt{\Sigma_i^*}}}_{:=\theta_i}. \quad (4.27)$$

We show in Lemma 19 that the residual  $\theta_i$  is negligible. More specifically that  $\theta_i = o(1)$  with high probability. Using the decomposition of  $\frac{\mathbf{a}_i - \mathbf{a}_i^*}{\sqrt{\Sigma_i}}$  above combined with the union bound and making use of the property of the CDF of standard normal variables, we get that for  $\tau \in \mathbb{R}$  we have,

$$\begin{aligned} \mathbb{P}\{\mathbf{a} - \mathbf{a}^* \leq \tau \sqrt{\Sigma_i}\} - \Phi(\tau) &\leq \mathbb{P}\{\mathbf{a} - \mathbf{a}^* \leq (\tau + \epsilon) \sqrt{\Sigma_i^*}\} + \mathbb{P}\{|\theta_i| \geq \epsilon\} - \Phi(\tau) \\ &\leq \Phi(\tau + \epsilon) - \Phi(\tau) + o(1) + \mathbb{P}\{|\theta_i| \geq \epsilon\} \\ &\leq \epsilon + o(1) + \mathbb{P}\{|\theta_i| \geq \epsilon\}, \end{aligned}$$

where the second inequality is due to the bound in (4.27) and the third inequality is due to the property of the standard normal CDF. Given that the bound on  $\theta_i$  proven in Lemma 19 holds with probability at least  $1 - n^{-9}$ , we can define  $\epsilon$  above to be equal to the upper bound of  $\theta_i$ . If that is the case and using the union bound we get,

$$\sup_{\tau \in \mathbb{R}} |\mathbb{P}\{\mathbf{a} - \mathbf{a}^* \leq \tau \sqrt{\Sigma_i}\} - \Phi(\tau)| = o(1).$$

which completes the proof of the theorem. □

**Lemma 19.** *Given the assumptions of Theorem 4.2.3, with probability  $1 - n^{-9}$  we have,*

$$|\theta_i| = o(1). \quad (4.28)$$

The bound in Lemma 19 allows us to show that the estimate  $\Sigma_i$  using the construction and plug-in estimate proposed in Chapter 4.2.3 is close to the true error variance  $\Sigma_i^*$ . Obtaining this bound requires showing that the tensor variance estimate used in the construction of the confidence interval  $\hat{\mathcal{E}}_{T_{i,j,k}} = \frac{1}{p}(\mathcal{T}_{i,j,k}^{obs} - \mathcal{T}_{i,j,k})$  with  $(i, j, k) \in \Omega$ ; and  $(\hat{\mathcal{E}}_{Ma})_{i,l} = (\mathbf{M}_a^{obs})_{i,j,k} - (\mathbf{M}_a)_{i,j,k}$ , are good enough so that the plug-in estimate  $\Sigma_i$  is good approximation of  $\Sigma_i^*$ . Details of the proof of Lemma 19 are left for Chapter 4.5.

## 4.5 Additional Results

In this section we provide details of the derivation for the proofs of Lemmas 14- 19.

### 4.5.1 Proof of Lemma 14

Define  $\{z_{i,j,k}\}_{1 \leq i,j,k \leq n}$  and  $\{y_{i,l}\}_{1 \leq i,l \leq n}$  as follows:

$$z_{i,j,k} = \lambda^* \mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{A}_{(jk)}^* \mathbf{D}_{ii}^{*-1} \quad \text{for } 1 \leq i, j, k \leq n, \quad (4.29)$$

$$x_{i,l} = \omega^* \mathcal{E}_{M_{i,l}} \mathbf{v}_l^* \mathbf{D}_{ii}^{*-1} \quad \text{for } 1 \leq i, l \leq n, \quad (4.30)$$

where recall the expression  $(jk)$  is defined as  $(jk) := (j-1)n + k$ . Also define  $\mathbf{Q}_T \in \mathbb{R}^{n^2 \times n^2}$  and  $\mathbf{Q}_M \in \mathbb{R}^{n \times n}$  as :

$$(\mathbf{Q}_T^*(i))_{(jk)(jk)} := \sigma_{i,j,k}^{*2} \quad \text{for } 1 \leq i, j, k \leq n, \quad (4.31)$$

$$(\mathbf{Q}_M^*(i))_{l,l} := (\sigma_M^*)_{i,l} \quad \text{for } 1 \leq i, l \leq n, \quad (4.32)$$

Then we can decompose  $\mathbf{Y}$  in the following way,

$$\mathbf{Y}_i = \mathbf{Z}_i + \mathbf{X}_i = \sum_{jk} z_{i,j,k} + \sum_l x_{i,l}. \quad (4.33)$$

Notice the form of the following expected values,

$$\Sigma_T^*(i) := E(\mathbf{Z}_i^\top \mathbf{Z}_i) = \lambda^{*2} p \mathbf{D}_{ii}^{*-1} \mathbf{A}^{*\top} (\mathbf{Q}_T^*)_i \mathbf{A}^* \mathbf{D}_{ii}^{-1}, \quad (4.34)$$

$$\Sigma_M^*(i) := E(\mathbf{X}_i^\top \mathbf{X}_i) = \omega^{*2} \mathbf{D}_{ii}^{*-1} \mathbf{v}^{*\top} (\mathbf{Q}_M^*)_i \mathbf{v}^* \mathbf{D}_{ii}^{-1}. \quad (4.35)$$

We define the following two variables  $\mathbf{S}_T^*(i)$  and  $\mathbf{S}_M^*(i)$  for  $1 \leq i \leq n$  which conditional of  $\{\delta_{i,j,k}\}_{1 \leq i,j,k \leq n}$  meets the following property  $\mathbb{E}(\mathbf{S}_T^*(i)) = \Sigma_T^*(i)$  and  $\mathbb{E}(\mathbf{S}_M^*(i)) = \Sigma_M^*(i)$  respectively in the following way,

$$\mathbf{S}_T^*(i) := \lambda^{*2} \sum_{jk} \sigma_{i,j,k}^2 \delta_{i,j,k} \mathbf{D}_{ii}^{*-1} \mathbf{A}_{(jk)(jk)}^{*\top} \mathbf{A}_{(jk)(jk)}^* \mathbf{D}_{ii}^{*-1} \quad (4.36)$$

$$\mathbf{S}_M^*(i) := \omega^{*2} \sum_l \sigma_{i,l}^2 \mathbf{D}_{ii}^{*-1} \mathbf{v}_l^{*\top} \mathbf{v}_l^* \mathbf{D}_{ii}^{*-1}. \quad (4.37)$$

It is then easy to see that conditional on  $\{\delta_{i,j,k}\}_{1 \leq i,j,k \leq n}$   $\mathbf{Z}_i$  is zero mean Gaussian with variance parameter  $\mathbf{S}_T^*(i)$  and  $\mathbf{X}_i$  is also zero mean Gaussian with variance  $\mathbf{S}_M^*(i) = \Sigma_M^*(i)$ . We now need to show that  $\Sigma_T^*(i)$  is a good approximation for  $\mathbf{S}_T^*(i)$ .

Given the expression in (4.34) we can bound  $\Sigma_T^*(i)$  in the following way  $p \frac{\lambda^{*2} \sigma_{min}^{*2}}{\mathbf{D}_{ii}^2} \leq \Sigma_T^*(i) \leq p \frac{\lambda^{*2} \sigma_{max}^{*2}}{\mathbf{D}_{ii}^2}$  by using the fact that  $\mathbf{A}^{*\top} \mathbf{A}^* = \sum_{jk} \mathbf{b}_j^{*2} \mathbf{c}_k^{*2} = 1$  since  $\|\mathbf{b}^*\|_2 = \|\mathbf{c}^*\|_2 = 1$  and by applying the results of Lemma 20. Notice that  $(\mathbf{S}_T^*)(i)$  is positive hence we can take its square root. Also since we have shown that  $\mathbf{Z}_i$  is Gaussian with mean zero and variance  $\Sigma_T^*(i)$ , it follows that the variable  $\mathbf{Z}_i (\mathbf{S}_T^{*-1/2})(i) (\Sigma_T^{*-1/2})(i)$  is also a Gaussian random variable with zero mean and variance  $\Sigma_T^*(i)$ .

For convenience sake, in the rest of the proof of this lemma we use the following simplified notations:  $(\mathbf{S}_T^*)(i) = \mathbf{S}_i^*$  and  $(\Sigma_T^{*-1/2})(i) = \Sigma_i^{*-1/2}$ . We now prove that under the high



probability event of Lemma 14,  $\mathbf{Z}_i^*$  and  $\mathbf{Z}_i^* \mathbf{S}_i^{*-1/2} \Sigma_i^{*1/2}$  are very close. To accomplish that, we bound their absolute difference as,

$$\begin{aligned}
|\mathbf{Z}_i^* - \mathbf{Z}_i^* \mathbf{S}_i^{*-1/2} \Sigma_i^{*1/2}| &\leq |\mathbf{Z}_i^* \mathbf{S}_i^{*-1/2} (\mathbf{S}_i^{*1/2} - \Sigma_i^{*1/2})| \\
&\leq |\mathbf{Z}_i^*| |\mathbf{S}_i^{*-1/2}| |\mathbf{S}_i^{*1/2} - \Sigma_i^{*1/2}| \\
&\leq |\mathbf{Z}_i^*| |\mathbf{S}_i^{*-1/2}| \left| \frac{1}{\mathbf{S}_i^{*1/2} + \Sigma_i^{*1/2}} \right| (\mathbf{S}_i - \Sigma_i) | \\
&\leq |\mathbf{Z}_i^*| \frac{(\lambda^{*2} p + \omega^{*2})^2}{\lambda^{*2} p \sigma_{min}^{*2} + \omega^{*2} 2 \sigma_{min}^{*2}} (\mathbf{S}_i - \Sigma_i) |,
\end{aligned}$$

with probability  $1 - n^{-11}$ . Where the third inequality above is obtained using the fact that for any positive real numbers  $a$  and  $b$  it holds that  $|(a - b)| = |(a^{1/2} - b^{1/2})(a^{1/2} + b^{1/2})|$ . The fourth inequality is obtained from the bound  $|\mathbf{S}_i^{*-1/2}|$  derived using Lemma 26. Then using the bound on  $|\mathbf{Z}_i^*|$  and  $|\mathbf{S}_i - \Sigma_i|$  from Lemma 24 and 25 and applying the union bound we get that

$$\begin{aligned}
\|\mathbf{W}_0\|_\infty &:= \max_i |\mathbf{Z}_i^* - \mathbf{Z}_i^* \mathbf{S}_i^{*-1/2} \Sigma_i^{*1/2}| \\
&\leq \frac{\lambda \sigma_{max}}{\lambda^{*2} p + \omega^{*2}} \left( \frac{\mu^2 \log(n)}{n} \right),
\end{aligned} \tag{4.38}$$

with probability  $1 - n^{-9}$ , where the last inequality holds for  $p \geq \frac{\mu^4 \log(n)^2}{n^2}$  and  $\sigma_{min} \asymp \sigma_{max}$ . Hence  $\mathbf{Z}_i$  is well approximated by a Gaussian distribution with mean zero and variance  $(\Sigma_T^*)(i)$ .

Combining this result to the fact that  $\mathbf{X}_i$  is also Gaussian with mean zero and variance  $(\Sigma_M^*)(i)$  and defining  $(\Sigma^*)(i) := (\Sigma_T^*)(i) + (\Sigma_M^*)(i)$  and the fact that  $\mathbf{Z}_i$  and  $\mathbf{X}_i$  are independent, leads to the desired results. That is  $\mathbf{Y} = \mathbf{Z} + \mathbf{X} + \mathbf{W}_0$  is approximately Gaussian with mean zero and variance  $\Sigma^*$ .  $\square$

### 4.5.2 Proof of Lemma 15

Before we start the proof of Lemma 15 we visit the notion of the **Leave-one-out** method first introduced in [47]. We adapt the method to the case of the coupled tensor and matrices and use this new version of the method as an essential part in the proof of this lemma. The method entails decoupling certain slices of the tensor, so that the entries used for the estimation of a particular tensor entry is independent to the tensor and matrix noise. Using this method allows us to circumvent the hurdle of using concentration inequalities when the assumption of independence between the variables are not met. Full detail of the **Leave-one-out** method for the couple tensor and matrix is provided in Chapter 4.7.1. We refer the reader to that section before proceeding with the rest of the the lemma's proof. Given the expression of  $\mathbf{W}_2$  in (4.18), we can fix  $i$  and decompose each entry of  $\mathbf{W}_2$  in the following manner,

$$\begin{aligned} W_2(i) &= e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T)) (\lambda^* \mathbf{D}^{*-1} \mathbf{A}^* - \lambda \mathbf{D}^{-1} \mathbf{A}) \\ &= \underbrace{e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T)) (\lambda \mathbf{A} - \lambda^* \mathbf{A}^*) \mathbf{D}^{-1}}_{res_1} + \underbrace{e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T)) \lambda^* \mathbf{A}^* (\mathbf{D}^{-1} - \mathbf{D}^{*-1})}_{res_2}. \end{aligned}$$

We proceed with bounding  $|res_1|$  and  $|res_2|$  respectively. Since  $res_1$  is a scalar it holds that  $|res_1| = \|res_1\|_2$  the same goes for  $res_2$ . Hence we can write,

$$\begin{aligned} \|res_1\|_2 &= \|e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T)) (\lambda \mathbf{A} - \lambda^* \mathbf{A}^*) \mathbf{D}^{-1}\|_2 \\ &\leq \|\mathbf{D}^{-1}\|_2 |e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T)) (\lambda \mathbf{A} - \lambda^* \mathbf{A}^*)| \\ &= \|\mathbf{D}^{-1}\|_2 |e_i^\top \underbrace{(\lambda P_\Omega(\mathcal{E}_T) \times_2 \mathbf{b} \times_3 \mathbf{c} - \lambda^* P_\Omega(\mathcal{E}_T) \times_2 \mathbf{b}^* \times_3 \mathbf{c}^*)}_{:=res_{11}}|. \end{aligned}$$

Given the definition of the estimate  $\mathbf{b}^{(i)}$  and  $\mathbf{c}^{(i)}$  introduced in Chapter 4.7.2, that is we let  $m = i$ , we can decompose  $res_{11}$  as following,

$$res_{11} = \left( \underbrace{\lambda P_{\Omega}(\mathcal{E}_T) \times_2 \mathbf{b}^{(i)} \times_3 \mathbf{c}^{(i)} - \lambda^* P_{\Omega}(\mathcal{E}_T) \times_2 \mathbf{b}^* \times_3 \mathbf{c}^*}_{:=res_{111}} \right) + \left( \underbrace{\lambda P_{\Omega}(\mathcal{E}_T) \times_2 \mathbf{b} \times_3 \mathbf{c} - \lambda^* P_{\Omega}(\mathcal{E}_T) \times_2 \mathbf{b}^{(i)} \times_3 \mathbf{c}^{(i)}}_{:=res_{112}} \right).$$

Recall the definition of  $\mathbf{d} := \mathbf{b} - \mathbf{b}^*$  and define  $\mathbf{d}^{(i)} := \mathbf{b}^{(i)} - \mathbf{b}^*$ . The  $i$ th element of the vector  $res_{111}$  can be written as:

$$\begin{aligned} res_{111}(i) &= \lambda \mathbf{b}^{(i)\top} P_{\Omega}(\mathcal{E}_T)_{i,:,\cdot} \mathbf{c}^{(i)} - \lambda^* \mathbf{b}^{*\top} P_{\Omega}(\mathcal{E}_T)_{i,:,\cdot} \mathbf{c}^* \\ &\leq 4\lambda^* \mathbf{d}^{(i)\top} P_{\Omega}(\mathcal{E}_T)_{i,:,\cdot} \mathbf{c}^* + 2\lambda^* \mathbf{d}^{(i)\top} P_{\Omega}(\mathcal{E}_T)_{i,:,\cdot} \mathbf{d}^{(i)}, \end{aligned} \quad (4.39)$$

where  $P_{\Omega}(\mathcal{E}_T)_{i,:,\cdot}$  represents the  $i$ th mode-1 slice of the tensor  $P_{\Omega}(\mathcal{E}_T)$ . The inequality above was obtained from the fact that  $\|\lambda \mathbf{b} - \lambda^* \mathbf{b}^*\| \leq 2\lambda^* \mathbf{d}$ .

Notice that by construction,  $\mathbf{d}^{(i)}$  is independent of the  $i$ th mode-1 slice of  $P_{\Omega}(\mathcal{E}_T)$  (refer to the details of the `leave-one-out` method in Chapter 4.7.2). We can therefore write (4.39) as the sum of independent zero mean random variables. That is

$$\mathbf{d}^{(i)\top} P_{\Omega}(\mathcal{E}_T)_{i,:,\cdot} \mathbf{c}^* = \sum_{jk} \mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{d}_j^{(i)} \mathbf{c}_k^* \text{ and } \mathbf{d}^{(i)\top} P_{\Omega}(\mathcal{E}_T)_{i,:,\cdot} \mathbf{d}^{(i)} = \sum_{jk} \mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{d}_j^{(i)} \mathbf{d}_k^{(i)}.$$

Making use of the the u-mass condition and using the fact that  $\mathcal{E}_{i,j,k}$  is Gaussian, we get,

$$L_3 := \|\mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{d}_j^{(i)} \mathbf{c}_k^*\|_{\psi_1} \leq \sigma_{max} \frac{\mu}{\sqrt{n}} \|\mathbf{d}^{(i)}\|_{\infty}$$

$$B_3 := \sum_{jk} \mathcal{E}_{i,j,k}^2 \delta_{i,j,k}^2 (\mathbf{d}_j^{(i)} \mathbf{c}_k^*)^2 \leq p \sigma_{max}^2 \|\mathbf{d}^{(i)}\|_2^2$$

$$L_4 := \|\mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{d}_j^{(i)} \mathbf{d}_k^{(i)}\|_{\psi_1} \leq \sigma_{max} \|\mathbf{d}^{(i)}\|_{\infty}^2$$

$$B_3 := \sum_{jk} \mathcal{E}_{i,j,k}^2 \delta_{i,j,k}^2 (\mathbf{d}_j^{(i)} \mathbf{c}_k^*)^2 \leq p \sigma_{max}^2 \|\mathbf{d}^{(i)}\|_2^4,$$

where  $\|*\|_{\psi_1}$ , represents the sub-exponential norm and the first and third equations above are obtained by applying the properties of sub-Gaussian and sub-exponential norms similar

to that used in Lemma 25.

Applying Bernstein inequality on each of the random variables in (4.39) we get with probability  $1 - n^{-11}$ ,

$$|res_{111}| \leq 6\lambda^* \sigma_{max} \|\mathbf{d}^{(i)}\|_2 \left( \sqrt{2p \log(n)} + \frac{\mu}{\sqrt{n}} \log n \right) \quad (4.40)$$

Next we bound  $res_{112}$ . Using the Cauchy-Schwartz inequality we get

$$\begin{aligned} \|res_{112}\|_2 &= \|(\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)})^\top P_\Omega(\mathcal{E}_T)_{i,:} \mathbf{c}^{(i)} + (\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)})^\top P_\Omega(\mathcal{E}_T)_{i,:} (\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)})\|_2 \\ &\leq \|(\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)})\|_2 \|P_\Omega(\mathcal{E}_T)_{i,:} \mathbf{c}^{(i)}\|_2 + \|P_\Omega(\mathcal{E}_T)_{i,:}\|_2 \|\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)}\|_2^2 \end{aligned} \quad (4.41)$$

We then make use of Lemmas 27 in order to bound the right hand side of the inequality above, which yields with probability  $1 - n^{-11}$ ,

$$\|res_{112}\|_2 \leq \sigma_{max} \|(\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)})\|_2 \left( (\sqrt{np \log(n)} + \frac{\mu}{\sqrt{n}} \log n) + \|(\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)})\|_2 (\sqrt{np} + \log n) \right) \quad (4.42)$$

Combining the (4.40) and (4.42) and the bound in Lemma 20 for  $\|\mathbf{D}^{-1}\|_2$  we get that with probability  $1 - n^{-9}$ ,

$$\begin{aligned} \|res_1\|_2 &\leq \frac{1}{\lambda^{*2}p + \omega^{*2}} 6\lambda^* \sigma_{max} \|\mathbf{d}^{(i)}\|_2 \left( \sqrt{2p \log(n)} + \frac{\mu}{\sqrt{n}} \log n \right) + \\ &\frac{\sigma_{max} \|(\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)})\|_2}{\lambda^{*2}p + \omega^{*2}} \left( \sqrt{np \log(n)} + \frac{\mu}{\sqrt{n}} \log n \right) + \\ &\frac{\sigma_{max} \|(\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)})\|_2}{\lambda^{*2}p + \omega^{*2}} \left( \|\lambda \mathbf{b} - \lambda^{(i)} \mathbf{b}^{(i)}\|_2 (\sqrt{np} + \log n) \right). \end{aligned} \quad (4.43)$$

We move to bounding  $\|res_2\|_2$  in the following way.

$$\begin{aligned} \|res_2\|_2 &= |e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T)) \lambda^* \mathbf{A}^* (\mathbf{D}^{-1} - \mathbf{D}^{*-1})| \\ &\leq \|e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T)) \lambda^* \mathbf{A}^*\|_2 \|(\mathbf{D}^{-1} - \mathbf{D}^{*-1})\|_2 \end{aligned}$$

We bound each of the terms on the right side of the inequality above. Notice that  $e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T))\lambda^* \mathbf{A}^* = \lambda^* \sum_{jk} \mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{A}_{(jk)}^*$  is the sum of independent random variables. Hence,

$$L_5 := \max_{jk} \|\mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{A}_{(jk)}^*\|_{\psi_1} \leq \sigma_{max}^* \|\mathbf{A}_{(jk)}^*\|_\infty \leq \sigma_{max}^* \frac{\mu^2}{n}$$

$$B_5 := \sum_{jk} E(\mathcal{E}_{i,j,k}^2 \delta_{i,j,k}^2 \mathbf{A}_{(jk)}^{*2}) \leq \sigma_{max}^2 p$$

It follows from the Bernstein inequality that

$$\|e_i^\top \mathbf{unfold}(P_\Omega(\mathcal{E}_T))\lambda^* \mathbf{A}^*\|_2 \leq \lambda^* \sigma_{max}^* (\sqrt{p \log(n)} + \frac{\mu^2}{n} \log n) \quad (4.44)$$

$$\asymp \sigma_{max}^* (\sqrt{p \log(n)}), \quad (4.45)$$

with probability  $1 - n^{-11}$ .

Next we have

$$\begin{aligned} \|\mathbf{D}^{-1} - \mathbf{D}^{*-1}\|_2 &\leq \|\mathbf{D}^{-1}\|_2 \|\mathbf{D} - \mathbf{D}^*\|_2 \|\mathbf{D}^{*-1}\|_2 \\ &\leq \frac{6\lambda^{*2} p \|\mathbf{d}\|_2 + o(\lambda^* \|\Delta\|_2 + \omega^* \|\Delta\|_2)}{(\lambda^{*2} p + \omega^{*2})^2} \end{aligned} \quad (4.46)$$

with probability  $1 - n^{-9}$ , were for the last inequality we applied Lemmas 20 and 21. Combining (4.45) and (4.46) we get

$$\|res_2\|_2 \leq \lambda^* \sigma_{max}^* (\sqrt{p \log(n)} + \frac{\mu^2}{n} \log n) \frac{6\lambda^{*2} p \|\mathbf{d}\|_2 + o(\lambda^* \|\Delta\|_2 + \omega^* \|\Delta\|_2)}{(\lambda^{*2} p + \omega^{*2})^2} \quad (4.47)$$

with probability  $1 - n^{-9}$ .

The proof of the Lemma 15 is then completed by combining (4.43) and (4.47).  $\square$

### 4.5.3 Proof of Lemma 16

Given the expression of  $\mathbf{W}$  in (4.18) we get

$$\begin{aligned} \mathbf{W}_3(\mathbf{i}) &= \mathcal{E}_M \left( \omega^* \mathbf{D}^{*-1} \mathbf{V}^* - \omega \mathbf{D}^{-1} \mathbf{V} \right) \\ &\leq \underbrace{e_i^\top \mathbf{unfold}(\mathcal{E}_M) (\omega \mathbf{V} - \omega^* \mathbf{V}^*) \mathbf{D}^{-1}}_{res_{M1}} + \underbrace{e_i^\top \mathbf{unfold}(\mathcal{E}_M) \omega^* \widetilde{\mathbf{V}}^* (\mathbf{D}^{-1} - \mathbf{D}^{*-1})}_{res_{M2}} \end{aligned}$$

We bound  $|res_{M1}|$  and  $|res_{M2}|$  in what follows.

$$\begin{aligned} \|res_{M1}\|_2 &= \|e_i^\top \mathbf{unfold}(\mathcal{E}_M) (\omega \mathbf{v} - \omega^* \mathbf{v}^*) \mathbf{D}^{-1}\|_2 \\ &\leq \|\mathbf{D}^{-1}\|_2 |e_i^\top \mathbf{unfold}(\mathcal{E}_M) (\omega \mathbf{v} - \omega^* \mathbf{v}^*)| \\ &= \|\mathbf{D}^{-1}\|_2 |e_i^\top \underbrace{(\omega \mathcal{E}_M \times \mathbf{v} - \omega^* \mathcal{E}_M \times \mathbf{v}^*)}_{res_{M11}}| \end{aligned}$$

Since we already have a bound for  $\|\mathbf{D}^{-1}\|_2$  we proceed with bounding the term  $res_{M11}$  in the inequality above.

$$res_{M11} = \left( \underbrace{\omega \mathcal{E}_M \times \mathbf{v}^{(i)} - \omega^* \mathcal{E}_M \times \mathbf{v}^*}_{res_{M111}} \right) + \left( \underbrace{\omega \mathcal{E}_M \times \mathbf{v} - \omega^* \mathcal{E}_M \times \mathbf{v}^{(i)}}_{res_{112}} \right).$$

Recall the definition of  $\mathbf{d}_v := \mathbf{v} - \mathbf{v}^*$  and define  $\mathbf{d}_v^{(i)} := \mathbf{v}^{(i)} - \mathbf{v}^*$ . Then the  $i$ th element of the vector above can be written as:

$$\begin{aligned} res_{M111}(\mathbf{i}) &= \omega (\mathcal{E}_M)_{i,:} \mathbf{v}^{(i)} - \omega^* (\mathcal{E}_M)_{i,:} \mathbf{v}^* \\ &= \omega^* (\mathcal{E}_M)_{i,:} \mathbf{d}_v^{(i)}, \end{aligned} \tag{4.48}$$

where we  $(\mathcal{E}_M)_{i,:}$  represents the  $i$ th row slice of the matrix  $\mathcal{E}_M$ .

By construction,  $\mathbf{d}^{(i)}$  is independent of the  $i$ th row of  $(\mathcal{E}_M)$ . We can therefore write (4.48) as

the sum of independent zero mean random variables. That is  $(\mathcal{E}_M)_{i,:} \mathbf{d}^{(i)} = \sum_l \mathcal{E}_{M_{i,l}} \mathbf{d}_l^{(i)}$ . By applying the u-mass assumption we get

$$\begin{aligned} L_{M3} &:= \|\mathcal{E}_{M_{i,l}} \mathbf{d}_l^{(i)}\|_{\psi_1} \leq (\sigma_M)_{\max} \|\mathbf{d}^{(i)}\|_{\infty} \\ B_{M3} &:= \sum_l \mathcal{E}_{M_{i,l}}^2 \mathbf{d}_l^{(i)2} \leq \sigma_{M_{\max}}^2 \|\mathbf{d}^{(i)}\|_2^2. \end{aligned}$$

Applying Bernstein inequality on each of the random variables  $res_{M111}(i)$  we get with probability  $1 - n^{-11}$

$$|res_{M111}| \leq \omega^*(\sigma_M)_{\max} \left( \|\mathbf{d}_v^{(i)}\|_2 + \|\mathbf{d}_v^{(i)}\|_{\infty} \log n \right). \quad (4.49)$$

Next we bound  $res_{112}$  by applying the Cauchy Schwartz inequality followed by the Bernstein inequality to get,

$$\begin{aligned} \|res_{M112}\|_2 &\leq \|\omega \mathbf{v} - \omega \mathbf{v}^{(i)}\|_2 \|(\mathcal{E}_M)_{i,:}\|_{\infty} \\ &\leq (\sigma_M)_{\max} \|\omega \mathbf{v} - \omega \mathbf{v}^{(i)}\|_2 (\sqrt{n} + \log(n)). \end{aligned} \quad (4.50)$$

Combining the (4.49) and (4.50) and the bound on  $\|\mathbf{D}^{-1}\|_2$  we get with probability  $1 - n^{-11}$

$$\|res_{M1}\|_2 \leq \frac{\omega^*(\sigma_M)_{\max} \left( \|\mathbf{d}_v^{(i)}\|_2 + \|\mathbf{d}_v^{(i)}\|_{\infty} \log n \right) + \left( (\sigma_M)_{\max} \|\omega \mathbf{v} - \omega \mathbf{v}^{(i)}\|_2 (\sqrt{n} + \log(n)) \right)}{\lambda^{*2} p + \omega^{*2}}. \quad (4.51)$$

We move on to bounding  $res_{M2}$

$$\begin{aligned} \|res_{M2}\|_2 &= e_i^{\top} \mathcal{E}_M \omega^* \mathbf{v}^* (\mathbf{D}^{-1} - \mathbf{D}^{*-1}) \\ &\leq \|e_i^{\top} \mathcal{E}_M \omega^* \mathbf{v}^*\|_2 |(\mathbf{D}^{-1} - \mathbf{D}^{*-1})|. \end{aligned}$$

Notice that  $e_i^\top \mathcal{E}_M \omega^* \mathbf{v}^* = \omega^* \sum_l \mathcal{E}_{i,l} \mathbf{v}_l^*$  is the sum of independent random variables. Hence

$$\begin{aligned} L_{M6} &:= \max_l \|\mathcal{E}_{M,i,l} \mathbf{v}_l^*\|_{\psi_1} \leq (\sigma_M)_{\max} \|\mathbf{v}_l^*\|_\infty \leq (\sigma_M)_{\max} \frac{\mu}{\sqrt{n}} \\ B_{M6} &:= \sum_l \mathbb{E}(\mathcal{E}_{M,i,l}^2) \mathbf{v}_l^{*2} \leq (\sigma_M^2)_{\max}. \end{aligned}$$

It follows from the Bernstein inequality that,

$$\|e_i^\top \mathcal{E}_M \omega^* \mathbf{v}^*\|_2 \leq \omega^* (\sigma_M)_{\max} (1 + \frac{\mu}{\sqrt{n}} \log n), \quad (4.52)$$

with probability  $1 - n^{-11}$ .

Combining (4.52) and the bound on  $\|\mathbf{D}^{-1} - \mathbf{D}^{*-1}\|_2$  established in (4.46) we get

$$\|res_{M2}\|_2 \leq \omega^* \sigma_{\max}^* (1 + \frac{\mu}{\sqrt{n}} \log n) \frac{6\lambda^{*2} p \|\mathbf{d}\|_2 + o(\lambda^* \|\Delta\|_2 + \omega^* \|\Delta\|_2)}{(\lambda^{*2} p + \omega^{*2})^2}, \quad (4.53)$$

with probability  $1 - n^{-9}$ . The proof of the Lemma is completed by combining (4.51) and (4.53).  $\square$

#### 4.5.4 Proof of Lemma 17

Let  $\mathbf{E}$  and  $\mathbf{H}$  be  $n \times n$  diagonal matrices with diagonal elements,

$$\mathbf{E}_{ii} = \sum_{j,k} \delta_{i,j,k} \mathbf{b}^*(j) \mathbf{c}^*(k) \mathbf{b}(j) \mathbf{c}(k) ; \quad \mathbf{H}_{ii} = \sum_l \mathbf{v}^*(l) \mathbf{v}(l).$$

We can then express  $\|\mathbf{W}_1\|$  as

$$\begin{aligned} \|\mathbf{W}_1\| &= \|\mathbf{a}^* - \mathbf{D}^{-1} (\text{unfold}(P_\Omega(\lambda \mathcal{T}^*)) \mathbf{A} + \omega \mathbf{M}^* \mathbf{v})\|_2 \\ &= \|\mathbf{D}^{-1} (\lambda \lambda^* \mathbf{E} + \omega \omega^* \mathbf{H} - \mathbf{DI}) \mathbf{a}^*\| \\ &\leq \|\mathbf{D}^{-1} (\lambda \lambda^* \mathbf{E} + \omega \omega^* \mathbf{H} - \mathbf{DI})\|_2 \|\mathbf{a}^*\|_2 \\ &\leq \max_i \underbrace{|\mathbf{D}_{ii}^{-1}|}_{err_{11}} \underbrace{|\lambda \lambda^* \mathbf{E} + \omega \omega^* \mathbf{H} - \mathbf{DI}|_{ii}}_{err_{12}}, \end{aligned} \quad (4.54)$$



where the third inequality is due to the fact  $\mathbf{D}^{-1}(\lambda\lambda^*\mathbf{E} + \omega\omega^*\mathbf{H} - \mathbf{D}\mathbb{I})$  is a diagonal matrix hence its spectral norm is obtained by taking the maximum absolute value of its diagonal elements. We therefore proceed to getting an upper bound each of the maximum of each of the random variable elements in the equation above with high probability. To do that we first get an upper bound on each of the diagonal elements with high probability and make use of the union bound method to get a high probability bound on the maximums.

$$\begin{aligned} err_{12} &= |\lambda\lambda^* \sum_{jk} \delta_{i,j,k} \mathbf{b}^*(j) \mathbf{c}^*(k) \mathbf{b}(j) \mathbf{c}(k) + \omega\omega^* \langle \mathbf{v}^*, \mathbf{v} \rangle - (\lambda^2 \sum_{jk} \delta_{i,j,k} \mathbf{b}^2(j) \mathbf{c}^2(k) + \omega^2)| \\ &\leq \underbrace{|\lambda\lambda^* \sum_{jk} \delta_{i,j,k} \mathbf{b}^*(j) \mathbf{c}^*(k) \mathbf{b}(j) \mathbf{c}(k) - \lambda^2 \sum_{jk} \delta_{i,j,k} \mathbf{b}^2(j) \mathbf{c}^2(k)|}_{I_{121}} + \underbrace{|\omega\omega^* \langle \mathbf{v}^*, \mathbf{v} \rangle - \omega^2|}_{I_{122}}. \end{aligned}$$

We can bound  $I_{121}$  and  $I_{122}$  next

$$\begin{aligned} I_{122} &= |\omega\omega^* \langle \mathbf{v}^*, \mathbf{v} \rangle - \omega^2| \\ &\leq \omega\omega^* (|\langle \mathbf{v}^*, \mathbf{v} \rangle - 1| + \Delta_\omega) \\ &\leq \omega^{*2} (1 - \Delta_\omega) \left( \frac{1}{2} \|d_v\|_2^2 + \Delta_\omega \right) \end{aligned} \tag{4.55}$$

where the first inequality is due to using the triangle inequality, the fact that  $\omega = \omega - \omega^* + \omega^*$ .

The second inequality is obtained from the results of Lemma 23.

Next we also bound  $I_{121}$ .

$$\begin{aligned} I_{121} &= |\lambda\lambda^* \sum_{jk} \delta_{i,j,k} \mathbf{b}^*(j) \mathbf{c}^*(k) \mathbf{b}(j) \mathbf{c}(k) - \lambda^2 \sum_{jk} \delta_{i,j,k} \mathbf{b}^2(j) \mathbf{c}^2(k)| \\ &\leq \lambda\lambda^* \left( \left| \sum_{jk} \left( \delta_{i,j,k} \mathbf{b}^*(j) \mathbf{c}^*(k) \mathbf{b}(j) \mathbf{c}(k) - \delta_{i,j,k} \mathbf{b}^2(j) \mathbf{c}^2(k) \right) \right| + \Delta_\lambda \sum_{jk} \delta_{i,j,k} \mathbf{b}^2(j) \mathbf{c}^2(k) \right) \\ &\leq \left| \sum_{jk} \delta_{i,j,k} \mathbf{b}^*(j) \mathbf{d}^*(k) \mathbf{b}(j) \mathbf{c}(k) \right| + \left| \sum_{jk} \delta_{i,j,k} \mathbf{d}^*(j) \mathbf{c}^*(k) \mathbf{b}(j) \mathbf{c}(k) \right| \\ &\quad + \left| \sum_{jk} \delta_{i,j,k} \mathbf{d}^*(j) \mathbf{d}^*(k) \mathbf{b}(j) \mathbf{c}(k) \right|, \end{aligned}$$

where the last inequality is obtained using the triangle inequality and the fact that  $\mathbf{b}(j) = \mathbf{b}^*(j) + \mathbf{d}(j)$  and  $\mathbf{c}(k) = \mathbf{c}^*(k) + \mathbf{d}(k)$ . Next applying the results of Lemma 20, we get

$$\begin{aligned}
I_{121} &\leq \lambda^{*2}p (|\langle \mathbf{b}^*, \mathbf{b} \rangle \langle \mathbf{d}, \mathbf{c} \rangle| + |\langle \mathbf{d}, \mathbf{b} \rangle \langle \mathbf{c}^*, \mathbf{c} \rangle| + |\langle \mathbf{d}, \mathbf{b} \rangle \langle \mathbf{d}, \mathbf{c} \rangle| + \Delta_\lambda) \\
&\quad + \lambda^{*2}p \left( \|\mathbf{d}\|_2 \sqrt{\frac{\mu^3 \log(n)}{pn^{1.5}}} \right) \\
&\leq 4\lambda^{*2}p \|\mathbf{d}\|_2 \left( \|\mathbf{d}\|_2 + \sqrt{\frac{\mu^3 \log(n)}{pn^{1.5}}} \right), \tag{4.56}
\end{aligned}$$

where the last inequality above holds with probability  $1 - 2n^{-11}$  and holds for  $p \geq \frac{\mu^3 \log(n)}{n^{1.5}}$ . Combining equations (4.55) and (4.56) followed by making use of lemma (20) to bound the denominator of  $\|\text{err}_1\|_2$ , we get

$$\|\mathbf{W}_1\| \leq \frac{4\lambda^{*2}p \|\mathbf{d}\|_2 \left( \|\mathbf{d}\|_2 + \sqrt{\frac{\mu^3 \log(n)}{pn^{1.5}}} \right) + \omega^{*2}(\frac{1}{2}\|\mathbf{d}\|_2^2 + \Delta_\omega)}{\lambda^{*2}p + \omega^{*2}} \tag{4.57}$$

with probability  $1 - 2n^{-9}$ .

Provided the initialization error specified in Theorem 4.2.1 and after the number of iterations specified in Theorem 2 of COSTCO we get that  $\|\mathbf{W}_1\| = \frac{\lambda^{*2}po(1) + \omega^{*2}o(1)}{\lambda^{*2}p + \omega^{*2}}$ , which completes the proof of the lemma.  $\square$

#### 4.5.5 Proof of Lemma 18

The proof consists of showing that  $\mathbf{Y}_i$  converges in distribution to a Gaussian random variable  $g_i \sim N(0, \Sigma_i^*)$ . Specifically it involves quantifying the rate of the approximation error and showing that under the assumption of Theorem 4.2.2 it is negligible. To do so we make

use a Lyapunov-type theorem [66] stated in section 4.7.1. More specifically we look to bound the value of  $\rho$  defined in (4.79). In our case the  $\rho$  takes the following form,

$$\begin{aligned}
\rho &= \sum_{1 \leq i \leq n} \mathbb{E} \left[ \|\Sigma_i^{*-1/2} \mathbf{Y}_i\|_2^3 \right] \leq \sum_{1 \leq i \leq n} \mathbb{E} \left[ \|(\Sigma_T^{*-1/2})_i \mathbf{Z}_i\|_2^3 \right] + \sum_{1 \leq i \leq n} \mathbb{E} \left[ \|(\Sigma_M^{*-1/2})_i \mathbf{X}_i\|_2^3 \right] \\
&\leq \sum_{1 \leq i \leq n} \mathbb{E} \left[ \|\lambda^*(\mathcal{E}_T)_{i,j,k} \delta_{i,j,k} (\Sigma_T^{*-1/2})_i \mathbf{A}_{(jk),:}^* \mathbf{D}^{-1}\|_2^3 \right] + \sum_{1 \leq i \leq n} \mathbb{E} \left[ \|\omega^*(\mathcal{E}_M)_{i,l} (\Sigma_M^{*-1/2})_i \mathbf{v}_{l,:}^* \mathbf{D}^{-1}\|_2^3 \right] \\
&\leq \sum_{1 \leq i \leq n} \mathbb{E} \left[ \lambda^{*3}(\mathcal{E}_T)_{i,j,k}^3 \delta_{i,j,k}^3 \|(\Sigma_T^{*-1/2})_i\|^3 |\mathbf{A}_{(jk),:}^*|^3 |\mathbf{D}^{-1}|^3 \right] + \mathbb{E} \left[ \omega^{*3}(\mathcal{E}_M)_{i,l}^3 \|(\Sigma_M^{*-1/2})_i\|^3 |\mathbf{v}_l^*|^3 |\mathbf{D}^{-1}|^3 \right],
\end{aligned}$$

where in the first inequality we made use of the triangle inequality and in the third inequality we applied the Cauchy Schwarz inequality. We further simplify the right side of the inequality above by making use of the derivation of the raw absolute moment of standard Gaussian random variables from Winkelbauer [67] which yields,

$$\begin{aligned}
\rho &\leq \frac{2^{5/2} p \lambda^{*3} \sigma_T^3}{\sqrt{\pi}} |(\Sigma_T^{*-1/2})_i|^3 |\mathbf{A}_{(jk),:}^*| |\mathbf{D}^{-1}|^3 + \frac{2^{5/2} \omega^{*3} \sigma_M^3}{\sqrt{\pi}} |(\Sigma_M^{*-1/2})_i|^3 |\mathbf{v}_l^*| |\mathbf{D}^{-1}|^3 \\
&\leq \frac{2^{5/2}}{\sqrt{\pi}} \left( \frac{\sqrt{p} (\sigma_T^3)_{\max} \mu^2 / n}{(\sigma_T^3)_{\min}} + \frac{(\sigma_M^3)_{\max} \mu / \sqrt{n}}{(\sigma_M^3)_{\min}} \right), \tag{4.58}
\end{aligned}$$

The inequality in (4.58) is obtained by using the  $\mu$ -mass condition and the unit norm condition on the normalized tensor and matrix factors. Given the bound on  $\rho$  in (4.58) and invoking Theorem 4.78 with  $d = 1$  completes the proof of the Lemma.  $\square$

#### 4.5.6 Proof of Lemma 19

We start with fixing  $1 \leq i \leq n$ . We can decompose the expression of  $\theta_i$  in the following manner,

$$|\theta_i| := \left| \frac{\mathbf{a}_i - \mathbf{a}_i^*}{\sqrt{\Sigma_i}} - \frac{\mathbf{a}_i - \mathbf{a}_i^*}{\sqrt{\Sigma_i^*}} \right| = \frac{\overbrace{|\mathbf{a}_i - \mathbf{a}_i^*|}^{:=\beta_1} \overbrace{|\Sigma_i - \Sigma_i^*|}^{:=\beta_2}}{\underbrace{\sqrt{\Sigma_i \Sigma_i^*}}_{:=\beta_3} \underbrace{|\sqrt{\Sigma_i} + \sqrt{\Sigma_i^*}|}_{:=\beta_4}}. \tag{4.59}$$

In what follows, we work on bounding  $\beta_1$ ,  $\beta_2$ ,  $\beta_3$  and  $\beta_4$  defined above.

**Bounding  $\beta_1$ :**

$$|\mathbf{a}_i - \mathbf{a}_i^*| \leq \|\mathbf{a} - \mathbf{a}^*\|_2 \leq \sqrt{\Sigma_i} \left(1 + \frac{\mu}{\sqrt{n}} \log n\right), \quad (4.60)$$

where the last inequality is obtained by using the bound on  $\|\mathbf{a} - \mathbf{a}^*\|_2$  in Theorem 3.4.1 provided in Chapter 3 with both tensor and matrix noise being Gaussian.

**Bounding  $\beta_3$  and  $\beta_4$ :** We know that,

$$\sqrt{\Sigma_i^*} \geq \frac{\sqrt{\lambda^{*2} p \sigma_{\min}^2 + \omega^{*2} (\sigma_M)_{\min}^2}}{\lambda^{*2} p + \omega^{*2}}. \quad (4.61)$$

We claim that

$$\sqrt{\Sigma_i} \geq \sqrt{\Sigma_i^*} - |\sqrt{\Sigma_i} - \sqrt{\Sigma_i^*}| = \sqrt{\Sigma_i^*} (1 - o(1)).$$

If that is the case then it follows that

$$\beta_3 := \sqrt{\Sigma_i \Sigma_i^*} \geq \Sigma_i^* (1 - o(1)) \quad \text{and} \quad \beta_4 := \sqrt{\Sigma_i} + \sqrt{\Sigma_i^*} \geq 2\sqrt{\Sigma_i^*} (1 - o(1)). \quad (4.62)$$

It remains for us to show that  $|\Sigma_i - \Sigma_i^*| \leq o(1)\Sigma_i^*$ , which is done by bounding  $\beta_2$ .

**Bounding  $\beta_2$ :**

Let  $\dot{\mathbf{A}} := \mathbf{A} \odot \mathbf{D}^{-1}$ , where we recall that the symbol  $\odot$  represents the element-wise product operator. We can now decompose the expression of  $\beta_2$  in the following manner,

$$\begin{aligned} |\Sigma_i - \Sigma_i^*| &= \lambda^2 \dot{\mathbf{A}}^\top \mathbf{Q}_i \dot{\mathbf{A}} + \omega^2 \dot{\mathbf{V}}^\top (\mathbf{Q}_M)_i \dot{\mathbf{V}} - \lambda^{*2} \dot{\mathbf{A}}^{*\top} \mathbf{Q}_i^* \dot{\mathbf{A}}^* - \omega^{*2} \dot{\mathbf{V}}^{*\top} (\mathbf{Q}_M^*)_i \dot{\mathbf{V}}^* \\ &= \underbrace{\lambda^2 \dot{\mathbf{A}}^\top (\mathbf{Q}_i - \widehat{\mathbf{Q}}_i) \dot{\mathbf{A}} + \omega^2 \dot{\mathbf{V}}^\top ((\mathbf{Q}_M)_i - (\widehat{\mathbf{Q}}_M)_i) \dot{\mathbf{V}}}_{\beta_{21}} + \\ &\quad \underbrace{\lambda^2 \dot{\mathbf{A}}^\top \widehat{\mathbf{Q}}_i \dot{\mathbf{A}} + \omega^2 \dot{\mathbf{V}}^\top (\widehat{\mathbf{Q}}_M)_i \dot{\mathbf{V}} - \lambda^{*2} \dot{\mathbf{A}}^{*\top} \mathbf{Q}_i^* \dot{\mathbf{A}}^* - \omega^{*2} \dot{\mathbf{V}}^{*\top} (\mathbf{Q}_M^*)_i \dot{\mathbf{V}}^*}_{\beta_{22}}. \end{aligned}$$

We proceed with bounding the two variables  $\beta_{21}$  and  $\beta_{22}$  above.

**Bounding  $\beta_{21}$ :**

We begin by decomposing  $\beta_{21}$  in the following way,

$$\begin{aligned}\beta_{21} = & \underbrace{\lambda^{*2} \dot{\mathbf{A}}^{*\top} (\mathbf{Q}_i - \widehat{\mathbf{Q}}_i) \dot{\mathbf{A}}^* + \omega^{*2} \dot{\mathbf{V}}^{*\top} ((\mathbf{Q}_M)_i - (\widehat{\mathbf{Q}}_M)_i) \dot{\mathbf{V}}^*}_{\beta_{211}} + \\ & \underbrace{2\lambda^* (\lambda^* \dot{\mathbf{A}}^* - \lambda \dot{\mathbf{A}})^\top (\mathbf{Q}_i - \widehat{\mathbf{Q}}_i) \dot{\mathbf{A}}^* + \omega^* (\omega^* \dot{\mathbf{V}}^* - \omega \dot{\mathbf{V}})^\top ((\mathbf{Q}_M)_i - (\widehat{\mathbf{Q}}_M)_i) \dot{\mathbf{V}}^*}_{\beta_{212}} + \\ & \underbrace{(\lambda^* \dot{\mathbf{A}}^* - \lambda \dot{\mathbf{A}})^\top (\mathbf{Q}_i - \widehat{\mathbf{Q}}_i) (\lambda^* \dot{\mathbf{A}}^* - \lambda \dot{\mathbf{A}}) + (\omega^* \dot{\mathbf{V}}^* - \omega \dot{\mathbf{V}})^\top ((\mathbf{Q}_M)_i - (\widehat{\mathbf{Q}}_M)_i) (\omega^* \dot{\mathbf{V}}^* - \omega \dot{\mathbf{V}})}_{\beta_{213}}.\end{aligned}$$

The expression of  $\beta_{211}$  can be written as

$$\begin{aligned}|\beta_{211}| &= |\lambda^{*2} \sum_{jk} (\widehat{\mathcal{E}}_{i,j,k}^2 - \mathcal{E}_{i,j,k}) \delta_{i,j,k} \dot{\mathbf{A}}_{(j,k)}^{*2} + \omega^{*2} \sum_l ((\widehat{\mathcal{E}}_M^2)_{i,l} - (\mathcal{E}_m)_{i,l}) \dot{\mathbf{V}}_l^{*2}| \\ &\leq \lambda^{*2} \max_{i,j,k} |\widehat{\mathcal{E}}_{i,j,k}^2 - \mathcal{E}_{i,j,k}| \sum_{jk} \delta_{i,j,k} \dot{\mathbf{A}}_{(j,k)}^{*2} + \omega^{*2} \max_{i,l} |(\widehat{\mathcal{E}}_M^2)_{i,l} - (\mathcal{E}_m)_{i,l}| \sum_l \dot{\mathbf{V}}_l^{*2}.\end{aligned}\quad (4.63)$$

Note that since  $\max_{i,j,k} |\mathcal{T}_{i,j,k}^{obs} - \mathcal{T}_{i,j,k} - \mathcal{E}_{i,j,k}| \leq \|\mathcal{T} - \mathcal{T}^*\|_\infty$  and  $\max_{i,l} |\mathbf{M}_{i,l}^{obs} - \mathbf{M}_{i,l} - (\mathcal{E}_M)_{i,l}| \leq \|\mathbf{M} - \mathbf{M}^*\|_\infty$ , it follows that

$$\max_{i,j,k} |\widehat{\mathcal{E}}_{i,j,k}^2 - \mathcal{E}_{i,j,k}| \leq \|\mathcal{T} - \mathcal{T}^*\|_\infty \quad \text{and} \quad \max_{i,l} |(\widehat{\mathcal{E}}_M^2)_{i,l} - (\mathcal{E}_m)_{i,l}| \leq \|\mathbf{M} - \mathbf{M}^*\|_\infty. \quad (4.64)$$

Also from the general results of Sub-Gaussian theory [68], we know that

$$\|\mathcal{E}\|_\infty \leq \sigma_{\max} \sqrt{\log(n)} \quad \text{and} \quad \|\mathcal{E}_M\|_\infty \leq (\sigma_M)_{\max} \sqrt{\log(n)}, \quad (4.65)$$

with probability  $1 - n^{-11}$ . Next we have  $|(\delta_{jk} - p) \dot{\mathbf{A}}_{(j,k)}^{*2}| \leq (1 - p)(\mu/\sqrt{n})^4$

and  $\mathbb{E}(\delta_{jk} - p) \sum_{j,k} \dot{\mathbf{A}}_{(j,k)}^{*4} \leq p(1 - p)(\mu/\sqrt{n})^4$ . Using Bernstein inequality we get that

$$\left| \sum_{jk} \delta_{i,j,k} \dot{\mathbf{A}}_{(j,k)}^{*2} \right| \leq \frac{p + \sqrt{p(1 - p)(\mu/\sqrt{n})^4 \log(n)} + (1 - p)(\mu/\sqrt{n})^4 \log(n)}{(\lambda^{*2}p + \omega^{*2})^2} \leq \frac{p(1 + \frac{\mu^4 \log(n)}{n^2})}{(\lambda^{*2}p + \omega^{*2})^2}, \quad (4.66)$$

with probability  $1 - n^{-11}$ . Combining the results of (4.63) - (4.66) and using Bernstein inequality to bound  $\sum_{jk} \delta_{i,j,k} \dot{\mathbf{A}}_{(j,k)}^{*2}$  we get

$$|\beta_{211}| \leq \frac{\lambda^{*2} \sigma_{max} \sqrt{\log(n)} \|\mathcal{T} - \mathcal{T}^*\|_\infty * p(1 + o(1)) + \omega^{*2} (\sigma_M)_{max} \sqrt{\log(n)} \|\mathbf{M} - \mathbf{M}^*\|_\infty}{(\lambda^2 p + \omega^2)^2}, \quad (4.67)$$

with probability  $1 - n^{-11}$  and where the inequality hold for  $(\lambda^{*2} p + \omega^{*2})^2 \leq \lambda^2 \frac{\mu^4 \log(n)}{n^2}$ .

Next we can express  $\beta_{212}$  as

$$\begin{aligned} |\beta_{212}| &\leq |\lambda^* \sum_{jk} (\hat{\mathcal{E}}_{i,j,k}^2 - \mathcal{E}_{i,j,k}) \delta_{i,j,k} \dot{\mathbf{A}}_{(j,k)}^* (\lambda^* \dot{\mathbf{A}}^* - \lambda \dot{\mathbf{A}})_{(j,k)}| + \\ &\quad |\omega^* \sum_l ((\hat{\mathcal{E}}_M^2)_{i,l} - (\mathcal{E}_m)_{i,l}) \dot{\mathbf{V}}_l^* (\omega^* \dot{\mathbf{V}}^* - \omega \dot{\mathbf{V}})_l| \\ &\leq \|\mathcal{T} - \mathcal{T}^*\|_\infty \sum_{jk} \delta_{i,j,k} \dot{\mathbf{A}}_{(j,k)}^* (\lambda^* \dot{\mathbf{A}}^* - \lambda \dot{\mathbf{A}})_{(j,k)} + \|\mathbf{M} - \mathbf{M}^*\|_\infty \sum_l \dot{\mathbf{V}}_l^* (\omega^* \dot{\mathbf{V}}^* - \omega \dot{\mathbf{V}})_l \\ &\leq \frac{\lambda^{*2} \sigma_{max} \sqrt{\log(n)} \|\mathcal{T} - \mathcal{T}^*\|_\infty \|\mathbf{a} - \mathbf{a}^*\|_2 * p + \omega^{*2} (\sigma_M)_{max} \sqrt{\log(n)} \|\mathbf{M} - \mathbf{M}^*\|_\infty \|\mathbf{v} - \mathbf{v}^*\|_2}{(\lambda^2 p + \omega^2)^2}, \end{aligned} \quad (4.68)$$

with probability  $1 - n^{-11}$  and where the last inequality is obtained by applying the Bernstein inequality to bound  $\sum_{jk} \delta_{i,j,k} \dot{\mathbf{A}}_{(j,k)}^* (\lambda^* \dot{\mathbf{A}}^* - \lambda \dot{\mathbf{A}})_{(j,k)}$  and the fact that  $\langle \mathbf{b}^*, \mathbf{b} - \mathbf{b}^* \rangle + \langle \mathbf{c}^*, \mathbf{c} - \mathbf{c}^* \rangle + \langle \mathbf{b} - \mathbf{b}^*, \mathbf{c} - \mathbf{c}^* \rangle \leq 3 \max_{\mathbf{u}=\{\mathbf{a}, \mathbf{b}, \mathbf{c}\}} \|\mathbf{u} - \mathbf{u}^*\|$  and we assumed for ease of notation that  $\|\mathbf{a} - \mathbf{a}^*\| = \max_{\mathbf{u}=\{\mathbf{a}, \mathbf{b}, \mathbf{c}\}} \|\mathbf{u} - \mathbf{u}^*\|$ .

Using a similar method as the one used to bound  $|\beta_{212}|$  we get that

$$|\beta_{123}| \leq \frac{\lambda^{*2} \sigma_{max} \sqrt{\log(n)} \|\mathcal{T} - \mathcal{T}^*\|_\infty \|\mathbf{a} - \mathbf{a}^*\|_2^2 * p + \omega^{*2} (\sigma_M)_{max} \sqrt{\log(n)} \|\mathbf{M} - \mathbf{M}^*\|_\infty \|\mathbf{v} - \mathbf{v}^*\|_2^2}{(\lambda^2 p + \omega^2)^2}. \quad (4.69)$$

Combining the bounds on  $\beta_{211}, \beta_{212}$  and  $\beta_{213}$  in (4.67), (4.68) and (4.69) respectively we get

$$|\beta_{21}| \leq \frac{\lambda^{*2} \sigma_{max} \sqrt{\log(n)} \|\mathcal{T} - \mathcal{T}^*\|_{\infty p} (1 + \|\mathbf{a} - \mathbf{a}^*\|_2 + \|\mathbf{a} - \mathbf{a}^*\|_2^2)}{(\lambda^{*2} p + \omega^{*2})^2} + \frac{\omega^{*2} (\sigma_M)_{max} \sqrt{\log(n)} \|\mathbf{M} - \mathbf{M}^*\|_{\infty} (1 + \|\mathbf{v} - \mathbf{v}^*\|_2 + \|\mathbf{v} - \mathbf{v}^*\|_2^2)}{(\lambda^{*2} p + \omega^{*2})^2}. \quad (4.70)$$

**Bounding  $\beta_{22}$ :** We start by decomposing the expression of  $\beta_{22}$  as follows,

$$\begin{aligned} |\beta_{22}| &= |\lambda^2 \dot{\mathbf{A}}^\top \widehat{\mathbf{Q}}_i \dot{\mathbf{A}} - \lambda^{*2} \dot{\mathbf{A}}^{*\top} \mathbf{Q}_i^* \dot{\mathbf{A}}^* + \omega^2 \dot{\mathbf{V}}^\top (\widehat{\mathbf{Q}}_M)_i \dot{\mathbf{V}} - \omega^{*2} \dot{\mathbf{V}}^{*\top} (\mathbf{Q}_M^*)_i \dot{\mathbf{V}}^*| \\ &= 2 \left( \underbrace{\lambda^* \dot{\mathbf{A}}^{*\top} \widehat{\mathbf{Q}}_i (\lambda \dot{\mathbf{A}} - \lambda^* \dot{\mathbf{A}}^*) + \omega^* \dot{\mathbf{V}}^{*\top} (\widehat{\mathbf{Q}}_M)_i (\omega \dot{\mathbf{V}} - \omega^* \dot{\mathbf{V}}^*)}_{\beta_{221}} \right) + \\ &\quad \underbrace{(\lambda \dot{\mathbf{A}} - \lambda^* \dot{\mathbf{A}}^*)^\top \widehat{\mathbf{Q}}_i (\lambda \dot{\mathbf{A}} - \lambda^* \dot{\mathbf{A}}^*) + (\omega \dot{\mathbf{V}} - \omega^* \dot{\mathbf{V}}^*)^\top (\widehat{\mathbf{Q}}_M)_i (\omega \dot{\mathbf{V}} - \omega^* \dot{\mathbf{V}}^*)}_{\beta_{222}} + \\ &\quad \underbrace{\lambda^{*2} \dot{\mathbf{A}}^{*\top} (\widehat{\mathbf{Q}}_i - \mathbf{Q}_i^*) \dot{\mathbf{A}}^{*\top} + \omega^{*2} \dot{\mathbf{V}}^{*\top} ((\widehat{\mathbf{Q}}_M)_i - (\mathbf{Q}_M^*)_i) \dot{\mathbf{V}}^{*\top}}_{\beta_{223}} \end{aligned}$$

We bound the expressions  $\beta_{221}$ ,  $\beta_{222}$  and  $\beta_{223}$  in what follows. We invoke the Cauchy-Schwartz inequality to bound the first expression

$$\begin{aligned} |\beta_{221}| &= |\lambda^* \sum_{j,k} \mathcal{E}_{i,j,k}^2 \delta_{i,j,k} \dot{\mathbf{A}}_{(jk)}^* (\lambda \dot{\mathbf{A}} - \lambda^* \dot{\mathbf{A}}^*)_{(jk)} + \omega^* \sum_l (\mathcal{E}_M^2)_{i,l} \dot{\mathbf{V}}_l^* (\omega \dot{\mathbf{V}} - \omega^* \dot{\mathbf{V}}^*)_l| \\ &\leq \lambda^{*2} \|\mathbf{a} - \mathbf{a}^*\|_2 \sum_{j,k} \mathcal{E}_{i,j,k}^2 \delta_{i,j,k} \dot{\mathbf{A}}_{(jk)}^* + \omega^{*2} \|\mathbf{v} - \mathbf{v}^*\|_2 \sum_l (\mathcal{E}_M^2)_{i,l} \dot{\mathbf{V}}_l^* \\ &\leq \frac{\lambda^{*2} \|\mathbf{a} - \mathbf{a}^*\|_2 (\mu^2 p n \sigma_{max}^2 + \mu^2 \log(n)/n) + \omega^{*2} \|\mathbf{v} - \mathbf{v}^*\|_2 (\mu \sqrt{n} (\sigma_M)_{max}^2 + \mu^2 (\log(n)/\sqrt{n}))}{(\lambda^{*2} p + \omega^{*2})^2} \\ &\leq \frac{\lambda^{*2} \|\mathbf{a} - \mathbf{a}^*\|_2 (\mu^2 p n \sigma_{max}^2 + o(1)) + \omega^{*2} \|\mathbf{v} - \mathbf{v}^*\|_2 (\mu \sqrt{n} (\sigma_M)_{max}^2 + o(1))}{(\lambda^{*2} p + \omega^{*2})^2} \end{aligned} \quad (4.71)$$

where the second inequality is obtained by using the inequality  $\|\lambda \mathbf{a} - \lambda^* \mathbf{a}^*\|_2 = \lambda^* \|\mathbf{a} - \mathbf{a}^*\|_2 + o(1)$  and the third inequality is obtained by noting that

$$\max \|\mathcal{E}_{i,j,k}^2 \delta_{i,j,k} \dot{\mathbf{A}}_{(jk)}^*\|_{\phi_1} \leq \frac{\sigma_{max}^2 (\mu/\sqrt{n})^2}{(\lambda^{*2} p + \omega^{*2})^2} \quad \text{and} \quad \mathbb{E} \left( \sum_{j,k} \mathcal{E}_{i,j,k}^4 \delta_{i,j,k}^2 \dot{\mathbf{A}}_{(jk)}^{*2} \right) \leq \frac{p \sigma_{max}^4}{(\lambda^{*2} p + \omega^{*2})^4},$$

and  $\|\mathcal{E}\|_\infty \leq \sigma_{max}\sqrt{\log(n)}$  with high probability, followed by applying Bernstein inequality. The third and fourth inequality holds with probability  $1 - n^{-11}$ . In a similar manner we get the following bounds for  $\beta_{222}$  and  $\beta_{223}$ .

$$|\beta_{222}| \leq \frac{\lambda^{*2}\|\mathbf{a} - \mathbf{a}^*\|_2^2 p \sigma_{max}^2 \log(n) + \omega^{*2}\|\mathbf{v} - \mathbf{v}^*\|_2^2 (\sigma_M^2)_{max} \log(n) + o(1)}{(\lambda^{*2}p + \omega^{*2})^2}, \quad (4.72)$$

$$|\beta_{223}| \leq \frac{\lambda^{*2}\sigma_{max}^2(\mu^4(\log(n)/pn^2) + \sqrt{\mu^4(\log(n)/pn^2)}) + \omega^{*2}(\sigma_M^2)_{max}(\mu^2(\log(n)/n) + \sqrt{\mu^2(\log(n)/n)})}{(\lambda^{*2}p + \omega^{*2})^2}, \quad (4.73)$$

Combining the results in (4.71), (4.72), (4.73), we get the bound on  $\beta_{22}$  to be

$$|\beta_{22}| \leq \frac{\lambda^{*2}\sigma_{max}^2(\|\mathbf{a} - \mathbf{a}^*\|_2 \mu^2 pn + \|\mathbf{a} - \mathbf{a}^*\|_2^2 p \log(n) + (\mu^4(\log(n)/pn^2) + \sqrt{\mu^4(\log(n)/pn^2)}))}{(\lambda^{*2}p + \omega^{*2})^2} + \frac{\omega^{*2}(\sigma_M^2)_{max}(\|\mathbf{v} - \mathbf{v}^*\|_2(\mu\sqrt{n} + \|\mathbf{v} - \mathbf{v}^*\|_2^2 \log(n) + (\mu^2(\log(n)/n) + \sqrt{\mu^2(\log(n)/n)}))}{(\lambda^{*2}p + \omega^{*2})^2}. \quad (4.74)$$

Finally combining the bounds on  $\beta_{12}, \beta_{12}, \beta_{12}$  and simplifying the expression yields the following bound on  $|\Sigma_i - \Sigma_i^*|$

$$|\Sigma_i - \Sigma_i^*| \leq \Sigma_i^* o(1). \quad (4.75)$$

With the bound on  $|\Sigma_i - \Sigma_i^*|$  established we then finish the proof of the lemma by combining the results of (4.75), (4.59), (4.60), (4.61) and (4.62), which completes the proof of the Lemma.  $\square$

## 4.6 Auxillary Lemmas

In this section we state and prove a series of helper lemmas.

**Lemma 20.** *Let  $\mathbf{u}$  and  $\mathbf{w}$  be unit vectors in  $\mathbb{R}^n$  such that  $|\mathbf{u}(i)| \leq \frac{\mu}{\sqrt{d}}$  and  $|\mathbf{w}(j)| \leq \frac{\beta}{\sqrt{d}}$ . Also let  $\delta_{i,j,k}$  be i.i.d. Bernoulli random variables with  $P(\delta_{i,j,k} = 1) = p$  and  $1 \leq i \leq n, 1 \leq j \leq n,$*



$1 \leq k \leq n$ .

Then provided  $p \geq \frac{C\mu^2\beta^2(1+\gamma/3)\log(d^{10})}{d^2\gamma^2}$  we have

$$| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^2(j) \mathbf{w}^2(k) | \leq p \langle \mathbf{u}, \mathbf{u} \rangle \langle \mathbf{w}, \mathbf{w} \rangle - p\gamma,$$

with probability  $1 - d^{-10}$ .

**Proof:** Let  $X_{jk} = \frac{1}{p} (\delta_{i,j,k} \mathbf{u}^2(j) \mathbf{w}^2(k) - E(\delta_{i,j,k} \mathbf{u}^2(j) \mathbf{w}^2(k)))$ . Using the bound on the elements of  $\mathbf{u}$  and  $\mathbf{w}$ , we have  $|X_{jk}| = |\frac{1}{p} (\delta_{i,j,k} - p) \mathbf{u}^2(j) \mathbf{w}^2(k)| \leq \frac{\mu^2 \beta^2}{pd^2}$ . Also

$$\sum_{j,k} E[X_{jk}^2] = \frac{1}{p} (1-p) \sum_{j,k} \mathbf{u}^4(j) \mathbf{w}^4(k) \leq \frac{\mu^2 \beta^2}{pd^2}.$$

Applying Bernstein tail bound inequality we get:

$$P \left( \left| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^2(j) \mathbf{w}^2(k) - p \langle \mathbf{u}, \mathbf{u} \rangle \langle \mathbf{w}, \mathbf{w} \rangle \right| \geq pt \right) \leq \exp \left( \frac{-d^2 pt^2 / 2}{\mu^2 \beta^2 (1 + \frac{1}{3}t)} \right).$$

Setting the right side of the inequality to be less than  $q$  yields:

$$P \left( \left| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^2(j) \mathbf{w}^2(k) \right| \leq p \langle \mathbf{u}, \mathbf{u} \rangle \langle \mathbf{w}, \mathbf{w} \rangle - p\gamma \right) \geq 1 - q,$$

for  $p \geq \frac{\mu^2 \beta^2 (1+\gamma/3) \log(1/q)}{n^2 \gamma^2}$ . Choosing  $q \leq n^{-10}$  completes the proof of Lemma 20.  $\square$

**Lemma 21.** Let  $\mathbf{u}^*$ ,  $\mathbf{u}$  and  $\mathbf{w}$  be unit vectors in  $\mathbb{R}^n$  such that  $|\mathbf{u}_i^*| \leq \frac{\mu}{\sqrt{d}}$ ,  $|\mathbf{u}|$  and  $|\mathbf{w}| \leq \frac{\beta}{\sqrt{d}}$ . Let  $\mathbf{d}$  be another vector with  $\|\mathbf{d}\|_2 \leq 1$ . Also let  $\delta_{i,j,k}$  be i.i.d. Bernoulli random variables with  $P(\delta_{i,j,k} = 1) = p$  and  $1 \leq i \leq n$ ,  $1 \leq j \leq n$ ,  $1 \leq k \leq n$ . Provided  $p \geq \frac{C\mu\beta^2(1+\gamma/3)\log^2(\frac{1}{2}n^{10})}{n^{3/2}\gamma^2}$ , with probability greater than  $1 - 2n^{-10}$ , we have

$$| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - p \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{d}, \mathbf{w} \rangle | \leq p\gamma \|\mathbf{d}\|_2.$$

**Proof:** Let  $X_{jk} = \frac{1}{p} (\delta_{i,j,k} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - E(\delta_{i,j,k} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k)))$ . Then we have That is  $|X_{jk}| = \frac{1}{p} (\delta_{i,j,k} - p) \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) \leq \frac{1}{p} (1-p) \frac{\mu \beta^2}{d^{3/2}} \|\mathbf{d}\|_2$ . Also,

$$\sum_{j,k} E[X_{jk}^2] = \frac{1}{p} \sum_{j,k} (\mathbf{u}(j)^2 \mathbf{d}(k)^2 \mathbf{u}(j)^2 \mathbf{w}(k)^2) \leq \frac{\mu \beta^2 \|\mathbf{d}\|_2^2}{p d^{3/2}}.$$

Applying Bernstein tail bound inequality we get:

$$P \left( \left| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - p \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{d}, \mathbf{w} \rangle \right| \geq pt \right) \leq 2 \exp \left( \frac{-d^{3/2} p t^2}{\mu \beta^2 \|\mathbf{d}\|_2 (\|\mathbf{d}\|_2 + \frac{1}{3} t)} \right). \quad (4.76)$$

Setting the right side of the inequality to be less than  $q$  and choosing  $t \leq \gamma \|\mathbf{d}\|_2$  then solving for  $p$  yields:

$$P \left( \left| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - p \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{d}, \mathbf{w} \rangle \right| \leq p \gamma \|\mathbf{d}\|_2 \right) \geq 1 - 2q,$$

for  $p \geq \frac{\mu \beta^2 (1+\gamma/3) \log(\frac{1}{q})}{d^{3/2} \gamma^2}$ . Choosing  $q \leq n_{-10}$  completes the proof of Lemma 21.  $\square$

**Lemma 22.** Let  $\mathbf{u}^*$ ,  $\mathbf{w}^*$ ,  $\mathbf{u}$  and  $\mathbf{w}$  be unit vectors in  $\mathbb{R}^n$  such that  $|\mathbf{u}^*(i)|$  and  $|\mathbf{w}^*(j)| \leq \frac{\mu}{\sqrt{d}}$ ,  $|\mathbf{u}_i|$  and  $|\mathbf{w}_i| \leq \frac{\beta}{\sqrt{d}}$ . Let  $\delta_{i,j,k}$  be i.i.d. Bernoulli random variables with  $P(\delta_{i,j,k} = 1) = p$  and  $1 \leq i, j, k \leq n$ . Provided  $p \geq \frac{C \mu^2 \beta^2 (1+\gamma/3) \log(\frac{1}{2} d^{10})}{d^2 \gamma^2}$ , with probability greater than  $1 - 2n^{-10}$ , we have

$$\left| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k) \right| \leq p |\langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{w}^*, \mathbf{w} \rangle| + p \gamma.$$

**Proof:** Let  $X_{jk} = \frac{1}{p} (\delta_{i,j,k} \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k) - E(\delta_{i,j,k} \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k)))$ . Then we have  $|X_{jk}| = \frac{1}{p} (\delta_{i,j,k} - p) \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k) \leq \frac{1}{p} (1-p) \frac{\mu^2 \beta^2}{d^2}$ . Also

$$\sum_{j,k} E[X_{jk}^2] = \frac{1}{p} (1-p) \sum_{j,k} (\mathbf{u}(j)^2 \mathbf{w}(k)^2 \mathbf{u}(j)^2 \mathbf{w}(k)^2) \leq \frac{1}{p} (1-p) \frac{\mu^2 \beta^2}{d^2}.$$

Applying Bernstein tail bound inequality we get:

$$P \left( \left| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^*(j) \mathbf{d}(k) \mathbf{u}(j) \mathbf{w}(k) - p \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{d}, \mathbf{w} \rangle \right| \geq pt \right) \leq 2 \exp \left( \frac{-d^2 p t^2}{\mu^2 \beta^2 (1-p) (1 + \frac{1}{3} t)} \right).$$

Setting the right side of the inequality to be less than  $q$  and choosing  $t \leq \gamma$  then solving for  $p$  yields:

$$P \left( \left| \sum_{j,k} \delta_{i,j,k} \mathbf{u}^*(j) \mathbf{w}^*(k) \mathbf{u}(j) \mathbf{w}(k) - \langle \mathbf{u}^*, \mathbf{u} \rangle \langle \mathbf{w}^*, \mathbf{w} \rangle \right| \leq p\gamma \right) \geq 1 - 2q,$$

and  $p \geq \frac{\mu^2 \beta^2 (1+\gamma/3) \log(\frac{1}{q})}{d^2 \gamma^2}$ . Letting  $q \leq n^{-10}$  completes the proof of Lemma 22.  $\square$

**Lemma 23.** *Let  $\mathbf{u}$  and  $\mathbf{w}$  be unit vectors and let  $\mathbf{d}$  be a vector such that  $\mathbf{d} = \mathbf{u} - \mathbf{w}$  then*

$$|\langle \mathbf{w}, \mathbf{d} \rangle| = \frac{1}{2} \|\mathbf{d}\|_2^2.$$

**Proof:** Note that  $\|\mathbf{u}\|_2^2 = \sum (\mathbf{w}(i) + \mathbf{d}(i))^2$ . Hence given that  $\mathbf{u}$  is a unit vector we get

$$\begin{aligned} \sum \mathbf{w}(i)^2 + 2 \sum \mathbf{w}(i) \mathbf{d}(i) + \sum \mathbf{d}(i)^2 &= 1 \\ 2 \sum \mathbf{w}(i) \mathbf{d}(i) + \sum \mathbf{d}(i)^2 &= 0 \\ 2 \sum \mathbf{w}(i) \mathbf{d}(i) &= - \sum \mathbf{d}(i)^2 \\ |\langle \mathbf{w}, \mathbf{d} \rangle| &= \frac{1}{2} \|\mathbf{d}\|_2^2, \end{aligned}$$

Which completes the proof of the lemma.  $\square$

**Lemma 24.** *Let  $\mathbf{S}_i^*$  and  $\Sigma_i^*$  be as defined in (4.36) and (4.34). Given the assumptions in Lemma 14, with probability  $1 - n^{-10}$ , we have*

$$\max_n |\mathbf{S}_i^* - \Sigma_i^*| \leq \frac{\sqrt{2} \lambda^{*2} \sigma_{max}^{*2} (1-p) \frac{\mu^2}{n} \left(1 + \frac{\mu^2}{n} \log(n)\right)}{(\lambda^{*2} p + \omega^{*2})^2}. \quad (4.77)$$

**Proof:** Define  $\mathbf{F}_{jk}$  as,  $\mathbf{F}_{jk} := \sigma_{i,j,k}^{*2} (\delta_{i,j,k} - p) \mathbf{A}_{(jk)}^{*\top} \mathbf{A}_{(jk)}^*$ . Notice that  $|\mathbf{S}_i^* - \Sigma_i^*| = \lambda^{*2} \sum_{i,j,k} \mathbf{D}_{ii}^{*2} \mathbf{F}_{jk}$ . Using the u-mass condition we get,

$$L_1 := |\mathbf{F}_{jk}| \leq \sigma_{max}^{*2} (1-p) \frac{\mu^4}{n^2}$$

and

$$B_1 := \sum_{jk} E[\mathbf{F}_{jk}^2] \leq 2\sigma_{max}^{*4} p(1-p) \frac{\mu^4}{n^2}$$

Using Bernstein inequality we get

$$P\left(\frac{1}{\lambda^{*2}} |\mathbf{S}_i^* - \Sigma_i^*| \geq t\right) \leq \exp\left(\frac{-1/2t^2}{B_1 + 1/3L_1 t}\right)$$

with probability  $1 - n^{-11}$ . Using the results of Lemma 4.6 and as long as  $p < (1-p)$  and  $p$  meets the assumption provided in section 4.2 it follows then that with probability  $1 - n^{-11}$ .

$$|\mathbf{S}_i^* - \Sigma_i^*| \leq \frac{\sqrt{2}\lambda^{*2}\sigma_{max}^{*2}(1-p)\frac{\mu^2}{n}\left(1 + \frac{\mu^2}{n}\log(n)\right)}{(\lambda^{*2}p + \omega^{*2})^2}$$

applying the union bound completes the proof of the lemma.

**Lemma 25.** *Let  $\mathbf{Z}_i^*$  be as defined in (4.33). Given the assumptions of Lemma 14, with probability  $1 - n^{-10}$ , we have*

$$|\mathbf{Z}_i^*| \leq \frac{\lambda^* \sigma_{max} \left(\sqrt{2p} + 2C/3 \frac{\mu^2}{n} \log(n)\right)}{\lambda^{*2}p + \omega^{*2}}.$$

**Proof:** Recall the that  $\mathbf{Z}_i^* = \sum_{jk} z_{i,j,k}$  and  $z_{i,j,k} := \lambda^* \mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{A}_{(jk)}^* \mathbf{D}_{ii}^{*-1}$  for  $1 \leq i, j, k \leq n$ . Let  $\tilde{z}_{i,j,k} = \lambda^* \mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{A}_{(jk)}^*$ . Since  $\mathcal{E}_{i,j,k}$  and  $\delta_{i,j,k}$  are both sub-Gaussian random variables, it follows that  $\tilde{z}_{i,j,k}$  is a sub exponential random variable. Hence its sub exponential norm  $\psi_1$  can be bounded in the following way,

$$\begin{aligned} L_2 := \|\mathcal{E}_{i,j,k} \delta_{i,j,k} \mathbf{A}_{(jk)}^*\|_{\psi_1} &\leq \frac{\mu^2}{n} \|\mathcal{E}_{i,j,k}\|_{\psi_2} \|\delta_{i,j,k}\|_{\psi_2} \\ &\leq C \frac{\mu^2}{n} \sigma_{max}^* \|\delta_{i,j,k}\|_{\infty} \\ &\leq C \frac{\mu^2}{n} \sigma_{max}^*, \end{aligned}$$

where  $\|\cdot\|_{\psi_2}$ , represents the sub-Gaussian norm and  $C$  is an absolute constant. The second inequality above is obtained by using the  $\mu$ -mass condition on the tensor factor entries provided in Section 4.2 and also applying the property of sub-exponential and sub-Gaussian

norms namely that the sub exponential norm of the product of two sub-Gaussian random variables is less than the product of the sub-Gaussian norms of the two random variables. Also, since  $\delta_{i,j,k}$  and  $\mathcal{E}_{i,j,k}$  are independent we have

$$B_2 := \sum_{jk} \mathbb{E}(\mathcal{E}_{i,j,k}^2 \delta_{i,j,k}^2 \mathbf{A}_{(jk)}^{*2}) \leq \sigma_{max}^{*2} p.$$

The above inequality is obtained by applying the  $\mu$ -mass condition of the tensor factor entries. Using the Bernstein inequality and the results of Lemma 4.6 we get that with probability  $1 - n^{-11}$

$$|\mathbf{Z}_i^*| \leq \frac{\lambda^* \sigma_{max} \left( \sqrt{2p} + 2C/3 \frac{\mu^2}{n} \log(n) \right)}{\lambda^{*2} p + \omega^{*2}},$$

Which completes the proof of the lemma.  $\square$

**Lemma 26.** *Let  $\mathbf{S}_i^*$  be as defined as in (4.36). Then given the assumptions in Lemma 14, with probability  $1 - n^{-11}$  we have,*

$$\mathbf{S}_i^{*1/2} \geq \frac{\sqrt{\lambda^{*2} p \sigma_{min}^2 + \omega^{*2} (\sigma_M^{*2})_{min}}}{\lambda^{*2} p + \omega^{*2}}.$$

**Proof:** The expression of  $\mathbf{S}_i^*$  can be decomposed as

$$\mathbf{S}_i^* \geq \Sigma_i^* - |\mathbf{S}_i^* - \Sigma_i^*| \geq \frac{\lambda^{*2} p \sigma_{min}^2 + \omega^{*2} (\sigma_M^{*2})_{min}}{(\lambda^{*2} p + \omega^{*2})^2},$$

where the last inequality is obtained by applying the results of Lemma 24 as well as bounding  $\Sigma_i^*$  using the  $\mu$ -mass condition stated in section 4.2. Since  $\mathbf{S}_i^*$  is positive, applying the square root function to both side of the inequality yields the desired results for the lemma.  $\square$

**Lemma 27.** *Given the assumption in Lemma 15. Let  $P_\Omega(\mathcal{E}_T)_{i,:}$  and  $\mathbf{c}^{(i)}$  be as defined in (4.41), then with probability  $1 - n^{-11}$  we have*

$$\begin{aligned} \|P_\Omega(\mathcal{E}_T)_{i,:}\|_2 &\leq \sigma_{max}(\sqrt{pn} + \log n) \text{ and} \\ \|P_\Omega(\mathcal{E}_T)_{i,:}\mathbf{c}^{(i)}\|_2 &\leq \sigma_{max}(\sqrt{np} + \frac{\mu}{\sqrt{n}} \log n) \end{aligned}$$

**Proof:** Given the construction of  $\mathbf{c}^{(i)}$ , it follows that  $P_\Omega(\mathcal{E}_T)_{i,:}\mathbf{c}^{(i)}$  is a sum of independents random variables and can be written as  $P_\Omega(\mathcal{E}_T)_{i,:}\mathbf{c}^{(i)} = \sum_{jk} \mathcal{E}_{i,j,k} \delta_{i,j,k} e_j e_k^\top \mathbf{c}^{(i)}$ . We start by bounding the norms of these random variables. Using the u-mass assumption and property of exponential norm we get

$$\begin{aligned} L_6 &:= \max_{jk} \|\mathcal{E}_{i,j,k} \delta_{i,j,k}\|_{\psi_1} \leq \sigma_{max}^*, \\ L_7 &:= \max_{jk} \|\mathcal{E}_{i,j,k} \delta_{i,j,k} e_j e_k^\top \mathbf{c}^{(i)}\|_{\psi_1} \leq \sigma_{max}^* \|\mathbf{c}^{(i)}\|_\infty \leq \sigma_{max}^* \frac{\mu}{\sqrt{n}}, \\ B_6 &:= \sum_{jk} E(\mathcal{E}_{i,j,k}^2 \delta_{i,j,k}^2) \leq \sigma_{max}^2 n^2 p \\ B_7 &:= \sum_{jk} E(\mathcal{E}_{i,j,k}^2 \delta_{i,j,k}^2 \mathbf{c}_k^{(i)2}) \leq \sigma_{max}^2 np \end{aligned}$$

Hence by Bernstein inequality we have with probability  $1 - n^{-11}$

$$\begin{aligned} \|P_\Omega(\mathcal{E}_T)_{i,:}\|_2 &\leq \sigma_{max}(\sqrt{pn} + \log n) \text{ and} \\ \|P_\Omega(\mathcal{E}_T)_{i,:}\mathbf{c}^{(i)}\|_2 &\leq \sigma_{max}(\sqrt{np} + \frac{\mu}{\sqrt{n}} \log n) \end{aligned}$$

## 4.7 Additional Material

For convenience this section contain a list of TheoremS and mathematical derivations which were used in the proofs of the paper. We refer the reader to the original papers for the proof of these results.

### 4.7.1 Lyapunov-type Bound

In the proof of Theorem 4.2.2 we use make use of the following Lyapunov-type bound results which allows us to explicitly quantify the rate of convergence of the central limit theorem. This done by establishing a bound on the maximal approximation error the normal distribution and the true distribution of the sum of random variables. Al though in Theorem 4.2.2 the random variables of interest live in one dimension we still make use of the d-dimensional version of the theorem as a way to allow us to extent our results to high dimension in the future.

**Theorem 4.7.1.** *Let  $\{\mathbf{x}_i\}_{1 \leq i \leq n}$  be a sequence of independent zero-mean random vectors in  $\mathbb{R}^d$  and let  $\Sigma$  be the covariance matrix of  $\sum_{1 \leq i \leq n} \mathbf{x}_i$  and  $\mathbf{z} \sim N(0, \Sigma)$  be a Gaussian vector in  $\mathbb{R}^d$ . Then we have,*

$$\sup_{\mathcal{A} \in \mathcal{C}} |\mathbb{P}\{\sum_{1 \leq i \leq n} \mathbf{x}_i \in \mathcal{A}\} - \mathbb{P}\{\mathbf{z} \in \mathcal{A}\}| \leq d^{1/4} \rho, \quad (4.78)$$

where  $\mathcal{C}$  is the set of all convex subset of  $\mathbb{R}^d$  and  $\rho$  is defined as follows,

$$\rho := \sum_{1 \leq i \leq n} \mathbb{E}[\|\Sigma^{-1/2} \mathbf{x}_i\|_2^3]. \quad (4.79)$$

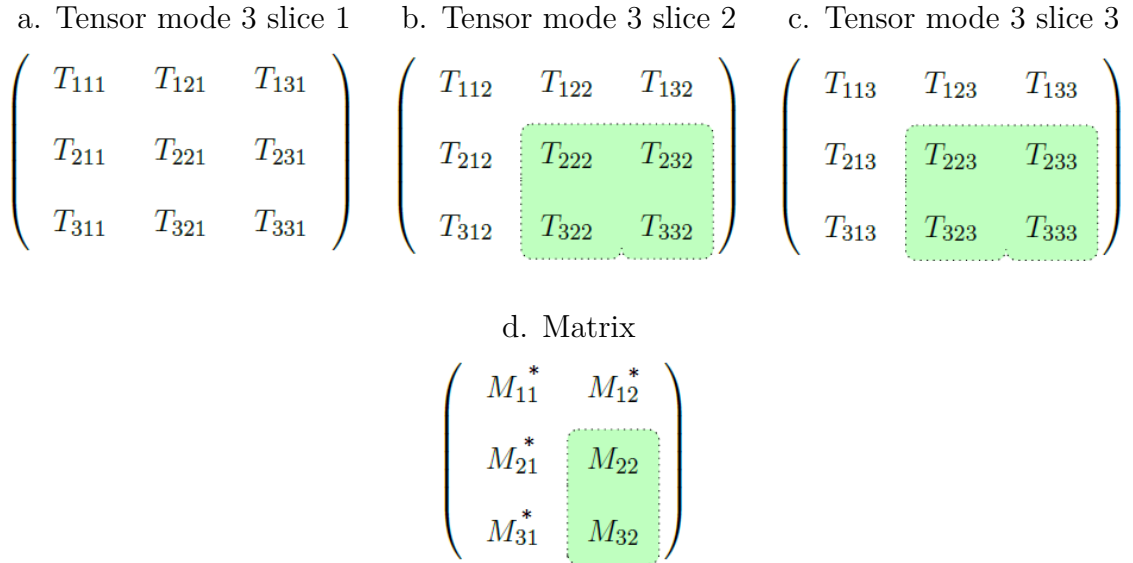
### 4.7.2 The Leave-One-Out Method

In this section we present the details of the leave one out method which is used in the proofs of various lemmas in the paper. We follow the procedure presented in Cai, Poor, and Chen [47] but adjust their method to the case of the non-symmetric tensor coupled to a matrix. This procedure is used in order to decoupled a given slice of the tensor and matrix from the noise tensor and noise matrix, allowing us therefore to partially remove the

statistical dependency that exists between the error tensor and error matrix with a given tensor factor. We present the method based on the recovered of tensor component  $\mathbf{a}$ .

$$\begin{aligned}\mathcal{T}^{(m)} &:= P_{\Omega-m}(\mathcal{T}) + pP_m(\mathcal{T}^*); & \mathbf{M}_{\mathbf{a}}^{(m)} &:= P_{-m}(\mathbf{M}_{\mathbf{a}}) + P_m(\mathbf{M}_{\mathbf{a}}^*) \\ \mathbf{M}_{\mathbf{b}}^{(m)} &:= P_{-m}(\mathbf{M}_{\mathbf{b}}) + P_m(\mathbf{M}_{\mathbf{b}}^*); & \mathbf{M}_{\mathbf{c}}^{(i)} &:= P_{-m}(\mathbf{M}_{\mathbf{c}}) + P_m(\mathbf{M}_{\mathbf{c}}^*),\end{aligned}$$

where  $P_{\Omega-m}$  and  $P_{-m}$  is the projection of the tensor onto the set  $\{(i, j, k) \in \Omega : i \neq m \text{ and } j \neq m \text{ and } k \neq m\}$  and the set  $\{(i, l) : i \neq m \text{ and } l \neq m\}$  respectively and  $P_m$  is there complement sets. This means that the  $m$ -th slice along all three modes is independent tot he noise tensor and noise matrices We use an example to illustrate the impact of this decomposition on the tensor factor and tensor noise.



**Figure 4.6.** Illustration of the leave one out procedure applied to a third order tensor and a matrix. The tensor was sliced along the third mode. The green shades values represent tensor and matrix entries which still contain some noise. Whereas the non shades values are replaced by the true value of tensor. The missing probability is set to  $p = 1$  for the sake of the illustration.



## 5. CONCLUDING REMARKS

Tensor completion is a popular subject of theoretical and applied study in a wide range of research fields such as statistics, mathematics, computer science and engineering. The method offers great potential in application driven studies in computer vision, recommender systems, community detection, personalized medicine etc. Due to the natural representation of high-order interactions as tensor data it has been shown that tensor completion based algorithms such as recommender systems generally outperform their matrix based counterparts. However, in practice, high percents of missing entries and high sparsity levels often observed in real tensor data, forbid the use of those tensor completion based algorithms. This is the case for instance in online advertising where users click-through-rates tensors can have up to 96% missing entries in addition to being highly sparse, making standard tensor completion algorithms fail to produce reasonable recovery results.

We effectively addressed this issue in the first part of the dissertation, by proposing **COSTCO**, a framework for tensor completion, suited for very sparse tensors with high percent of missing entries. Our method leverage the power of extra information often present under the form of covariate data, beside the tensor data, to deliver ameliorated tensor recovery results. **COSTCO** uses a joint latent components extraction mechanism along with a truncation procedure to learn a synthetic representation of the tensor with missing entries and to enforce sparsity. The proposed method is easy to implement and general enough to be applied to tensors of various dimensions in the presence of any number of covariate matrices coupled along the tensor modes.

Theoretically, we showed that the error bound derived for the recovered components using **COSTCO** represents an improvement over known standard tensor completion methods, provided the noise levels in the covariate matrices do not dominate that of the tensor. We also showed that our method leads to a relaxation in the number of required tensor entries observed for the completion algorithm to work. The performance of our method was illustrated through several simulation studies which revealed **COSTCO** to outperform the state of the art tensor completion methods. It was also demonstrated through a real data analysis on advertisement

data that our method can be used to boost the performance of other machine learning methods such as clustering.

The topic of uncertainty quantification in tensor completion is an area which lacks theoretical research contribution. Therefore, beside proposing a powerful completion algorithm for sparse and high missing tensors, in the second part of the dissertation we provided a theoretical analysis for the uncertainty quantification for the tensor components recovered using `COSTCO`. We focused our theoretical analysis to the case of the rank one tensor. Theoretically, we characterized the distribution of the recovered tensor components under various noise distribution assumptions and then proposed a simple yet valid confidence interval construction technique for the recovered tensor components. The proposed construction allows for heteroskedasticity in the tensor data entries and generate confidence intervals at an entry level for the tensor components. We also proved the validity of the constructed confidence intervals. We then showed through a series of simulation studies the improvement in tightness of these proposed confidence intervals compared to those generated using standard tensor completion methods.

Given the method proposed in the dissertation, numerous directions for further studies naturally arise. One such interesting direction is in extending our method to work for the online data setting. In fact, in real scenarios, tensor and/or covariate matrix information are rarely available all at once and are rather revealed sequentially. In this case, the batch updating algorithm proposed in this dissertation is no longer feasible as accessing the entire data for every added observation is impractical. To fill this gap, it would be advantageous to adapt our algorithm to the online setting. This could be done by utilizing stochastic gradient decent (SGD) to solve the optimization problem proposed in our work by updating only one row of the latent component matrix at a time. The proof for the recovery results for such a method would then rely of non-convex SGD methods beside the proof techniques developed in this work.

Another avenue for extending the work in this dissertation is to study the uncertainty quantification of the recovered tensor components for the general rank  $R$  tensor case. Also bootstrap methods represent a very popular and often hassle free technique for constructing confidence intervals. It would be interesting to compared the properties of the confidence

intervals generated using our proposed construction technique which first characterises the distribution of the tensor components and engineers a variance estimation technique to the confidence intervals obtained through the resampling approach used under bootstrap methods.

## REFERENCES

- [1] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos, “Tag recommendations based on tensor dimensionality reduction,” 2008. DOI: [10.1145/1454008.1454017](https://doi.org/10.1145/1454008.1454017).
- [2] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver, “Multiverse recommendation: N-dimensional tensor factorization for context-aware collaborative filtering,” *ACM Recommender Systems*, 2010.
- [3] X. Bi, A. Qu, X. Shen, *et al.*, “Multilayer tensor factorization with applications to recommender systems,” *Annals of Statistics*, vol. 46, no. 6B, pp. 3308–3333, 2018.
- [4] H. Zhou, L. Li, and H. Zhu, “Tensor regression with applications in neuroimaging data analysis,” *Journal of the American Statistical Association*, vol. 108, no. 502, pp. 540–552, 2013.
- [5] X. Wang, H. Zhu, and A. D. N. Initiative, “Generalized scalar-on-image regression models via total variation,” *Journal of the American Statistical Association*, vol. 112, no. 519, pp. 1156–1168, 2017.
- [6] X. Tang, X. Bi, and A. Qu, “Individualized multilayer tensor learning with an application in imaging analysis,” *Journal of the American Statistical Association*, vol. 115, no. 530, pp. 836–851, 2020.
- [7] E. E. Papalexakis, C. Faloutsos, and N. D. Sidiropoulos, “Tensors for data mining and data fusion: Models, applications, and scalable algorithms,” *ACM Transactions on Intelligent Systems and Technology*, vol. 8, 2 2016, ISSN: 21576912. DOI: [10.1145/2915921](https://doi.org/10.1145/2915921).
- [8] N. D. Sidiropoulos, L. D. Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, “Tensor decomposition for signal processing and machine learning,” *IEEE Transactions on Signal Processing*, vol. 65, 13 2017, ISSN: 1053587X. DOI: [10.1109/TSP.2017.2690524](https://doi.org/10.1109/TSP.2017.2690524).
- [9] P. Hoff, “Multilinear tensor regression for longitudinal relational data,” *Ann. Appl. Stat.*, vol. 9, no. 3, pp. 1169–1193, 2015, ISSN: 1932-6157.
- [10] B.-Y. Jing, T. Li, Z. Lyu, and D. Xia, “Community detection on mixture multi-layer networks via regularized tensor decomposition,” *arXiv preprint arXiv:2002.04457*, 2020.
- [11] “Tensor factorization for precision medicine in heart failure with preserved ejection fraction,” *Journal of Cardiovascular Translational Research*, vol. 10, 3 2017, ISSN: 19375395. DOI: [10.1007/s12265-016-9727-8](https://doi.org/10.1007/s12265-016-9727-8).

- [12] H. Wang, Q. Zhang, F. Y. Chen, E. Y. Man Leung, E. L. Yi Wong, and E.-K. Yeoh, “Tensor factorization-based prediction with an application to estimating the risk of chronic diseases,” *bioRxiv*, 2019. DOI: [10.1101/810556](https://doi.org/10.1101/810556).
- [13] R. Chen, D. Yang, and C.-h. Zhang, “Factor models for high-dimensional tensor time series,” *arXiv preprint arXiv:1905.07530*, 2019.
- [14] Q. Song, H. Ge, J. Caverlee, and X. Hu, “Tensor completion algorithms in big data analytics,” *ACM Transactions on Knowledge Discovery from Data*, vol. 13, 1 2019, ISSN: 1556472X. DOI: [10.1145/3278607](https://doi.org/10.1145/3278607).
- [15] X. Bi, X. Tang, Y. Yuan, Y. Zhang, and A. Qu, “Tensors in statistics,” *Annual Review of Statistics and Its Application*, vol. 8, 2020.
- [16] D. Xia, M. Yuan, and C.-H. Zhang, “Statistically optimal and computationally efficient low rank tensor completion from noisy entries,” *Ann. Statist.*, vol. 49, no. 1, pp. 76–99, Feb. 2021. DOI: [10.1214/20-AOS1942](https://doi.org/10.1214/20-AOS1942).
- [17] C. Cai, G. Li, H. V. Poor, and Y. Chen, “Nonconvex low-rank tensor completion from noisy data,” in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alch e-Buc, E. Fox, and R. Garnett, Eds., vol. 32, Curran Associates, Inc., 2019, pp. 1863–1874.
- [18] A. Zhang, “Cross: Efficient low-rank tensor completion,” *Annals of Statistics*, vol. 47, 2 2019, ISSN: 00905364. DOI: [10.1214/18-AOS1694](https://doi.org/10.1214/18-AOS1694).
- [19] Q. Zhao, G. Zhou, L. Zhang, A. Cichocki, and S. I. Amari, “Bayesian robust tensor factorization for incomplete multiway data,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, 4 2016, ISSN: 21622388. DOI: [10.1109/TNNLS.2015.2423694](https://doi.org/10.1109/TNNLS.2015.2423694).
- [20] D. Goldfarb and Z. Qin, “Robust low-rank tensor recovery: Models and algorithms,” *SIAM Journal on Matrix Analysis and Applications*, vol. 35, 1 2014, ISSN: 10957162. DOI: [10.1137/130905010](https://doi.org/10.1137/130905010).
- [21] Y. Liu, Z. Long, H. Huang, and C. Zhu, “Low cp rank and tucker rank tensor completion for estimating missing components in image data,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, 4 2020, ISSN: 15582205. DOI: [10.1109/TCSVT.2019.2901311](https://doi.org/10.1109/TCSVT.2019.2901311).
- [22] Z. Zhang and S. Aeron, “Exact tensor completion using t-svd,” *IEEE Transactions on Signal Processing*, vol. 65, pp. 1511–1526, 2017.
- [23] B. Barak and A. Moitra, “Noisy tensor completion via the sum-of-squares hierarchy,” vol. 49, 2016.

- [24] P. Jain and S. Oh, “Provable tensor factorization with missing data,” in *Advances in Neural Information Processing Systems*, 2014, pp. 1431–1439.
- [25] J. Liu, P. Musialski, P. Wonka, and J. Ye, “Tensor completion for estimating missing values in visual data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 208–220, 2013. DOI: [10.1109/TPAMI.2012.39](https://doi.org/10.1109/TPAMI.2012.39).
- [26] S. Gandy, B. Recht, and I. Yamada, “Tensor completion and low-n-rank tensor recovery via convex optimization,” *Inverse Problems*, vol. 27, 2 2011, ISSN: 02665611. DOI: [10.1088/0266-5611/27/2/025010](https://doi.org/10.1088/0266-5611/27/2/025010).
- [27] A. Cichocki, D. Mandic, L. D. Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and H. A. Phan, *Tensor decompositions for signal processing applications: From two-way to multiway component analysis*, 2015. DOI: [10.1109/MSP.2013.2297439](https://doi.org/10.1109/MSP.2013.2297439).
- [28] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, “A survey of multilinear subspace learning for tensor data,” *Pattern Recognition*, vol. 44, 7 2011, ISSN: 00313203. DOI: [10.1016/j.patcog.2011.01.004](https://doi.org/10.1016/j.patcog.2011.01.004).
- [29] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.
- [30] J. Håstad, “Tensor rank is np-complete,” *Journal of Algorithms*, vol. 11, 4 1990, ISSN: 01966774. DOI: [10.1016/0196-6774\(90\)90014-6](https://doi.org/10.1016/0196-6774(90)90014-6).
- [31] G. Tomasi and R. Bro., “A comparison of algorithms for fitting the parafac model,” *Computational Statistics and Data Analysis*, vol. 50, pp. 1700–1734, 2006.
- [32] A. P. Singh and G. J. Gordon, “Relational learning via collective matrix factorization,” 2008. DOI: [10.1145/1401890.1401969](https://doi.org/10.1145/1401890.1401969).
- [33] A. K. Smilde, J. A. Westerhuis, and R. Boqué, “Multiway multiblock component and covariates regression models,” *Journal of Chemometrics*, vol. 14, 3 2000, ISSN: 08869383. DOI: [10.1002/1099-128X\(200005/06\)14:3<301::AID-CEM594>3.0.CO;2-H](https://doi.org/10.1002/1099-128X(200005/06)14:3<301::AID-CEM594>3.0.CO;2-H).
- [34] Y.-R. Lin, J. Sun, P. Castro, R. Konuru, H. Sundaram, and A. Kelliher, “Metafac: Community discovery via relational hypergraph factorization,” in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, 2009, pp. 527–536.
- [35] Y. Koren, R. Bell, C. Volinsky, *et al.*, “Matrix factorization techniques for recommender systems,” *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

- [36] E. Acar, T. G. Kolda, and D. M. Dunlavy, “All-at-once optimization for coupled matrix and tensor factorizations,” *arXiv preprint arXiv:1105.3422*, 2011.
- [37] E. Acar, M. A. Rasmussen, F. Savorani, T. Næs, and R. Bro, “Understanding data fusion within the framework of coupled matrix and tensor factorizations,” *Chemometrics and Intelligent Laboratory Systems*, vol. 129, pp. 53–63, 2013.
- [38] M. Haghighat, M. Abdel-Mottaleb, and W. Alhalabi, “Discriminant correlation analysis: Real-time feature level fusion for multimodal biometric recognition,” *IEEE*, 2016.
- [39] T. Zhou, H. Qian, Z. Shen, C. Zhang, and C. Xu, “Tensor completion with side information: A riemannian manifold approach,” in *IJCAI*, 2017, pp. 3539–3545.
- [40] L. Li, J. Kang, S. N. Lockhart, J. Adams, and W. J. Jagust, “Spatially adaptive varying correlation analysis for multimodal neuroimaging data,” *IEEE transactions on medical imaging*, vol. 38, no. 1, pp. 113–123, 2018.
- [41] W. Kishan, Y. Makoto, and M. Hiroshi, “Convex coupled matrix and tensor completion,” *arXiv preprint arXiv:1705.05197*, 2018.
- [42] D. Choi, J. G. Jang, and U. Kang, “S3cmtf: Fast, accurate, and scalable method for incomplete coupled matrix-tensor factorization,” *PLoS ONE*, vol. 14, 6 2019, ISSN: 19326203. DOI: [10.1371/journal.pone.0217316](https://doi.org/10.1371/journal.pone.0217316).
- [43] H. Huang, Y. Liu, and C. Zhu, “A unified framework for coupled tensor completion,” *arXiv preprint arXiv:2001.02810*, 2020.
- [44] L. Li, J. Zeng, and X. Zhang, “Generalized liquid association analysis for multimodal data integration,” *arXiv preprint arXiv:2008.03733*, 2020.
- [45] F. Xue and A. Qu, “Integrating multisource block-wise missing data in model selection,” *Journal of the American Statistical Association*, pp. 1–14, 2020.
- [46] D. Xia and M. Yuan, “On polynomial time methods for exact low-rank tensor completion,” *Foundations of Computational Mathematics*, vol. 19, 6 2019, ISSN: 16153383. DOI: [10.1007/s10208-018-09408-6](https://doi.org/10.1007/s10208-018-09408-6).
- [47] C. Cai, H. V. Poor, and Y. Chen, “Uncertainty quantification for nonconvex tensor completion: Confidence intervals, heteroscedasticity and optimality,” 2020. arXiv: [2006.08580 \[stat.ML\]](https://arxiv.org/abs/2006.08580).
- [48] D. Xia, A. R. Zhang, and Y. Zhou, “Inference for low-rank tensors – no need to debias,” 2020. arXiv: [2012.14844 \[math.ST\]](https://arxiv.org/abs/2012.14844).

- [49] W. W. Sun, J. Lu, H. Liu, and G. Cheng, “Provable sparse tensor decomposition,” *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, vol. 79, 3 2017, ISSN: 14679868. DOI: [10.1111/rssb.12190](https://doi.org/10.1111/rssb.12190).
- [50] A. Zhang and R. Han, “Optimal sparse singular value decomposition for high-dimensional high-order data,” *Journal of the American Statistical Association*, vol. 114, no. 528, pp. 1708–1725, 2019.
- [51] B. Hao, A. R. Zhang, and G. Cheng, “Sparse and low-rank tensor estimation via cubic sketchings,” in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, S. Chiappa and R. Calandra, Eds., ser. Proceedings of Machine Learning Research, vol. 108, PMLR, 26–28 Aug 2020, pp. 1319–1330.
- [52] Y. Pan, Q. Mai, and X. Zhang, “Covariate-adjusted tensor classification in high dimensions,” *Journal of the American Statistical Association*, vol. 114, no. 527, pp. 1305–1319, 2019.
- [53] L. Li and X. Zhang, “Parsimonious tensor response regression,” *Journal of the American Statistical Association*, vol. 112, no. 519, pp. 1131–1146, 2017.
- [54] D. Xia and M. Yuan, “Effective tensor sketching via sparsification,” *IEEE Transactions on Information Theory*, vol. 67, no. 2, pp. 1356–1369, 2021. DOI: [10.1109/TIT.2021.3049174](https://doi.org/10.1109/TIT.2021.3049174).
- [55] Z. Wang, Q. Gu, Y. Ning, and H. Liu, “High dimensional expectation-maximization algorithm: Statistical optimization and asymptotic normality,” *arXiv preprint arXiv:1412.8729*, 2014.
- [56] Z. Wang, H. Liu, and T. Zhang, “Optimal computational and statistical rates of convergence for sparse nonconvex learning problems,” *Annals of statistics*, vol. 42, no. 6, p. 2164, 2014.
- [57] A. Anandkumar, R. Ge, D. Hsu, S. M. Kakade, and M. Telgarsky, “Tensor decompositions for learning latent variable models,” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 2773–2832, 2014.
- [58] G. W. Stewart, “Perturbation theory for the singular value decomposition,” in *SVD and Signal Processing Part II: Algorithms Analysis and Applications*, 1990, pp. 99–109.
- [59] C. F. Ipsen, “Relative perturbation results for matrix eigenvalues and singular values,” *Acta Numerica* 7, pp. 151–201, 1998.
- [60] G. Allen, “Sparse higher-order principal components analysis,” in *International Conference on Artificial Intelligence and Statistics*, 2012.



- [61] A. Anandkumar, R. Ge, and M. Janzamin, “Guaranteed non-orthogonal tensor decomposition via alternating rank-1 updates,” *arXiv preprint arXiv:1402.5180*, 2014.
- [62] M. Yuan and C. H. Zhang, “Incoherent tensor norms and their applications in higher order tensor completion,” *IEEE Transactions on Information Theory*, vol. 63, no. 10, 2017, ISSN: 00189448. DOI: [10.1109/TIT.2017.2724549](https://doi.org/10.1109/TIT.2017.2724549).
- [63] A. Montanari and N. Sun, “Spectral algorithms for tensor completion,” *Communications on Pure and Applied Mathematics*, vol. 71, no. 11, 2018, ISSN: 10970312. DOI: [10.1002/cpa.21748](https://doi.org/10.1002/cpa.21748).
- [64] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.
- [65] R. Tomioka and T. Suzuki, “Spectral norm of random tensors,” 2014. arXiv: [1407.1870](https://arxiv.org/abs/1407.1870) [[math.ST](#)].
- [66] V. Bentkus, *A lyapunov-type bound in  $r$  d*, 2005. DOI: [10.1137/S0040585X97981123](https://doi.org/10.1137/S0040585X97981123).
- [67] A. Winkelbauer, “Moments and absolute moments of the normal distribution,” 2014. arXiv: [1209.4340](https://arxiv.org/abs/1209.4340) [[math.ST](#)].
- [68] P. Rigollet, “18.s997 high-dimensional statistics chapter 1. spring 2015,” *MIT OpenCourseWare: Massachusetts Institute of Technology*, 2015. [Online]. Available:

## VITA

Somnooma Hilda Marie Bernadette Ibriga was born in Ouagadougou, Burkina Faso. She holds a double bachelor degree in Mathematics and Economics from Westminster College and a Master's degree in Mathematics from the University of Arkansas. She pursued her Doctorate studies in the department of Statistics at Purdue University under the supervision of Dr. Wei Sun and Dr. Bruce Craig.