Learning Mixtures of Separable Dictionaries for Tensor Data: Analysis and Algorithms

Mohsen Ghassemi, *Student Member, IEEE,* Zahra Shakeri, *Member, IEEE,*
Anand D. Sarwate, *Senior Member, IEEE,* and Waheed U. Bajwa, *Senior Member, IEEE*

***Abstract*—This work addresses the problem of learning sparse representations of tensor data using structured dictionary learning. It proposes learning a mixture of separable dictionaries to better capture the structure of tensor data by generalizing the separable dictionary learning model. Two different approaches for learning mixture of separable dictionaries are explored and sufficient conditions for local identifiability of the underlying dictionary are derived in each case. Moreover, computational algorithms are developed to solve the problem of learning mixture of separable dictionaries in both batch and online settings. Numerical experiments are used to show the usefulness of the proposed model and the efficacy of the developed algorithms.**

***Index Terms*—Dictionary learning, Kronecker structure, sample complexity, separation rank, tensor rearrangement.**

I. Introduction

Many data processing tasks such as feature extraction, data compression, classification, signal denoising, image inpainting, and audio source separation use sparse representations learned from data [3]–[5]. In many cases, these applications also involve data samples that are naturally structured as multiway arrays, also known as multidimensional arrays or tensors. Instances of *multidimensional* or *tensor* data include videos, hyperspectral images, tomographic images, and multiple-antenna wireless channels. Despite the ubiquity of tensor data in many applications, traditional data-driven sparse representation approaches disregard their multidimensional structure. This can result in sparsifying models with a large number of parameters. With the increasing availability of large data sets, it is crucial to keep sparsifying models reasonably small to ensure their scalable learning and efficient storage within devices such as smartphones and drones.

Our focus in this paper is on learning of "compact" models that yield sparse representations of tensor data. To this end, we study *dictionary learning* (DL) for tensor data. The goal in DL, which is an effective and popular data-driven technique for obtaining sparse representations of data [3]–[5], is to learn a dictionary $\mathbf{D}$ such that every data sample can be approximated by a linear combination of a few atoms (columns) of $\mathbf{D}$. While DL has been widely studied, traditional DL approaches flatten tensor data and then employ methods designed for vector data [5], [6]. This ignores the multidimensional structure in tensor data, resulting in dictionaries with a large number of parameters. One intuitively expects that dictionaries which exploit the correlation and structure across tensor modes will have fewer parameters, leading to improvements in storage requirements, computational complexity, and generalization performance, especially when training data are noisy or scarce.

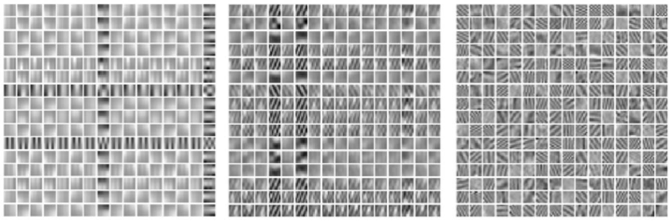


Fig. 1: Dictionary atoms for representing RGB image `Barbara` for separation rank (left-to-right) 1, 4, and 256.

To reduce the number of parameters in dictionaries for tensor data, and to better exploit the correlation among different tensor modes, some recent DL works use tensor decompositions such as the Tucker decomposition [7] and CANDECOMP/PARAFAC decomposition (CPD) [8] for learning of "structured" dictionaries. The idea in *structured DL* for tensor data is to restrict the class of dictionaries during training to the one imposed by the tensor decomposition under consideration [9]. For example, structured DL based on the Tucker decomposition of N-way tensor data corresponds to the dictionary class in which any dictionary $\mathbf{D} \in \mathbb{R}^{m \times p}$ consists of the Kronecker product [10] of N smaller *subdictionaries* $\{\mathbf{D}_n \in \mathbb{R}^{m_n \times p_n}\}_{n=1}^N$ [11]–[16]. The resulting DL techniques in this instance are interchangeably referred to in the literature as *separable DL* or *Kronecker-structured DL* (KS-DL).

In terms of parameters to be estimated and stored, the advantages of KS-DL for tensor data are straightforward: defining $m \triangleq \prod_{n=1}^N m_n$ and $p \triangleq \prod_{n=1}^N p_n$, unstructured dictionary learning uses $mp = \Pi_{n=1}^N m_n p_n$ parameters, whereas the KS-DL model uses only the sum of the subdictionary sizes $\sum_{n=1}^N m_n p_n$. Nonetheless, while existing KS-DL methods enjoy lower sample/computational complexity and better storage efficiency over unstructured DL [16], the KS-DL model makes a strong separability assumption among different modes of tensor data. Such an assumption can be overly restrictive for many classes of data [17], resulting in an unfavorable tradeoff between model compactness and representation power.

M. Ghassemi, A.D. Sarwate, and W.U. Bajwa are with the Department of Electrical and Computer Engineering, Rutgers, The State University of New Jersey, Piscataway, NJ, 08854 USA (Emails: {m.ghassemi, anand.sarwate, waheed.bajwa}@rutgers.edu). Z. Shakeri was with the Department of Electrical and Computer Engineering, Rutgers, The State University of New Jersey, Piscataway, NJ, 08854 USA and is now with Electronic Arts (Email: zshakeri@ea.com)

Some of the results reported here were presented at CAMSAP 2017 [1] and ISIT 2019 [2]. This work is supported in part by the U.S. National Science Foundation under awards CCF-1453073 and CCF-1910110 and by the U.S. Army Research Office under award W911NF-17-1-0546.

In this paper, we overcome this limitation by proposing and analyzing a generalization of KS-DL that we interchangeably refer to as *learning a mixture of separable dictionaries* or *low separation rank DL* (LSR-DL). The separation rank of a matrix $\mathbf{A}$ is defined as the minimum number of KS matrices whose sum equals $\mathbf{A}$ [18], [19]. The LSR-DL model interpolates between the under-parameterized separable model (a special case of LSR-DL model with separation rank 1) and the over-parameterized unstructured model. In particular, this model is a natural and consistent way to increase the number of parameters in structured DL (and therefore representation performance) while mimicking the compactness of the KS-DL model. Our numerical experiments confirm the advantages of the LSR-DL model (and algorithms) over both unstructured DL and KS-DL in terms of sample complexity/performance and the amount of memory needed to store the dictionary. Figure 1 also illustrates the difference between LSR-DL and KS-DL: while KS-DL learns dictionary atoms that cannot reconstruct diagonal structures perfectly because of the abundance of axis-aligned (horizontal/vertical) structures within them, LSR-DL also returns dictionary atoms with pronounced diagonal structures as the separation rank increases.

A. Main Contributions

We propose a new generalization of the separable DL model—which we call a mixture of separable dictionaries model or LSR-DL model—that has a smaller number of parameters than standard DL. We provide conditions under which a true dictionary is recoverable, up to a prescribed error, from tensor-valued training data generated from the LSR-DL model. Our analysis uses the conventional optimization-based formulation of the DL problem [4], except that the search space is constrained to the class of dictionaries with maximum separation rank r (and individual mixture terms having bounded norms when $N \geq 3$ and $r \geq 2$).[1] Due to our choice of Frobenius norm as the distance metric, similar to conventional DL problems, this LSR-DL problem is nonconvex with multiple global minima. Obtaining global convergence guarantees for this highly nonconvex problem is not only difficult but is also insufficient to guarantee global identifiability due to existence of multiple global minima. We therefore focus on *local identifiability* guarantees, meaning that a search algorithm initialized close enough to the true dictionary can recover that dictionary. Our local identifiability results show that the LSR-DL problem is well posed—i.e., it can return a good estimate of the true dictionary, up to a certain initialization distance, as a solution—and characterize the effect of the separation rank on the sample complexity of the learning problem. To this end, under certain assumptions on the generative model, we show that $\Omega\big(r(\sum_{n=1}^{N} m_n p_n) p^2 \rho^{-2}\big)$ samples ensure existence of a local minimum of the constrained LSR-DL problem for Nth-order tensor data within a neighborhood of radius ρ around the true LSR dictionary.

Our initial local identifiability results are based on an analysis of a separation rank-constrained optimization problem that exploits a connection between LSR (resp., KS) matrices and low-rank (resp., rank-1) tensors. The main challenge that we face as part of this analysis is understanding the topological properties of the class of dictionaries with separation rank at most r in terms of compactness and covering number. The resulting insights may be of independent interest to readers for other problems involving LSR matrices.

Next, we note that a result in tensor recovery literature [20] implies finding the separation rank of a matrix is NP-hard. While this means that the rank-constrained LSR-DL problem is computationally intractable, our analysis of this problem provides the basis for our second main contribution, which is development and analysis of two different relaxations of the LSR-DL problem that are computationally tractable in the sense that they do not require explicit computation of the separation rank. The first formulation once again exploits the connection between LSR matrices and low-rank tensors and uses a convex regularizer to implicitly constrain the separation rank of the learned dictionary. The second formulation enforces the LSR structure on the dictionary by explicitly writing it as a summation of r KS matrices. Our analyses of the two relaxations once again involve conditions under which the true LSR dictionary is locally recoverable from training tensor data. Our strongest result is for the factorized formulation, described formally in Section II, in which case we derive a sample complexity result that is similar to that of the intractable formulation by finding a correspondence between the local minima of the factorized problem and those of the intractable one. We also compare and contrast the three sets of identifiability results for LSR dictionaries in the body.

In addition to showing the well-posedness of the LSR-DL problem, our theoretical results on local identifiability confirm the advantages of exploiting low-rank structure in tensor problems. Moreover, in order to obtain our results, we acquire a better understanding of the topological properties of the space of LSR matrices. These properties may be useful in other works involving KS and LSR models.

Our third main contribution is the development of practical computational algorithms, which are based on the two relaxations of LSR-DL, for learning of an LSR dictionary in both batch and online settings. We use these algorithms for learning of LSR dictionaries for both synthetic and real tensor data and show their effectiveness in denoising and representation learning tasks. Numerical results obtained as part of these efforts help validate the usefulness of our proposed LSR-DL model and highlight the different strengths and weaknesses of the two LSR-DL relaxations and the corresponding algorithms. In particular, we show empirically that our algorithms provide better representations with a smaller number of parameters than the conventional approach.

B. Relation to Prior Work

Tensor decompositions [21], [22] are an important tool for avoiding overparameterization of tensor data models in a variety of areas. These include deep learning, collaborative filtering, multilinear subspace learning, source separation, topic modeling, and many other works (see recent surveys [23],

[1] We also provide asymptotic identifiability results for LSR dictionaries without requiring the boundedness assumption; see Section III for details.

[24] and references therein). However, the use of tensor decompositions for reducing the (model and sample) complexity of dictionaries for tensor data has been addressed only recently.

Many recent works provide theoretical analysis for the sample complexity of the conventional DL problem [25]–[28]. Among these, Gribonval et al. [27] focus on the local identifiability of the true dictionary underlying vectorized data using Frobenius norm as the distance metric. Shakeri et al. [16] extended this analysis for the sample complexity of the KS-DL problem for Nth-order tensor data. This analysis relies on expanding the objective function in terms of subdictionaries and exploiting the coordinate-wise Lipschitz continuity property of the objective function with respect to each subdictionary [16]. While this approach ensures the identifiability of the subdictionaries, it requires the dictionary coefficient vectors to follow the so-called *separable sparsity model* [29] and does not extend to the LSR-DL problem. By contrast, we provide local identifiability sample complexity results for the LSR-DL problem and its two relaxations. Further, our identifiability results hold for coefficient vectors following both the random and separable sparsity models.

In terms of computational algorithms, several works have proposed methods for learning KS dictionaries that rely on alternating minimization techniques to update the subdictionaries [12], [14], [29]. Among other works, Hawe et al. [11] employ a Riemannian conjugate gradient method combined with a nonmonotone line search for KS-DL. While they present the algorithm only for matrix data, its extension to higher-order tensor data is trivial. Schwab et al. [30] have also recently addressed the separable DL problem for matrix data; their contributions include a computational algorithm and global recovery guarantees. In terms of algorithms for LSR-DL, Dantas et al. [13] proposed one of the first methods for matrix data that uses a convex regularizer to impose LSR on the dictionary. One of our batch algorithms, named STARK [1], also uses a convex regularizer for imposing LSR structure. In contrast to Dantas et al. [13], however, STARK can be used to learn a dictionary from tensor data of any order. The other batch algorithm we propose, named TeFDiL, learns subdictionaries of the LSR dictionary by exploiting the connection to tensor recovery and using tensor CPD. Recently, Dantas et al. [31] proposed an algorithm for learning an LSR dictionary for tensor data in which the dictionary update stage is a projected gradient descent algorithm that involves a CPD after every gradient step. In contrast, TeFDiL only requires a single CPD at the end of each dictionary update stage. Finally, while there exist a number of online algorithms for DL [6], [32], [33], the online algorithm developed in here is the first one that enables learning of structured (either KS or LSR) dictionaries.

C. Organization

In Section II, we provide the necessary background on dictionary learning, introduce our LSR-DL model, and formulate three variants of the LSR-DL problem. In Section III, we show that LSR dictionaries are identifiable using the rank-constrained formulation of the LSR-DL problem. In Section IV, we study the local identifiability of the other two (regularized and factorized) formulations in both asymptotic and finite sample regimes. In Section V, we use the regularized and factorized formulations to design batch and online LSR-DL algorithms, which we evaluate experimentally in Section VI. We conclude the paper and discuss possible future work in Section VII. Proof of Lemma 1 (the rearrangement procedure) is explained in detail in Appendix A. Proofs of technical Lemmas 2, 3, 6, 7, and 8 are provided in Appendix B. A discussion on the convergence of our algorithms is provided in Appendix C.

II. Preliminaries and Problem Statement

Notation and Definitions: We use underlined bold upper-case ($\underline{\mathbf{A}}$), bold upper-case ($\mathbf{A}$), bold lower-case ($\mathbf{a}$), and lower-case (a) letters to denote tensors, matrices, vectors, and scalars, respectively. For any integer p, we define $[p] \triangleq \{1, 2, \cdots, p\}$. We denote the j-th column of a matrix $\mathbf{A}$ by $\mathbf{a}_j$. For an $m \times p$ matrix $\mathbf{A}$ and an index set $\mathcal{J} \subseteq [p]$, we denote the matrix constructed from the columns of $\mathbf{A}$ indexed by $\mathcal{J}$ as $\mathbf{A}_{\mathcal{J}}$. We denote by $(\mathbf{A}_n)_{n=1}^N$ an N-tuple $(\mathbf{A}_1, \cdots, \mathbf{A}_N)$, while $\{\mathbf{A}_n\}_{n=1}^N$ represents the set $\{\mathbf{A}_1, \cdots, \mathbf{A}_N\}$. We drop the range indicators if they are clear from the context.

Norms and inner products: We denote by $\|\mathbf{v}\|_p$ the ℓ_p norm of vector $\mathbf{v}$ (we abuse the terminology in case of $p = 0$), while we use $\|\mathbf{A}\|_2$, $\|\mathbf{A}\|_F$, and $\|\mathbf{A}\|_{\mathrm{tr}}$ to denote the spectral, Frobenius, and trace (nuclear) norms of matrix $\mathbf{A}$, respectively. Moreover, $\|\mathbf{A}\|_{2,\infty} \triangleq \max_j \|\mathbf{a}_j\|_2$ is the *max column norm* and $\|\mathbf{A}\|_{1,1} \triangleq \sum_j \|\mathbf{a}_j\|_1$. We define the inner product of two tensors (or matrices) $\underline{\mathbf{A}}$ and $\underline{\mathbf{B}}$ as $\langle \underline{\mathbf{A}}, \underline{\mathbf{B}} \rangle \triangleq \langle \mathrm{vec}(\underline{\mathbf{A}}), \mathrm{vec}(\underline{\mathbf{B}}) \rangle$ where $\mathrm{vec}(\cdot)$ is the vectorization operator. We define the Frobenius norm of tensor $\underline{\mathbf{A}}$ as $\|\underline{\mathbf{A}}\|_F = \sqrt{\langle \underline{\mathbf{A}}, \underline{\mathbf{A}} \rangle}$. The Euclidean distance between two tuples of the same size is defined as $\left\|(\mathbf{A}_n)_{n=1}^N - (\mathbf{B}_n)_{n=1}^N\right\|_F \triangleq \sqrt{\sum_{n=1}^N \|\mathbf{A}_n - \mathbf{B}_n\|_F^2}$.

Kronecker product: We denote by $\mathbf{A} \otimes \mathbf{B} \in \mathbb{R}^{m_1 m_2 \times p_1 p_2}$ the Kronecker product of matrices $\mathbf{A} \in \mathbb{R}^{m_1 \times p_1}$ and $\mathbf{B} \in \mathbb{R}^{m_2 \times p_2}$. We use $\bigotimes_{n=1}^N \mathbf{A}_n \triangleq \mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \cdots \otimes \mathbf{A}_N$ for the Kronecker product of N matrices. We drop the range indicators when there is no ambiguity. We call a matrix a (N-th order) Kronecker-structured (KS) matrix if it is a Kronecker product of $N \geq 2$ matrices.

Definitions for matrices: For a matrix $\mathbf{D}$ with unit ℓ_2-norm columns, we define the *cumulative coherence* $\mu_s(\mathbf{D})$ as $\mu_s(D) \triangleq \max_{|\mathcal{J}| \leq s} \max_{j \notin \mathcal{J}} \|\mathbf{D}_{\mathcal{J}}^T \mathbf{d}_j\|_1$. We say a matrix $\mathbf{D}$ satisfies the *s-restricted isometry property* (s-RIP) with constant δ_s if for any $\mathbf{v} \in \mathbb{R}^s$ and any $\mathcal{J} \subseteq [p]$ with $|\mathcal{J}| \leq s$, we have $(1 - \delta_s)\|\mathbf{v}\|_2^2 \leq \|\mathbf{D}_{\mathcal{J}} \mathbf{v}\|_2^2 \leq (1 + \delta_s)\|\mathbf{v}\|_2^2$.

Definitions for tensors: We briefly present required tensor definitions here: see Kolda and Bader [21] for more details. The mode-n unfolding matrix of $\underline{\mathbf{A}}$ is denoted by $\mathbf{A}_{(n)}$, where each column of $\mathbf{A}_{(n)}$ consists of the vector formed by fixing all indices of $\underline{\mathbf{A}}$ except the one in the nth-order. We denote the outer product (tensor product) of vectors by $\circ$, while $\times_n$ denotes the mode-n product between a tensor and a matrix. An N-way tensor is rank-1 if it can be written as outer product of N vectors: $\mathbf{v}_1 \circ \cdots \circ \mathbf{v}_N$. Throughout this paper, by the rank of a tensor, $\mathrm{rank}(\underline{\mathbf{A}})$, we mean the CP-rank of $\underline{\mathbf{A}}$, the minimum number of rank-1 tensors that construct $\underline{\mathbf{A}}$ as their sum. The *CP decomposition* (CPD), decomposes a tensor into sum of its rank-1 tensor components. The *Tucker decomposition* factorizes an N-way tensor $\underline{\mathbf{A}} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_N}$ as $\underline{\mathbf{A}} = \underline{\mathbf{X}} \times_1 \mathbf{D}_1 \times_2 \mathbf{D}_2 \times_3 \cdots \times_N \mathbf{D}_N$, where $\underline{\mathbf{X}} \in \mathbb{R}^{p_1 \times p_2 \times \cdots \times p_N}$ denotes the core tensor and $\mathbf{D}_n \in \mathbb{R}^{m_n \times p_n}$ denote factor matrices along the n-th mode of $\underline{\mathbf{A}}$ for $n \in [N]$.

Notations for functions and spaces: We denote the element-wise sign function by $\operatorname{sgn}(\cdot)$. For any function $f(\mathbf{x})$, we define the difference $\Delta f(\mathbf{x}_1; \mathbf{x}_2) \triangleq f(\mathbf{x}_1) - f(\mathbf{x}_2)$. We denote by $\mathcal{U}_{m \times p}$ the Euclidean unit sphere: $\mathcal{U}_{m \times p} \triangleq \{\mathbf{D} \in \mathbb{R}^{m \times p} | \|\mathbf{D}\|_F = 1\}$. We also denote the Euclidean sphere with radius α by $\alpha\mathcal{U}_{m \times p}$. The oblique manifold in $\mathbb{R}^{m \times p}$ is the manifold of matrices with unit-norm columns: $\mathcal{D}_{m \times p} \triangleq \{\mathbf{D} \in \mathbb{R}^{m \times p} | \forall j \in [p],\ \mathbf{d}_j^T \mathbf{d}_j = 1\}$. We drop the dimension subscripts and use only $\mathcal{D}$ when there is no ambiguity. The covering number of a set $\mathcal{A}$ with respect to a norm $\|\cdot\|_*$, denoted by $\mathcal{N}_*(\mathcal{A}, \epsilon)$, is the minimum number of balls of $*$-norm radius ϵ needed to cover $\mathcal{A}$.

Dictionary Learning Setup: In dictionary learning (DL) for vector data, we assume observations $\mathbf{y} \in \mathbb{R}^m$ are generated according to the following model:

$$\mathbf{y} = \mathbf{D}^0 \mathbf{x}^0 + \boldsymbol{\epsilon}, \tag{1}$$

where $\mathbf{D}^0 \in \mathcal{D}_{m \times p} \subset \mathbb{R}^{m \times p}$ is the true underlying dictionary, $\mathbf{x}^0 \in \mathbb{R}^p$ is a randomly generated sparse coefficient vector, and $\boldsymbol{\epsilon} \in \mathbb{R}^m$ is the observation noise vector. The goal in DL is to recover the true dictionary given the noisy observations $\mathbf{Y} \triangleq \{\mathbf{y}_l\}_{l=1}^L$ that are independent realizations of (1). The ideal objective is to solve the statistical risk minimization problem

$$\min_{\mathbf{D} \in \mathcal{C}} f_{\mathcal{P}}(\mathbf{D}) \triangleq \mathbb{E}_{\mathbf{y} \sim \mathcal{P}}\, f_{\mathbf{y}}(\mathbf{D}), \tag{2}$$

where $\mathcal{P}$ is the underlying distribution of the observations, $\mathcal{C} \subseteq \mathcal{D}_{m \times p}$ is the dictionary class, typically selected for vector data to be the same as the oblique manifold, and

$$f_{\mathbf{y}}(\mathbf{D}) \triangleq \inf_{\mathbf{x} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda \|\mathbf{x}\|_1. \tag{3}$$

However, since we have access to the distribution $\mathcal{P}$ only through noisy observations drawn from this distribution, we resort to solving the following empirical risk minimization problem as a proxy for Problem (2):

$$\min_{\mathbf{D} \in \mathcal{C}} F_{\mathbf{Y}}(\mathbf{D}) \triangleq \frac{1}{L} \sum_{l=1}^{L} f_{\mathbf{y}_l}(\mathbf{D}). \tag{4}$$

Dictionary Learning for Tensor Data: To represent tensor data, conventional DL approaches vectorize tensor data samples and treat them as one-dimensional arrays. One way to explicitly account for the tensor structure in data is to use the Kronecker-structured DL (KS-DL) model, which is based on the Tucker decomposition of tensor data. In the KS-DL model, we assume that observations $\underline{\mathbf{Y}}_l \in \mathbb{R}^{m_1 \times \cdots \times m_N}$ are generated according to

$$\underline{\mathbf{Y}}_l = \underline{\mathbf{X}}_l^0 \times_1 \mathbf{D}_1^0 \times_2 \mathbf{D}_2^0 \times_3 \cdots \times_N \mathbf{D}_N^0 + \underline{\mathcal{E}}_l, \tag{5}$$

where $\{\mathbf{D}_n^0 \in \mathbb{R}^{m_n \times p_n}\}_{n=1}^N$ are generating *subdictionaries*, and $\underline{\mathbf{X}}_l^0$ and $\underline{\mathcal{E}}_l$ are the coefficient and noise tensors, respectively. Equivalently, the generating model (5) can be stated for $\mathbf{y}_l \triangleq \operatorname{vec}(\underline{\mathbf{Y}}_l)$ as:

$$\mathbf{y}_l = \left(\mathbf{D}_N^0 \otimes \mathbf{D}_{N-1}^0 \otimes \cdots \otimes \mathbf{D}_1^0\right) \mathbf{x}_l^0 + \boldsymbol{\epsilon}_l, \tag{6}$$

where $\mathbf{x}_l^0 \triangleq \operatorname{vec}(\underline{\mathbf{X}}_l^0)$ and $\boldsymbol{\epsilon}_l \triangleq \operatorname{vec}(\underline{\mathcal{E}}_l)$ [21]. This is the same as the unstructured model $\mathbf{y}_l = \mathbf{D}^0 \mathbf{x}_l^0 + \boldsymbol{\epsilon}_l$ with the additional condition that the generating dictionary is a Kronecker product of N subdictionaries. As a result, in the KS-DL problem, the constraint set in (4) becomes $\mathcal{C} = \mathcal{K}_{\mathbf{m},\mathbf{p}}^N$, where $\mathcal{K}_{\mathbf{m},\mathbf{p}}^N \triangleq \{\mathbf{D} \in \mathcal{D}_{m \times p} | \mathbf{D} = \bigotimes_{n=1}^N \mathbf{D}_n,\ \mathbf{D}_n \in \mathbb{R}^{m_n \times p_n}\}$ is the set of KS matrices with unit-norm columns and $\mathbf{m}$ and $\mathbf{p}$ are vectors containing m_n's and p_n's, respectively.[2]

In summary, the structure in tensor data is exploited in the KS-DL model by assuming the dictionary is "separable" into subdictionaries for each mode. However, as discussed earlier, this separable model is rather restrictive. Instead, we generalize the KS-DL model using the notion of *separation rank*.[3]

Definition 1. *The separation rank* $\mathfrak{R}_{\mathbf{m},\mathbf{p}}^N(\cdot)$ *of a matrix* $\mathbf{A} \in \mathbb{R}^{\Pi_n m_n \times \Pi_n p_n}$ *is the minimum number* r *of* N*th-order KS matrices* $\mathbf{A}^k = \bigotimes_{n=1}^N \mathbf{A}_n^k$ *such that* $\mathbf{A} = \sum\limits_{k=1}^{r} \bigotimes_{n=1}^N \mathbf{A}_n^k$, *where* $\mathbf{A}_n^k \in \mathbb{R}^{m_n \times p_n}$.

The KS-DL model corresponds to dictionaries with separation rank 1. We instead propose the *low separation rank (LSR)* DL model in which the separation rank of the underlying dictionary is relatively small so that $1 \leq \mathfrak{R}_{\mathbf{m},\mathbf{p}}(\mathbf{D}^0) \ll \min\{m, p\}$. This generalizes the KS-DL model to a generating dictionary of the form $\mathbf{D}^0 = \sum_{k=1}^r [\mathbf{D}_N^k]^0 \otimes [\mathbf{D}_{N-1}^k]^0 \otimes \cdots \otimes [\mathbf{D}_1^k]^0$, where r is the separation rank of $\mathbf{D}^0$. Consequently, defining $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r} \triangleq \{\mathbf{D} \in \mathcal{D}_{m \times p} | \mathfrak{R}_{\mathbf{m},\mathbf{p}}^N(\mathbf{D}) \leq r\}$, the empirical *rank-constrained LSR-DL problem* is

$$\min_{\mathbf{D} \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}} F_{\mathbf{Y}}(\mathbf{D}). \tag{7}$$

However, the analytical tools at our disposal require the constraint set in (7) to be closed, which we show does not hold for $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ when $N \geq 3$ and $r \geq 2$. In that case, we instead analyze (7) with $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ replaced by (i) closure of $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ and (ii) a certain closed subset of $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$. We refer the reader to Section III for further discussion.

In our study of the LSR-DL model (which includes the KS-DL model as a special case), we use a correspondence between KS matrices and rank-1 tensors, stated in Lemma 1 below, which allows us to leverage techniques and results in the tensor recovery literature to analyze the LSR-DL problem and develop tractable algorithms. (This correspondence was first exploited in our earlier work [1].)

Lemma 1. *Any* N*th-order Kronecker-structured matrix* $\mathbf{A} = \mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \cdots \otimes \mathbf{A}_N$ *can be rearranged as a rank-*1*,* N*th-order tensor* $\underline{\mathbf{A}}^\pi = \mathbf{a}_N \circ \cdots \circ \mathbf{a}_2 \circ \mathbf{a}_1$ *with* $\mathbf{a}_n \triangleq \operatorname{vec}(\mathbf{A}_n)$.

Figure 2 provides an example of the rearrangement procedure, which involves finding corresponding indices on the KS matrix and the tensor. A proof of Lemma 1, which includes details of the rearrangement strategy, is provided in

[2] We have changed the indexing of subdictionaries for ease of notation.

[3] The term was introduced in Tsiligkaridis and Hero [19] for $N = 2$ (see also Beylkin and Mohlenkamp [18]).

TABLE I: Table of commonly used notation

Notation	Definition	Notation	Definition
m, p	$\prod_{n=1}^N m_n,\ \prod_{n=1}^N p_n$	$\mathbf{m}, \mathbf{p}$	$(m_n)_{n=1}^N,\ (p_n)_{n=1}^N$
$\mathcal{N}_*(\mathcal{A}, \epsilon)$	Covering number of set $\mathcal{A}$ w.r.t. norm $*$	$\mathfrak{R}_{\mathbf{m},\mathbf{p}}^N(\mathbf{D})$	Separation rank of matrix $\mathbf{D}$
$\mathcal{D}_{m \times p}$	Oblique manifold in $\mathbb{R}^{m \times p}$	$\mathcal{U}_{m \times p}$	Euclidean unit sphere in $\mathbb{R}^{m \times p}$
$\mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r}$	Set of LSR matrices: $\{\mathbf{D} \in \mathbb{R}^{m \times p} \mid \mathfrak{R}_{\mathbf{m},\mathbf{p}}^N(\mathbf{D}) \leq r\}$	$\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$	$\mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r} \cap \mathcal{D}_{m \times p}$
$\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N}$	$\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ with $r = 1$: Set of KS matrices on $\mathcal{D}_{m \times p}$	$\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r}$	$\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ with $N = 2$
${}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$	$\{\mathbf{D} \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r} \mid \lVert \bigotimes \mathbf{D}_n^k \rVert_F \leq c, c > 0\}$	$\overline{\mathcal{K}}_{\mathbf{m},\mathbf{p}}^{N,r}$	Closure of $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$
$\mathcal{C}$	Compact constraint set in LSR-DL problem: one of $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N}$, $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r}$, ${}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$, or $\overline{\mathcal{K}}_{\mathbf{m},\mathbf{p}}^{N,r}$	$\mathcal{B}_\rho$	$\{\mathbf{D} \in \mathcal{C} \mid \lVert \mathbf{D} - \mathbf{D}^0 \rvert_F \leq \rho\}$
$\Delta f(\mathbf{x}_1; \mathbf{x}_2)$	$f(\mathbf{x}_1) - f(\mathbf{x}_2)$	$f_{\mathbf{y}}(\mathbf{D})$	$\inf_{\mathbf{x} \in \mathbb{R}^p} \frac{1}{2} \lVert \mathbf{y} - \mathbf{D}\mathbf{x} \rVert_2^2 + \lambda \lVert \mathbf{x} \rVert_1$
$f_{\mathcal{P}}(\mathbf{D})$	$\mathbb{E}_{\mathbf{y} \sim \mathcal{P}}\ f_{\mathbf{y}}(\mathbf{D})$	$\Delta f_{\mathcal{P}}(\rho)$	$\inf_{\mathbf{D} \in \partial \mathcal{B}_\rho} \Delta f_{\mathcal{P}}(\mathbf{D}; \mathbf{D}^0)$
$F_{\mathbf{Y}}(\mathbf{D})$	$\frac{1}{L} \sum_{l=1}^L f_{\mathbf{y}_l}(\mathbf{D})$	$F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D})$	$\frac{1}{L} \sum_{l=1}^L f_{\mathbf{y}_l}(\mathbf{D}) + \lambda_1 g_1(\underline{\mathbf{D}}^\pi)$
$f_{\mathbf{y}}^{\text{fac}}(\{\mathbf{D}_n^k\})$	$\inf_{\mathbf{x} \in \mathbb{R}^p} \left\lVert \mathbf{y} - \left(\sum_{k=1}^r \bigotimes_{n=1}^N \mathbf{D}_n^k\right)\mathbf{x} \right\rVert_2^2 + \lambda \lVert \mathbf{x} \rVert_1$	$F_{\mathbf{Y}}^{\text{fac}}(\{\mathbf{D}_n^k\})$	$\frac{1}{L} \sum_{l=1}^L f_{\mathbf{y}_l}^{\text{fac}}(\{\mathbf{D}_n^k\})$

Appendix A. It follows immediately from Lemma 1 that if $\mathbf{D} = \sum_{k=1}^r \mathbf{D}_1^k \otimes \cdots \otimes \mathbf{D}_N^k$, then we can rearrange matrix $\mathbf{D}$ into the tensor $\underline{\mathbf{D}}^\pi = \sum_{k=1}^r \mathbf{d}_N^k \circ \mathbf{d}_{N-1}^k \circ \cdots \circ \mathbf{d}_1^k$, where $\mathbf{d}_n^k = \text{vec}(\mathbf{D}_n^k)$. Hence, we have the following equivalence:

$$\mathfrak{R}_{\mathbf{m},\mathbf{p}}^N(\mathbf{D}) \leq r \iff \text{rank}(\underline{\mathbf{D}}^\pi) \leq r.$$

This correspondence between separation rank and tensor rank highlights a challenge with the LSR-DL problem: finding the rank of a tensor is NP-hard [20] and thus so is finding the separation rank of a matrix. This makes Problem (7) in its current form (and its variants) intractable. To overcome this limitation, we introduce two tractable relaxations to the rank-constrained Problem (7) that do not require explicit computation of the tensor rank. The first relaxation uses a convex regularization term to implicitly impose low tensor rank structure on $\underline{\mathbf{D}}^\pi$, which results in a low separation rank $\mathbf{D}$. The resulting empirical *regularization-based LSR-DL problem* is

$$\min_{\mathbf{D} \in \mathcal{D}_{m \times p}} F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D}) \tag{8}$$

with $F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D}) \triangleq \frac{1}{L} \sum_{l=1}^L f_{\mathbf{y}_l}(\mathbf{D}) + \lambda_1 g_1(\underline{\mathbf{D}}^\pi)$, where $f_{\mathbf{y}}(\mathbf{D})$ is described in (3) and $g_1(\underline{\mathbf{D}}^\pi)$ is a convex regularizer to enforce low-rank structure on $\underline{\mathbf{D}}^\pi$. The second relaxation is a *factorization-based LSR-DL formulation* in which the LSR dictionary is explicitly written in terms of its subdictionaries. The resulting empirical risk minimization problem is

$$\min_{\{\mathbf{D}_n^k\}:\ \sum_{k=1}^r \bigotimes_{n=1}^N \mathbf{D}_n^k \in \mathcal{D}_{m \times p}} F_{\mathbf{Y}}^{\text{fac}}(\{\mathbf{D}_n^k\}), \tag{9}$$

where $F_{\mathbf{Y}}^{\text{fac}}(\{\mathbf{D}_n^k\}) \triangleq \frac{1}{L} \sum_{l=1}^L f_{\mathbf{y}_l}^{\text{fac}}(\{\mathbf{D}_n^k\})$ with

$$f_{\mathbf{y}}^{\text{fac}}(\{\mathbf{D}_n^k\}) \triangleq \inf_{\mathbf{x} \in \mathbb{R}^p} \left\| \mathbf{y} - \Big(\sum_{k=1}^r \bigotimes_{n=1}^N \mathbf{D}_n^k\Big)\mathbf{x} \right\|_2^2 + \lambda \|\mathbf{x}\|_1,$$

and the terms $\bigotimes_{n=1}^N \mathbf{D}_n^k$ are constrained as $\| \bigotimes_{n=1}^N \mathbf{D}_n^k \|_F \leq c$ for some positive constant c when $N \geq 3$ and $r \geq 2$.

In the rest of this paper, we study the problem of identifying the true underlying LSR-DL dictionary by analyzing the LSR-DL Problems (7)–(9) introduced in this section and developing algorithms to solve Problems (8) and (9) in both batch and online settings. Note that while Problem (7) (and its variants when $N \geq 3$ and $r \geq 2$) cannot be explicitly solved because of its NP-hardness, identifiability analysis of this problem—provided in Section III—provides the basis for the analysis of tractable Problems (8) and (9), provided in Section IV. To improve the readability of our notation-heavy discussions and analysis, we have provided a table of notations (Table I) for easy access to definitions of the most commonly used notation.

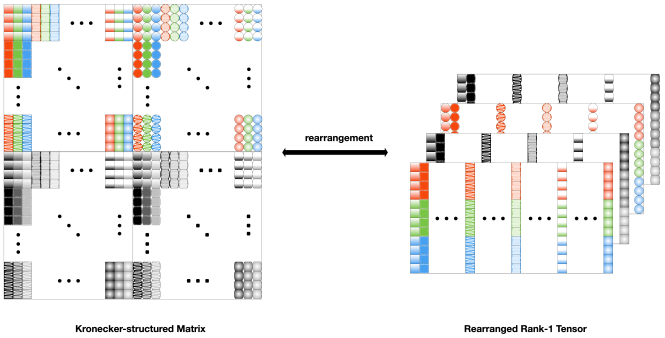


Fig. 2: Example of rearranging a Kronecker structured matrix ($N = 3$) into a third order rank-1 tensor.

III. Identifiability in the Rank-constrained LSR-DL Problem

In this section, we derive conditions under which a dictionary $\mathbf{D}^0 \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ is identifiable as a solution to either the separation rank-constrained problem in (7) or a slight variant of (7) when $N \geq 3$ and $r \geq 2$. Specifically, we show that under certain assumptions on the generative model, there is at least one local minimum $\mathbf{D}^*$ of either Problem (7) or one of its variants that is "close" to the underlying dictionary $\mathbf{D}^0$. Notwithstanding the fact that no efficient algorithm exists to solve the intractable Problem (7), this identifiability result is important in that it lays the foundation for the local identifiability results in tractable Problems (8) and (9).

Generative Model: Let $\mathbf{D}^0 \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ be the underlying dictionary. Each tensor data sample $\underline{\mathbf{Y}} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_N}$ in its vectorized form is *independently* generated using a linear combination of $s \ll p$ atoms of dictionary $\mathbf{D}^0$ with added noise: $\mathbf{y} \triangleq \mathrm{vec}(\underline{\mathbf{Y}}) = \mathbf{D}^0\mathbf{x}^0 + \boldsymbol{\epsilon}$, where $\left\|\mathbf{x}^0\right\|_0 \leq s$. Specifically, s atoms of $\mathbf{D}^0$ are selected uniformly at random, defining the support $\mathcal{J} \subset [p]$. Then, we draw a random sparse coefficient vector $\mathbf{x}^0 \in \mathbb{R}^p$ supported on $\mathcal{J}$. We state further assumptions on our model similar to prior works [16], [27].

Assumption 1 (Coefficient Distribution)**.** *Consider a random variable $x \in \mathbb{R}$ and positive constants M_x and $\underline{x}$. Define $\mathbf{s}^0 \triangleq \mathrm{sgn}(\mathbf{x}^0)$. We assume:* ***i)*** $\mathbb{E}\left\{\mathbf{x}^0_{\mathcal{J}}[\mathbf{x}^0_{\mathcal{J}}]^T|\mathcal{J}\right\} = \mathbb{E}\{x^2\} \cdot \mathbf{I}_s$*,* ***ii)*** $\mathbb{E}\left\{\mathbf{s}^0_{\mathcal{J}}[\mathbf{s}^0_{\mathcal{J}}]^T|\mathcal{J}\right\} = \mathbf{I}_s$*,* ***iii)*** $\mathbb{E}\left\{\mathbf{s}^0_{\mathcal{J}}[\mathbf{x}^0_{\mathcal{J}}]^T|\mathcal{J}\right\} = \mathbb{E}\{|x|\} \cdot \mathbf{I}_s$*, and* ***iv)*** $\left\|\mathbf{x}^0\right\|_2 \leq M_x$ *and* $\min\limits_{j\in\mathcal{J}} |\mathbf{x}^0_j| \geq \underline{x}$ *almost surely.*

Assumption 2 (Noise Distribution)**.** *Consider a random variable $\epsilon \in \mathbb{R}$ and positive constant M_ϵ. We assume:* ***i)*** $\mathbb{E}\left\{\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T|\mathcal{J}\right\} = \mathbb{E}\{\epsilon^2\}\cdot\mathbf{I}_m$*,* ***ii)*** $\mathbb{E}\left\{\mathbf{x}^0\boldsymbol{\epsilon}^T|\mathcal{J}\right\} = \mathbb{E}\left\{\mathbf{s}^0\boldsymbol{\epsilon}^T|\mathcal{J}\right\} = 0$*, and* ***iii)*** $\|\boldsymbol{\epsilon}\|_2 \leq M_\epsilon$ *almost surely.*

Note that Assumptions 1-iv and 2-iii imply the magnitude of $\mathbf{y}$ is bounded: $\|\mathbf{y}\|_2 \leq M_y$. Next, we define positive parameters $\bar{\lambda} \triangleq \frac{\lambda}{\mathbb{E}\{|x|\}}$, $C_{\min} \triangleq 24\frac{\mathbb{E}\{|x|\}^2}{\mathbb{E}\{x^2\}}\left(\left\|\mathbf{D}^0\right\|_2+1\right)^2 \frac{s}{p}\left\|[\mathbf{D}^0]^T\mathbf{D}^0 - \mathbf{I}\right\|_F$, and $C_{\max} \triangleq \frac{2\mathbb{E}\{|x|\}}{7M_x}\left(1 - 2\mu_s(\mathbf{D}^0)\right)$ for ease of notation. We use the following assumption, similar to Gribonval et al. [27, Thm. 1].

Assumption 3. *Assume $C_{\min} \leq C_{\max}$, $\lambda \leq \underline{x}/4$, $s \leq \frac{p}{16\left(\|\mathbf{D}^0\|_2+1\right)^2}$, $\mu_s(\mathbf{D}^0) \leq 1/4$, and the noise is relatively small in the sense that $\frac{M_\epsilon}{M_x} < \frac{7}{2}\left(C_{\max} - C_{\min}\right)\bar{\lambda}$.*

Our Approach: In our analysis of the separation rank-constrained LSR-DL problem, we will alternate between four different constraint sets that are related to our dictionary class $\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$, namely, $\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}$, $\mathcal{K}^{N}_{\mathbf{m},\mathbf{p}}$, the closure $\overline{\mathcal{K}}^{N,r}_{\mathbf{m},\mathbf{p}} \triangleq \mathrm{cl}(\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}})$ of $\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$ under the Frobenius norm, and a closed subset of $\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$, defined as ${}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}} \triangleq \{\mathbf{D} \in \mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}} | \| \bigotimes \mathbf{D}^k_n\|_F \leq c, c > 0\}$. We often use the generic notation $\mathcal{C}$ for the constraint set when our discussion is applicable to more than one of these sets.

We want to find conditions that imply the existence of a local minimum of $\min_{\mathbf{D}\in\mathcal{C}} F_{\mathbf{Y}}(\mathbf{D})$ within a ball of radius ρ around the true dictionary $\mathbf{D}^0 \in \mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$:

$$\mathcal{B}_\rho \triangleq \{\mathbf{D} \in \mathcal{C} | \left\|\mathbf{D} - \mathbf{D}^0\right\|_F \leq \rho\} \tag{10}$$

for some small $\rho > 0$. To this end, we first show that the expected risk function $f_{\mathcal{P}}(\mathbf{D})$ in (2) has a local minimum in $\mathcal{B}_\rho$ for the LSR-DL constraint set $\mathcal{C}$.

To show that a local minimum of $f_{\mathcal{P}} : \mathcal{C} \mapsto \mathbb{R}$ exists in $\mathcal{B}_\rho$, we need to show that $f_{\mathcal{P}}(\mathbf{D})$ attains its minimum over $\mathcal{B}_\rho$ in the interior of $\mathcal{B}_\rho$.[4] We show this in two stages. First, we use the Weierstrass Extreme Value Theorem [34], which dictates that the continuous function $f_{\mathcal{P}}(\mathbf{D})$ attains a minimum in (or on the boundary of) $\mathcal{B}_\rho$ as long as $\mathcal{B}_\rho$ is a compact set. Therefore, we first investigate compactness of $\mathcal{B}_\rho$ in Section III-A. Second, in order to be certain that the minimizer of $f_{\mathcal{P}}(\mathbf{D})$ over $\mathcal{B}_\rho$ is a local minimum of $\mathbf{D} \in \mathcal{C} \mapsto f_{\mathcal{P}}(\mathbf{D})$, we show that $f_{\mathcal{P}}(\mathbf{D})$ cannot obtain its minimum over $\mathcal{B}_\rho$ on the boundary of $\mathcal{B}_\rho$, denoted by $\partial\mathcal{B}_\rho$. To this end, in Section III-B we derive conditions that if $\partial\mathcal{B}_\rho$ is nonempty then we have[5]

$$\Delta f_{\mathcal{P}}(\rho) \triangleq \inf_{\mathbf{D}\in\partial\mathcal{B}_\rho} \Delta f_{\mathcal{P}}(\mathbf{D};\mathbf{D}^0) > 0, \tag{11}$$

which implies $f_{\mathcal{P}}(\mathbf{D})$ cannot achieve its minimum on $\partial\mathcal{B}_\rho$.

Finally, in Section III-C we use concentration of measure inequalities to relate $F_{\mathbf{Y}}(\mathbf{D})$ in (4) to $f_{\mathcal{P}}(\mathbf{D})$ and find the number of samples needed to guarantee (with high probability) that $F_{\mathbf{Y}}(\mathbf{D})$ also has a local minimum in the interior of $\mathcal{B}_\rho$.

A. Compactness of the Constraint Sets

When the constraint set $\mathcal{C}$ is a compact subset of the Euclidean space $\mathbb{R}^{m\times p}$, the subset $\mathcal{B}_\rho$ is also compact. Thus, we first investigate the compactness of the constraint set $\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$. Since $\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$ is a bounded set, according to the Heine-Borel Theorem [34], it is a compact subset of $\mathbb{R}^{m\times p}$ if and only if it is closed. Also, $\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$ can be written as the intersection of $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}} \triangleq \{\mathbf{D} \in \mathbb{R}^{m\times p} | \mathfrak{R}^N_{\mathbf{m},\mathbf{p}}(\mathbf{D}) \leq r\}$ and the oblique manifold $\mathcal{D}$. In order for $\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}} = \mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}} \cap \mathcal{D}$ to be closed, it suffices to show that $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ and $\mathcal{D}$ are closed. It is trivial to show $\mathcal{D}$ is closed; hence, we focus on whether $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ is closed.

In the following, we use the facts that the constraint $\mathfrak{R}^N_{\mathbf{m},\mathbf{p}}(\mathbf{D}) \leq r$ is equivalent to $\mathrm{rank}(\underline{\mathbf{D}}^\pi) \leq r$ and that the rearrangement mapping that sends $\mathbf{D}$ to $\underline{\mathbf{D}}^\pi$ preserves topological properties of sets such as the distances between the set elements under the Frobenius norm. These facts allow us to translate the topological properties of tensor sets into properties of the structured matrices that we study here.

Remark. Proofs of Lemmas 2, 3, 6, 7, and 8 are provided in Appendix B.

Lemma 2. *Let $N \geq 3$ and $r \geq 2$. Then, the set $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ is not closed. However, the set of KS matrices $\mathcal{L}^{N,1}_{\mathbf{m},\mathbf{p}}$ and the set $\mathcal{L}^{2,r}_{\mathbf{m},\mathbf{p}}$ are closed.*

To illustrate the non-closedness of $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ for $N \geq 3$ and $r \geq 2$ and motivate the use of the sets $\overline{\mathcal{K}}^{N,r}_{\mathbf{m},\mathbf{p}}$ and ${}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$ in lieu of $\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$, we provide an example.

Example. Consider the sequence $\mathbf{D}_t := t\left(\mathbf{A}_1 + \frac{1}{t}\mathbf{B}_1\right) \otimes \left(\mathbf{A}_2 + \frac{1}{t}\mathbf{B}_2\right)\otimes\left(\mathbf{A}_3 + \frac{1}{t}\mathbf{B}_3\right)\big) - t\mathbf{A}_1\otimes\mathbf{A}_2\otimes\mathbf{A}_3$ where $\mathbf{A}_i, \mathbf{B}_i \in \mathbb{R}^{m_i\times p_i}$ are linearly independent pairs. Here, $\mathfrak{R}^3_{\mathbf{m},\mathbf{p}}(\mathbf{D}_t) \leq 2$ for any t. The limit point of this sequence is $\lim_{t\to\infty} \mathbf{D}_t = \mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \mathbf{B}_3 + \mathbf{A}_1 \otimes \mathbf{B}_2 \otimes \mathbf{A}_3 + \mathbf{B}_1 \otimes \mathbf{A}_2 \otimes \mathbf{B}_3$, which is a separation-rank-3 matrix. Hence, the set $\mathcal{L}^{3,2}_{\mathbf{m},\mathbf{p}}$ is not closed.

The non-closedness of $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ means there exist sequences in $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ whose limit points are not in the set. Two possible solutions to circumvent this issue include: (i) use the closure of $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ as the constraint set, and (ii) eliminate such sequences from $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$. We discuss each solution in detail below.

a) Adding the limit points: We denote the closure of $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ by $\overline{\mathcal{L}}^{N,r}_{\mathbf{m},\mathbf{p}} \triangleq \mathrm{cl}(\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}})$. By slightly relaxing the constraint set in (7) to $\overline{\mathcal{L}}^{N,r}_{\mathbf{m},\mathbf{p}} \cap \mathcal{D}$, we can instead solve the following:

$$\min_{\mathbf{D}\in\overline{\mathcal{K}}^{N,r}_{\mathbf{m},\mathbf{p}}} F_{\mathbf{Y}}(\mathbf{D}), \tag{12}$$

[4] Having a minimum $\mathbf{D}^*$ on the boundary is not sufficient. If the minimizer of $f_{\mathcal{P}}(\mathbf{D})$ over $\mathcal{B}_\rho$ is on the boundary of $\mathcal{B}_\rho$, the value of $f_{\mathcal{P}}(\mathbf{D})$ in the neighborhood of $\mathbf{D}^*$ outside $\mathcal{B}_\rho$ can be smaller than $f_{\mathcal{P}}(\mathbf{D}^*)$; therefore, $\mathbf{D}^*$ is not necessarily a local minimum of $\mathbf{D} \in \mathcal{C} \mapsto f_{\mathcal{P}}(\mathbf{D})$.

[5] If the boundary is empty, it is trivial that the infimum is attained in the interior of the set.

where $\overline{\mathcal{K}}_{\mathbf{m},\mathbf{p}}^{N,r} = \overline{\mathcal{L}}_{\mathbf{m},\mathbf{p}}^{N,r} \cap \mathcal{D}$. Note that (i) a solution to (7) is a solution to (12) and (ii) a solution to (12) is either a solution to (7) or is arbitrarily close to a member of $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$.[6]

b) Eliminating the problematic sequences: In order to exclude the sequences $\mathbf{D}_t \to \mathbf{D}$ such that $\mathbf{D}_t \in \mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r}$ for all t and $\mathbf{D} \notin \mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r}$, we first need to characterize them.

Lemma 3. *Assume $\mathbf{D}_t \to \mathbf{D}$ where $\mathfrak{R}_{\mathbf{m},\mathbf{p}}^{N}(\mathbf{D}_t) \leq r$ and $\mathfrak{R}_{\mathbf{m},\mathbf{p}}^{N}(\mathbf{D}) > r$. We can write $\mathbf{D}_t = \sum_{k=1}^{r} \lambda_t^k \bigotimes_{n=1}^{N} [\mathbf{D}_n^k]_t$ where $\left\|[\mathbf{D}_n^k]_t\right\|_F = 1$. Then, $\max_k |\lambda_t^k| \to \infty$ as $t \to \infty$. In fact, at least two of the coefficient sequences λ_t^k are unbounded.*

The following corollary of Lemma 3 suggests that one can exclude the problematic sequences from $\mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r}$ by bounding the norm of individual KS (separation-rank-1) terms.

Corollary 1. *Consider the set $\mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r}$ whose members can be written as $\mathbf{D} = \sum_{k=1}^{r} \bigotimes_{n=1}^{N} \mathbf{D}_n^k$ such that $\mathbf{D}_n^k \in \mathbb{R}^{m_n \times p_n}$. Then, for any $c > 0$ the set ${}^{c}\mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r} = \big\{\mathbf{D} \in \mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r} \big| \big\| \bigotimes \mathbf{D}_n^k \big\|_F \leq c\big\}$ is closed.*

We have now shown that the sets $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r}$, $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N} \triangleq \mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,1}$, ${}^{c}\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r} = {}^{c}\mathcal{L}_{\mathbf{m},\mathbf{p}}^{N,r} \cap \mathcal{D}$, and $\overline{\mathcal{K}}_{\mathbf{m},\mathbf{p}}^{N,r} = \overline{\mathcal{L}}_{\mathbf{m},\mathbf{p}}^{N,r} \cap \mathcal{D}$ are compact subsets of $\mathbb{R}^{m \times p}$.

B. Asymptotic Analysis for Dictionary Identifiability

Now that we have discussed the compactness of the relevant constraint sets, we are ready to show that the minimum of $f_{\mathbf{y}}(\mathbf{D})$ over $\mathcal{B}_\rho$, defined in (10), is not attained on $\partial\mathcal{B}_\rho$. This will complete our proof of existence of a local minimum of $f_{\mathcal{P}}(\mathbf{D})$ in $\mathcal{B}_\rho$. In our proof, we make use of a result in Gribonval et al. [27], presented here in Lemma 4.

Lemma 4 (Theorem 1 in Gribonval et al. [27])**.** *Consider the statistical DL Problem* (2) *with constraint set $\mathcal{D}$. Suppose the generating dictionary $\mathbf{D}^0 \in \mathcal{D}$ and Assumptions 1–3 hold. Then, for any ρ such that $\bar{\lambda} C_{\min} < \rho \leq \bar{\lambda} C_{\max}$ and $\frac{M_\epsilon}{M_x} < \frac{7}{2}(\bar{\lambda} C_{max} - \rho)$, we have*

$$\Delta f_{\mathcal{P}}(\rho) \geq \frac{\mathbb{E}\{x^2\}}{8} \cdot \frac{s}{p} \cdot \rho \left(\rho - \bar{\lambda} C_{\min}\right) > 0. \tag{13}$$

for all $\mathbf{D} \in \mathcal{D}$ such that $\|\mathbf{D} - \mathbf{D}^0\|_F = \rho$.

Interested readers can find the detailed proof of Lemma 4 in Gribonval et al. [27]. The following theorem states our first identifiability result for the LSR-DL model.

Theorem 1. *Consider the statistical DL Problem* (2) *with constraint set $\mathcal{C}$ being either $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r}$, $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N}$, ${}^{c}\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ or $\overline{\mathcal{K}}_{\mathbf{m},\mathbf{p}}^{N,r}$. Suppose the generating dictionary $\mathbf{D}^0 \in \mathcal{C}$ and Assumptions 1–3 hold. Then, for any ρ such that $\bar{\lambda} C_{\min} < \rho < \bar{\lambda} C_{\max}$ and $\frac{M_\epsilon}{M_x} < \frac{7}{2}(\bar{\lambda} C_{max} - \rho)$, the function $\mathbf{D} \in \mathcal{C} \mapsto f_{\mathcal{P}}(\mathbf{D})$ has a local minimum $\mathbf{D}^*$ such that $\left\|\mathbf{D}^* - \mathbf{D}^0\right\|_F < \rho$.*

Proof. Since $f_{\mathcal{P}}(\mathbf{D})$ is a continuous function and the ball $\mathcal{B}_\rho = \{\mathbf{D} \in \mathcal{C} | \left\|\mathbf{D} - \mathbf{D}^0\right\|_F \leq \rho\}$ is compact, the function $\mathbf{D} \in \mathcal{B}_\rho \mapsto f_{\mathcal{P}}(\mathbf{D})$ attains its infimum at a point in the ball. If this minimum is attained in the interior of $\mathcal{B}_\rho$ then it is a local minimum of $\mathbf{D} \in \mathcal{C} \mapsto f_{\mathcal{P}}(\mathbf{D})$. Therefore, a key ingredient of the proof is showing that $f_{\mathcal{P}}(\mathbf{D}) > f_{\mathcal{P}}(\mathbf{D}^0)$ for all $\mathbf{D} \in \partial\mathcal{B}_\rho$ if $\partial\mathcal{B}_\rho$ is nonempty. Lemma 4 states the conditions under which $f_{\mathcal{P}}(\mathbf{D}) > f_{\mathcal{P}}(\mathbf{D}^0)$ on $\partial\mathcal{S}_\rho$, where $\mathcal{S}_\rho \triangleq \big\{\mathbf{D} \in \mathcal{D} \mid \left\|\mathbf{D} - \mathbf{D}^0\right\|_F \leq \rho\big\}$.

Since $\partial\mathcal{B}_\rho \subset \partial\mathcal{S}_\rho$, the result of Lemma 4 can be used for our problem as well, i.e. for any $\mathbf{D} \in \partial\mathcal{B}_\rho$, we have $f_{\mathcal{P}}(\mathbf{D}) > f_{\mathcal{P}}(\mathbf{D}^0)$, when $C_{\min}\bar{\lambda} < \rho < C_{\max}\bar{\lambda}$. It follows from this result together with the existence of the infimum of $f_{\mathcal{P}}(\mathbf{D}) : \mathcal{B}_\rho \mapsto \mathbb{R}$ in $\mathcal{B}_\rho$ that Problem (2) has a local minimum within a ball of radius ρ around the true dictionary $\mathbf{D}^0$. □

In Theorem 1, we guarantee that the true dictionary, $\mathbf{D}^0$, is identifiable as a solution to the statistical rank-constrained LSR-DL problem (2). Next, we take advantage of concentration of measure inequalities that relate the empirical objective in (4) to the statistical objective studied in Theorem 1 to find the number of samples needed to ensure $\mathbf{D}^0$ is also identifiable via the empirical rank-constrained LSR-DL problem (2).

C. Sample Complexity for Dictionary Identifiability

We now derive the number of samples required to guarantee, with high probability, that $F_{\mathbf{Y}} : \mathcal{C} \mapsto \mathbb{R}$ has a local minimum at a point "close" to $\mathbf{D}^0$ when the constraint set $\mathcal{C}$ is either $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r}$, $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N}$, or ${}^{c}\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ for $N \geq 3$ and $r \geq 2$. First, we use concentration of measure inequalities based on the covering number of the dictionary class $\mathcal{C} \subset \mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ to show that the empirical loss $F_{\mathbf{Y}}(\mathbf{D})$ uniformly converges to its expectation $f_{\mathcal{P}}(\mathbf{D})$ with high probability. This is formalized below.

Lemma 5 (Theorem 1 and Lemma 11, Gribonval et al. [35])**.** *Consider the empirical DL Problem* (4) *and suppose Assumptions 1 and 2 are satisfied. For any $u \geq 0$ and constants $c_1 \geq M_y^2/\sqrt{8}$ and $c_2 \geq \max(1, \log c_0\sqrt{8}M_y)$, with probability at least $1 - 2e^{-u}$ we have*

$$\sup_{\mathbf{D} \in \mathcal{C}} |F_{\mathbf{Y}}(\mathbf{D}) - f_{\mathcal{P}}(\mathbf{D})| \leq 3c_1\sqrt{\frac{c_2\nu\log L}{L}} + c_1\sqrt{\frac{c_2\nu + u}{L}}, \tag{14}$$

where ν is such that $\mathcal{N}_{2,\infty}(\mathcal{C}, \epsilon) = \left(\frac{c_0}{\epsilon}\right)^{\nu}$.

Define $\eta_L \triangleq 3c_1\sqrt{\frac{c_2\nu\log L}{L}} + c_1\sqrt{\frac{c_2\nu+u}{L}}$. It follows from (14) that with high probability (w.h.p.),

$$\Delta F_{\mathbf{Y}}(\mathbf{D}; \mathbf{D}^0) \geq \Delta f_{\mathcal{P}}(\mathbf{D}; \mathbf{D}^0) - 2\eta_L, \tag{15}$$

for all $\mathbf{D} \in \mathcal{C}$. Therefore, when $\eta_L < \Delta f_{\mathcal{P}}(\mathbf{D}; \mathbf{D}^0)/2$ for all $\mathbf{D} \in \partial\mathcal{B}_\rho$, we have $\Delta F_{\mathbf{Y}}(\mathbf{D}; \mathbf{D}^0) > 0$ for all $\mathbf{D} \in \partial\mathcal{B}_\rho$. In this case, we can use similar arguments as in the asymptotic analysis to show that $F_{\mathbf{Y}} : \mathcal{C} \to \mathbb{R}$ has a local minimum at a point in the interior of $\mathcal{B}_\rho$. Hence, our focus in this section is on finding the sample complexity L required to guarantee that $\eta_L \leq \Delta f_{\mathcal{P}}(\rho)/2$ w.h.p. We begin with characterization of covering numbers of the three constraint sets, which may also be of independent interest to some readers.

[6] The first argument holds since if $F_{\mathbf{Y}}(\mathbf{D}^*) \leq F_{\mathbf{Y}}(\mathbf{D})$ for all $\mathbf{D} \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$, by continuity it also holds for all $\mathbf{D} \in \overline{\mathcal{K}}_{\mathbf{m},\mathbf{p}}^{N,r}$. The second argument is trivial.

Covering Numbers: The covering number of the set $\mathcal{K}^N_{\mathbf{m},\mathbf{p}}$ with respect to the norm $\|\cdot\|_{2,\infty}$ is known in the literature to be upper bounded as follows [35]:

$$\mathcal{N}_{2,\infty}(\mathcal{K}^N_{\mathbf{m},\mathbf{p}},\epsilon) \leq (3/\epsilon)^{\sum_{i=1}^N m_i p_i}. \tag{16}$$

We now turn to finding the covering numbers of LSR sets $\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}$ and ${}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$. The following lemma establishes a bound on covering number of $\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}$, which depends on the separation rank r exponentially.

Lemma 6. *The covering number of the set $\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}$ with respect to the norm $\|\cdot\|_{2,\infty}$ is upper bounded as follows:*

$$\mathcal{N}_{2,\infty}(\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}},\epsilon) \leq (9p/\epsilon)^{r(m_1p_1+m_2p_2+1)}.$$

Next, we obtain an upper bound on the covering number of ${}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$ for a given constant c.

Lemma 7. *The covering number of the set ${}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$ with respect to the max-column norm $\|\cdot\|_{2,\infty}$ is bounded as follows:*

$$\mathcal{N}_{2,\infty}({}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}},\epsilon) \leq (3rc/\epsilon)^{r\sum_{i=1}^N m_i p_i}.$$

We can now find the sample complexity of the LSR-DL Problem (4) by plugging in the values of ν and c_0 in Lemma 5.

Theorem 2. *Consider the empirical LSR dictionary learning Problem* (4) *with constraint set $\mathcal{C}$ being $\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}$, $\mathcal{K}^N_{\mathbf{m},\mathbf{p}}$, or ${}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$. Fix any $u > 0$. Suppose the generating dictionary $\mathbf{D}^0 \in \mathcal{C}$ and Assumptions 1–3 are satisfied. Assume $\bar{\lambda}C_{\min} < \rho < \bar{\lambda}C_{\max}$ and $\frac{M_\epsilon}{M_x} < \frac{7}{2}(\bar{\lambda}C_{max} - \rho)$. Define a constant ν that depends on the dictionary class:*

- $\nu = \sum_{i=1}^N m_i p_i$ *and* $c_0 = 3$ *when* $\mathcal{C} = \mathcal{K}^N_{\mathbf{m},\mathbf{p}}$,
- $\nu = 2r(m_1p_1 + m_2p_2 + 1)$ *and* $c_0 = 9p$ *when* $\mathcal{C} = \mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}$,
- $\nu = r\sum_{i=1}^N m_i p_i$ *and* $c_0 = rc$ *when* $\mathcal{C} = {}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$.

Then, given a number of samples L satisfying

$$\frac{L}{\log L} \geq Cp^2\left(\nu\log c_0 + u\right)\frac{M_y^4}{\left(\rho\left(\rho - \bar{\lambda}C_{\min}\right)s\mathbb{E}\{x^2\}\right)^2} \tag{17}$$

where C is a constant, with probability no less than $1 - e^{-u}$, the empirical risk objective function $\mathbf{D} \in \mathcal{C} \mapsto F_{\mathbf{Y}}(\mathbf{D})$ has a local minimizer $\mathbf{D}^*$ such that $\left\|\mathbf{D}^* - \mathbf{D}^0\right\|_F < \rho$.

Proof. We take a similar approach to the proof of Theorem 1. Due to compactness of the ball $\mathcal{B}_\rho = \{\mathbf{D} \in \mathcal{C} | \left\|\mathbf{D} - \mathbf{D}^0\right\|_F \leq \rho\}$ and continuity of $F_{\mathbf{Y}}(\mathbf{D})$, it follows that $\mathbf{D} \in \mathcal{B}_\rho \mapsto F_{\mathbf{Y}}(\mathbf{D})$ attains its minimum at a point in $\mathcal{B}_\rho$. It remains to show that $\Delta F_{\mathbf{Y}}(\mathbf{D};\mathbf{D}^0) > 0$ for all $\mathbf{D} \in \partial\mathcal{B}_\rho$ which implies existence of a local minimizer of $F_{\mathbf{Y}} : \mathcal{C} \to \mathbb{R}$ at $\mathbf{D}^*$ such that $\left\|\mathbf{D}^* - \mathbf{D}^0\right\|_F < \rho$.

Inequality (15) shows that it suffices to set $\eta_L \leq \Delta f_{\mathcal{P}}(\mathbf{D};\mathbf{D}^0)/2$ to have $\Delta F_{\mathbf{Y}}(\mathbf{D};\mathbf{D}^0) > 0$. From Lemma 5 we know $\eta_L \geq 3c_1\sqrt{\frac{c_2\nu\log L}{L}} + c_1\sqrt{\frac{c_2\nu+u}{L}}$. Therefore, using the lower bound (13) on $\Delta f_{\mathcal{P}}(\rho)$ we have with probability at least $1 - e^{-u}$

$$3c_1\sqrt{\frac{c_2\nu\log L}{L}} + c_1\sqrt{\frac{c_2\nu+u}{L}} \leq \frac{\mathbb{E}\{x^2\}}{16}\cdot\frac{s}{p}\cdot\rho\left(\rho - \bar{\lambda}C_{\min}\right)$$

with $c_1 \geq M_y^2/\sqrt{8}$ and $c_2 \geq \max(1, \log c_0\sqrt{8}M_y)$[7]. Rearranging, we get

$$\frac{L}{\log L} \geq c_1^2\left(\frac{3\sqrt{c_2\nu} + \sqrt{c_2\nu+u}}{\rho\left(\rho - \bar{\lambda}C_{\min}\right)}\right)^2\left(\frac{16}{\mathbb{E}\{x^2\}}\cdot\frac{p}{s}\right)^2. \tag{18}$$

Setting $c_1 \geq M_y^2/\sqrt{8}$ and $c_2 = c_3\log c_0 \geq \max(1, \log c_0\sqrt{8}M_y)$ we get the lower bound

$$\frac{L}{\log L} \geq Cp^2\left(\nu\log c_0 + u\right)\left(\frac{M_y^2}{\rho\left(\rho - \bar{\lambda}C_{\min}\right)s\mathbb{E}\{x^2\}}\right)^2$$

with probability at least $1 - e^{-u}$. Given that the number of samples satisfies (17) for $\bar{\lambda}C_{\min} < \rho < \bar{\lambda}C_{\max}$, with high probability $\Delta F_{\mathbf{Y}} > 0$ for any $\mathbf{D} \in \partial\mathcal{B}_\rho$. Therefore, it follows from the existence of the infimum of $\mathbf{D} \in \mathcal{B}_\rho \mapsto F_{\mathbf{Y}}(\mathbf{D})$ in $\mathcal{B}_\rho$ that $\mathbf{D} \in \mathcal{C} \mapsto F_{\mathbf{Y}}(\mathbf{D})$ has a local minimum at a point within a ball of radius ρ around the true dictionary $\mathbf{D}^0$. □

The $\Omega\big(r(\sum_n m_n p_n)p^2\rho^{-2}\big)$ sample complexity upper bound we obtain here for rank-constrained LSR-DL is a reduction compared to the $\Omega(mp^3\rho^{-2})$ sample complexity of standard DL [27]. However, the minimax lower bound scaling of $\Omega(p\sum_n m_n p_n \rho^{-2})$ for KS-DL [15] ($r = 1$) indicates an $O(p)$ gap with our sample complexity upper bound. This gap could be due to looseness in the lower bound, our upper bound, or both. We leave an investigation of this and possible tightening of the bound(s) to future work.

IV. Identifiability in the Tractable LSR-DL Problems

In Section II, we introduced two tractable relaxations to the rank-constrained LSR-DL problem: a regularized problem (8) with a convex regularization term and a factorization-based problem (9) in which the dictionary is written in terms of its subdictionaries. We now provide results on the local identifiability of the true dictionary $\mathbf{D}^0$ in these problems, i.e., we find conditions under which at least one local minimizer of these problems is located near the true dictionary $\mathbf{D}^0$. Such local identifiability result implies that any DL algorithm that converges to a local minimum of these problems can recover $\mathbf{D}^0$ up to a small error if it is initialized close enough to $\mathbf{D}^0$.

A. Regularization-based LSR Dictionary Learning

The first tractable LSR-DL problem that we study is the regularized problem (8). Exploiting the relation between $\mathfrak{R}^N_{\mathbf{m},\mathbf{p}}(\mathbf{D})$ and $\operatorname{rank}(\underline{\mathbf{D}}^\pi)$, the LSR structure is enforced on the dictionary by a convex regularizer that imposes low tensor rank structure on $\underline{\mathbf{D}}^\pi$. The regularizer that we use here is a commonly used convex proxy for the tensor rank function, the *sum-trace-norm* [36], which is defined as the average of the trace (nuclear) norms of the *unfoldings* of the tensor: $\|\underline{\mathbf{A}}\|_{\text{str}} \triangleq \sum_{n=1}^N \left\|\mathbf{A}^{(n)}\right\|_{\text{tr}}$.

The first question we address is whether the reference dictionary that generates the observations $\{\underline{\mathbf{Y}}_l\}_{l=1}^L$ is identifiable

[7] Under the conditions of this theorem, $M_y \leq \sqrt{1+\delta_s(\mathbf{D}^0)}M_x + M_\epsilon$, where $\delta_s(\mathbf{D}^0)$ denotes the RIP constant of $\mathbf{D}^0$.

via Problem (8). Our local identifiability result here is limited to when $\mathbf{D}^0 \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^N$, i.e. the true dictionary is KS. For such $\mathbf{D}^0$, we show that there is at least one local minimizer $\mathbf{D}^*$ of $F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D})$ under Assumptions 1–3 that is close to $\mathbf{D}^0$.

Theorem 3. *Consider the regularized LSR-DL problem* (8)*. Suppose that the generating dictionary* $\mathbf{D}^0 \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^N$ *and Assumptions 1–3 are satisfied. Moreover, let* $\bar{\lambda} C_{\min} < \rho \leq \bar{\lambda} C_{\max}$ *and* $\frac{M_\epsilon}{M_x} < \frac{7}{2}(\bar{\lambda} C_{max} - \rho)$*. Then, the expected risk function* $\mathbf{D} \in \mathcal{D} \mapsto \mathbb{E}[F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D})]$ *has a local minimizer* $\mathbf{D}^*$ *such that* $\left\|\mathbf{D}^* - \mathbf{D}^0\right\|_F \leq \rho$*.*

Moreover, given L *samples such that*

$$L > C_0 p^2 (mp + u) \left(\frac{M_x^2}{\mathbb{E}x^2} \cdot \frac{\frac{M_\epsilon}{M_x} + \rho + (\frac{M_\epsilon}{M_x} + \rho)^2}{\rho - C_{\min}\bar{\lambda}} \right)^2, \quad (19)$$

where u *and* C_0 *are positive constants, then, we have with probability no less than* $1 - e^{-u}$ *that the empirical risk function* $\mathbf{D} \in \mathcal{D} \mapsto F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D})$ *has a local minimum at* $\mathbf{D}^*$ *such that* $\left\|\mathbf{D}^* - \mathbf{D}^0\right\|_F < \rho$*.*

Proof. Consider the ball $\mathcal{B}_\rho = \{\mathbf{D} \in \mathcal{D} | \left\|\mathbf{D} - \mathbf{D}^0\right\|_F \leq \rho\}$. Compactness of $\mathcal{B}_\rho = \{\mathbf{D} \in \mathcal{C} | \left\|\mathbf{D} - \mathbf{D}^0\right\|_F \leq \rho\}$ and continuity of $F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D})$ guarantee that $\mathbf{D} \in \mathcal{B}_\rho \mapsto F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D})$ attains its minimum at a point in $\mathcal{B}_\rho$. Similarly, $\mathbf{D} \in \mathcal{B}_\rho \mapsto f_{\mathcal{P}}^{\text{reg}}(\mathbf{D}) \triangleq \mathbb{E}[F_{\mathbf{Y}}^{\text{reg}}]$ reaches its minimum at a point in $\mathcal{B}_\rho$. We now need to show in either case the minimum is not attained on the boundary of $\mathcal{B}_\rho$. To this end, we show in the following that $\Delta F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D}; \mathbf{D}^0) > 0$ and $\Delta f_{\mathcal{P}}^{\text{reg}}(\mathbf{D}; \mathbf{D}^0) > 0$ for any $\mathbf{D} \in \partial \mathcal{B}_\rho$.

Incorporation of the trace-norm regularization term in (8) within the objective in (4) introduces a factor $\|\underline{\mathbf{D}}^\pi\|_{\text{str}} - \left\|[\underline{\mathbf{D}}^0]^\pi\right\|_{\text{str}} = \sum_{n=1}^N \left(\left\|\mathbf{D}^{(n)}\right\|_{\text{tr}} - \left\|[\mathbf{D}^0]^{(n)}\right\|_{\text{tr}}\right)$ to $\Delta f_{\mathcal{P}}(\mathbf{D}; \mathbf{D}^0)$ and $\Delta F_{\mathbf{Y}}(\mathbf{D}; \mathbf{D}^0)$. We know from Lemma 1 that when the true dictionary is a KS matrix ($\mathbf{D}^0 \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^N$), its rearrangement tensor $[\underline{\mathbf{D}}^0]^\pi$ is a rank-1 tensor and therefore all unfoldings $[\mathbf{D}^0]^{(n)}$ of $[\underline{\mathbf{D}}^0]^\pi$ are rank-1 matrices. This implies $\|[\mathbf{D}^0]^{(n)}\|_{\text{tr}} = \|[\mathbf{D}^0]^{(n)}\|_F$. Moreover, for all $\mathbf{D} \in \mathcal{D}_{m \times p}$ we have $\left\|\mathbf{D}^{(n)}\right\|_F = \left\|[\mathbf{D}^0]^{(n)}\right\|_F = \sqrt{p}$. Therefore,

$$\begin{aligned} \left\|\mathbf{D}^{(n)}\right\|_{\text{tr}} - \left\|[\mathbf{D}^0]^{(n)}\right\|_{\text{tr}} &= \sum\nolimits_{k=1}^{r_n} \sigma_k(\mathbf{D}^{(n)}) - \sqrt{p} \\ &\geq \sqrt{\sum\nolimits_{k=1}^{r_n} \sigma_k^2(\mathbf{D}^{(n)})} - \sqrt{p} = 0, \end{aligned}$$

where $r_n \triangleq \text{rank}(\mathbf{D}^{(n)})$ and $\sigma_k(\mathbf{D}^{(n)})$ denotes the k-th singular value of $\mathbf{D}^{(n)}$. Therefore, we conclude that $\Delta F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D}; \mathbf{D}^0) \geq \Delta F_{\mathbf{Y}}(\mathbf{D}; \mathbf{D}^0)$ and $\Delta f_{\mathcal{P}}^{\text{reg}}(\mathbf{D}; \mathbf{D}^0) \geq \Delta f_{\mathcal{P}}(\mathbf{D}; \mathbf{D}^0)$ for any $\mathbf{D} \in \mathcal{D}$. According to Lemma 4, $\Delta f_{\mathcal{P}}(\mathbf{D}; \mathbf{D}^0) > 0$ for all $\mathbf{D}$ on the boundary of the ball $\mathcal{B}_\rho$. Furthermore, under the assumptions of the current theorem, given a number of samples satisfying (19), Gribonval et al. [27] show that the empirical difference $\Delta F_{\mathbf{Y}}(\mathbf{D}; \mathbf{D}^0) > 0$ for all $\mathbf{D}$ on the boundary of $\mathcal{S}_\rho = \left\{\mathbf{D} \in \mathcal{D} \;\middle|\; \left\|\mathbf{D} - \mathbf{D}^0\right\|_F \leq \rho\right\}$, and therefore on the boundary of $\mathcal{B}_\rho \subseteq \mathcal{S}_\rho$, with probability at least $1 - e^{-u}$. Therefore, for both $f_{\mathcal{P}}^{\text{reg}}(\mathbf{D})$ and $F_{\mathbf{Y}}^{\text{reg}}(\mathbf{D})$, the minimum is attained in the interior of $\mathcal{B}_\rho$ and not on its boundary. □

We discuss the implications of Theorem 3 in Section IV-C.

B. Factorization-based LSR Dictionary Learning

We now shift our focus to Problem (9), which expands $\mathbf{D}$ as $\sum_{k=1}^r \bigotimes \mathbf{D}_n^k$ and optimizes over the individual subdictionaries, and show that there is at least one local minimum $\{[\mathbf{D}_n^k]^*\}$ of the factorization-based LSR-DL Problem (9) such that $\sum \bigotimes [\mathbf{D}_n^k]^*$ is close to the underlying dictionary $\mathbf{D}^0$. Our strategy here is to establish a connection between the local minima of (9) and those of (4). Specifically, we show that when the dictionary class in (7) matches that of (9), for every local minimum $\widehat{\mathbf{D}}$ of (4), there exists a local minimum $\{\widehat{\mathbf{D}}_n^k\}$ of (9) such that $\widehat{\mathbf{D}} = \sum \bigotimes \widehat{\mathbf{D}}_n^k$. Furthermore, we use the result of Theorems 1 and 2 that there exists a local minimum $\mathbf{D}^*$ of Problem (4) within a small ball around $\mathbf{D}^0$. It follows from these facts that under the generating model considered here, a local minimum $\{[\mathbf{D}_n^k]^*\}$ of (9) is such that $\sum \bigotimes [\mathbf{D}_n^k]^*$ is close to $\mathbf{D}^0$.

We begin with a bound on the distance between LSR matrices when the tuples of their factor matrices are ϵ-close.

Lemma 8. *For any two tuples* $(\mathbf{A}_n^k)$ *and* $(\mathbf{B}_n^k)$ *such that* $\mathbf{A}_n^k, \mathbf{B}_n^k \in \alpha \mathcal{U}_{m_n \times p_n}$ *for all* $n \in [N]$ *and* $k \in [r]$*, if the distance* $\left\|(\mathbf{A}_n^k) - (\mathbf{B}_n^k)\right\|_F \leq \epsilon$ *then* $\left\|\sum_{k=1}^r \bigotimes \mathbf{A}_n^k - \sum_{k=1}^r \bigotimes \mathbf{B}_n^k\right\|_F \leq \alpha^{N-1}\sqrt{Nr}\epsilon$*.*

Theorem 4. *Consider the factorization-based LSR-DL problem* (9)*. Suppose that Assumptions 1–3 are satisfied and* $\frac{M_\epsilon}{M_x} < \frac{7}{2}(\bar{\lambda} C_{max} - \rho)$ *with* $\bar{\lambda} C_{\min} < \rho \leq \bar{\lambda} C_{\max}$*. Then, the expected risk function* $\mathbb{E}[F_{\mathbf{Y}}^{\text{fac}}(\{\mathbf{D}_n^k\})]$ *has a local minimizer* $([\mathbf{D}_n^k]^*)$ *such that* $\left\|\sum \bigotimes [\mathbf{D}_n^k]^* - \mathbf{D}^0\right\|_F \leq \rho$*.*

Moreover, when the sample complexity requirements (17) *are satisfied for some positive constant* u*, then with probability no less than* $1 - e^{-u}$ *the empirical risk objective function* $F_{\mathbf{Y}}^{\text{fac}}(\{\mathbf{D}_n^k\})$ *has a local minimum achieved at* $([\mathbf{D}_n^k]^*)$ *such that* $\left\|\sum \bigotimes [\mathbf{D}_n^k]^* - \mathbf{D}^0\right\|_F \leq \rho$*.*

Proof. Let us first consider the finite sample case. Theorem 2 shows existence of a local minimizer $\mathbf{D}^*$ of Problem (7) for constraint sets $\mathcal{K}_{\mathbf{m},\mathbf{p}}^N$, $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r}$, and ${}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$, such that $\|\mathbf{D}^* - \mathbf{D}^0\|_F \leq \rho$ w.h.p. Here, we want to show that for such $\mathbf{D}^*$, there exists a $\{[\mathbf{D}_n^k]^*\}$ such that $\mathbf{D}^* = \sum \bigotimes [\mathbf{D}_n^k]^*$ and $\{[\mathbf{D}_n^k]^*\}$ is a local minimizer of Problem (9).

First, let us consider Problem (7) with ${}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$. It is easy to show that any $\mathbf{D} \in {}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ can be written as $\sum_{k=1}^r \bigotimes \mathbf{D}_n^k$ such that for all $k \in [r]$ and $n \in [N]$, without loss of generality, $\mathbf{D}_n^k \in \alpha \mathcal{U}_{m_n \times p_n}$ where $\alpha > \sqrt[N-1]{c}$. Define $\mathcal{C}^{\text{fac}} \triangleq \left\{(\mathbf{D}_n^k) \middle| \sum \bigotimes \mathbf{D}_n^k \in {}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r} : \forall k, n, \mathbf{D}_n^k \in \alpha \mathcal{U}_{m \times p}\right\}$. Since $\mathbf{D}^* \in {}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$, there is a $([\mathbf{D}_n^k]^*) \in \mathcal{C}^{\text{fac}}$ (with $\|[\mathbf{D}_n^k]^*\|_F = \sqrt[N-1]{c}$ for all $k \in [r]$ and $n \in [N]$) such that $\mathbf{D}^* = \sum \bigotimes [\mathbf{D}_n^k]^*$. According to Lemma 8, for any $\{\mathbf{D}_n^k\} \in \mathcal{C}^{\text{fac}}$ it follows from $\left\|(\mathbf{D}_n^k) - ([\mathbf{D}_n^k]^*)\right\|_F \leq \epsilon'$ that $\left\|\sum \bigotimes \mathbf{D}_n^k - \sum \bigotimes [\mathbf{D}_n^k]^*\right\|_F \leq \alpha^{N-1}\sqrt{Nr}\epsilon'$. Since $\mathbf{D}^*$ is a local minimizer of (7), there exists a positive ϵ such that for all $\mathbf{D} \in {}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$ satisfying $\|\mathbf{D} - \mathbf{D}^*\|_F \leq \epsilon$, we have $F_{\mathbf{Y}}(\mathbf{D}^*) \leq F_{\mathbf{Y}}(\mathbf{D})$. If we choose ϵ' small enough such that $\sqrt[N-1]{c} + \epsilon' \leq \alpha$ and $\alpha^{N-1}\sqrt{Nr}\epsilon' \leq \epsilon$, then for any $(\mathbf{D}_n^k)$ such that $(\mathbf{D}_n^k) \in \mathcal{C}^{\text{fac}}$ and $\left\|(\mathbf{D}_n^k) - ([\mathbf{D}_n^k]^*)\right\|_F \leq \epsilon'$, we have $\left\|\sum \bigotimes \mathbf{D}_n^k - \mathbf{D}^*\right\|_F \leq \epsilon$ and this means that $F_{\mathbf{Y}}^{\text{fac}}(\{\mathbf{D}_n^k\}) - F_{\mathbf{Y}}^{\text{fac}}(\{[\mathbf{D}_n^k]^*\}) = F_{\mathbf{Y}}(\sum \bigotimes \mathbf{D}_n^k) - F_{\mathbf{Y}}(\mathbf{D}^*) \geq 0$. Therefore,

$([\mathbf{D}_n^k]^*)$ is a local minimizer of Problem (9). This concludes our proof for the finite sample case with constraint set ${}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$.

Note that we can write $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N} = {}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,1}$ and $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r} = {}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r}$ with $c \geq p$. Therefore, the above results also hold for $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N}$ and $\mathcal{K}_{\mathbf{m},\mathbf{p}}^{2,r}$ since they are special cases of ${}^c\mathcal{K}_{\mathbf{m},\mathbf{p}}^{N,r}$.

It is easy to see similar relation exists between the local minima of $f_{\mathbf{y}}(\mathbf{D})$ and $f_{\mathbf{y}}^{\text{fac}}(\{\mathbf{D}_n^k\}) \triangleq \mathbb{E}[F_{\mathbf{Y}}^{\text{fac}}(\{\mathbf{D}_n^k\})]$, proving the asymptotic result in the statement of this theorem. □

C. Discussion

In this section, we discuss the local identifiability of the true dictionary in the regularization-based formulation (Theorem 3) and the factorization-based formulation (Theorem 4). For the regularization-based formulation, our results only hold for the case where the true dictionary is KS, i.e. $\mathbf{D}^0 \in \mathcal{K}_{\mathbf{m},\mathbf{p}}^{N}$. We obtain sample complexity requirement of $\Omega(mp^3\rho^{-2})$ in this case, which matches the sample complexity requirement of the unstructured formulation [27]. This is due to the fact that the class of dictionaries in the regularization-based formulation is $\mathcal{D}_{m\times p}$, i.e., the LSR constraint is not explicitly imposed in this formulation. Thus, our covering number-based approach does not provide improved sample complexity results compared to the unstructured formulation. The experimental results of the regularization-based algorithm STARK (see Section VI) suggest there is room to improve our sample complexity result for the regularized formulation. We leave investigating such improvement as future work. Nonetheless, our results imply well-posedness of the regularized LSR-DL problem.

For the factorization-based formulation, we show that $\Omega(p^2\rho^{-2}r\sum_n m_n p_n)$ samples are required for local identifiability of a dictionary of separation-rank r. This result matches that of our rank-constrained formulation stated in Theorem 2. Note that when the separation rank is 1, this result gives a bound on the sample complexity of the KS-DL model as a special case. To illustrate the implication of our bound, consider the case of $N = 2, m_1 = m_2 = \sqrt{m}$ and $p_1 = p_2 = \sqrt{p}$. In this case, our bound scales as $p^2\sqrt{mp}$, which results in $\sqrt{mp}$ reduction in sample complexity scaling compared to the unstructured DL bound [27]. In the case of $m_1 = m,\ m_2 = 1$ and $p_1 = p,\ p_2 = 1$ (unstructured DL), our bound scales as mp^3, which is consistent with the unstructured DL bound [27]. Note that unlike the KS-DL analysis [16], which shows a necessary sample complexity of $L = \max_{n\in\{1,\dots,N\}}\Omega(m_n p_n^3 \rho_n^{-2})$, our analysis of the factorized model does not ensure identifiability of the true subdictionaries in the LSR-DL model. However, the results for KS-DL require the dictionary coefficient vectors to follow the separable sparsity model. In contrast, our result does not require any constraints on the sparsity pattern of coefficients.

V. Computational Algorithms

In Section IV, we showed that the tractable LSR-DL Problems (8) and (9) each have at least one local minimum close to the true dictionary. In this section we develop algorithms to find these local minima. Solving Problems (8) and (9) require minimization with respect to (w.r.t.) $\mathbf{X} \triangleq [\mathbf{x}_1^T, \cdots, \mathbf{x}_L^T]$. Therefore, similar to conventional DL algorithms, we introduce alternating minimization-type algorithms that at every iteration, first perform minimization of the objective function w.r.t. $\mathbf{X}$ (sparse coding stage) and then minimize the objective w.r.t. the dictionary (dictionary update stage).

Algorithm 1 Dictionary Update in STARK for LSR-DL

```
Require: $\mathbf{Y}$, $\mathbf{\Pi}$, $\lambda_1 > 0$, $\gamma > 0$, $\mathbf{X}(t)$[8]
repeat
    Update $\underline{\mathbf{D}}^\pi$ according to update rule (26)
    for $n \in [N]$ do
        Update $\underline{\mathbf{W}}_n$ according to (27)
    end for
    for $n \in [N]$ do
        $\underline{\mathbf{A}}_n \leftarrow \underline{\mathbf{A}}_n - \gamma\,(\underline{\mathbf{D}}^\pi - \underline{\mathbf{W}}_n)$
    end for
until convergence
Normalize columns of $\mathbf{D}$
return $\mathbf{D}(t+1)$
```

The sparse coding stage is a simple Lasso problem and remains the same in our algorithms. However, the algorithms differ in their dictionary update stages, which we discuss next.

Remark. We leave the formal convergence results of our algorithms to future work. However, we provide a discussion on challenges and possible approaches to establish convergence of our algorithms in Appendix C.

A. STARK: A Regularization-based LSR-DL Algorithm

We first discuss an algorithm, which we term *STructured dictionAry learning via Regularized low-ranK Tensor Recovery (STARK)*, that helps solve the regularized LSR-DL problem given in (8) and discussed in Section IV using the Alternating Direction Method of Multipliers (ADMM) [37].

The main novelty in solving (8) using $g_1(\underline{\mathbf{D}}^\pi) = \|\underline{\mathbf{D}}^\pi\|_{\text{str}}$ is the dictionary update stage. This stage, which involves updating $\mathbf{D}$ for a fixed set of sparse codes $\mathbf{X}$, is particularly challenging for gradient-based methods because the dictionary update involves interdependent nuclear norms of different unfoldings of the rearranged tensor $\underline{\mathbf{D}}^\pi$. Inspired by many works in the literature on low-rank tensor estimation [36], [38], [39], we instead suggest the following reformulation of the dictionary update stage of (8):

$$\begin{aligned} \min_{\mathbf{D}\in\mathcal{D},\underline{\mathbf{W}}_1,\cdots,\underline{\mathbf{W}}_N} \quad & \frac{1}{2}\|\mathbf{Y}-\mathbf{D}\mathbf{X}\|_F^2 + \lambda_1 \sum_{n=1}^{N} \left\| \mathbf{W}_n^{(n)} \right\|_{\text{tr}} \\ \text{s.t.} \quad & \forall n \quad \underline{\mathbf{W}}_n = \underline{\mathbf{D}}^\pi. \end{aligned} \tag{20}$$

In this formulation, although the nuclear norms depend on one another through the introduced constraint, we can decouple the minimization problem into separate subproblems. To solve this problem, we first find a solution to the problem without the constraint $\mathbf{D} \in \mathcal{D}$, then project the solution onto $\mathcal{D}$ by normalizing the columns of $\mathbf{D}$. We adopt this approximation to avoid the complexity of solving the problem with the

[8] In the body of Algorithms 1–3 we drop the iteration index t for simplicity.

constraint $\mathbf{D} \in \mathcal{D}$. Such approach has been used in prior works; see, e.g., [13], [29]. We can solve the objective function (20) (without $\mathbf{D} \in \mathcal{D}$ constraint) using ADMM, which involves decoupling the problem into independent subproblems by forming the following augmented Lagrangian:

$$\begin{aligned}\mathcal{L}_\gamma = \frac{1}{2}\left\|\mathbf{Y}-\mathbf{D}\mathbf{X}\right\|_F^2 + \sum_{n=1}^{N} \Big(\lambda_1 \left\|\mathbf{W}_n^{(n)}\right\|_{\text{tr}} \\ - \langle \underline{\mathbf{A}}_n,\ \underline{\mathbf{D}}^\pi - \underline{\mathbf{W}}_n\rangle + \frac{\gamma}{2}\left\|\ \underline{\mathbf{D}}^\pi - \underline{\mathbf{W}}_n\right\|_F^2\Big),\end{aligned} \tag{21}$$

where $\mathcal{L}_\gamma$ is shorthand for $\mathcal{L}_\gamma(\underline{\mathbf{D}}^\pi, \{\underline{\mathbf{W}}_n\}, \{\underline{\mathbf{A}}_n\})$. In order to find the gradient of (21) with respect to $\underline{\mathbf{D}}^\pi$, we rewrite the Lagrangian function in the following form:

$$\begin{aligned}\mathcal{L}_\gamma = \frac{1}{2}\left\|\mathbf{y}-\mathcal{T}(\underline{\mathbf{D}}^\pi)\right\|_2^2 + \sum_{n=1}^{N} \Big(\lambda_1 \left\|\mathbf{W}_n^{(n)}\right\|_{\text{tr}} \\ - \langle \underline{\mathbf{A}}_n,\ \underline{\mathbf{D}}^\pi - \underline{\mathbf{W}}_n\rangle + \frac{\gamma}{2}\left\|\ \underline{\mathbf{D}}^\pi - \underline{\mathbf{W}}_n\right\|_F^2\Big).\end{aligned}$$

Here, $\mathbf{y} \triangleq \text{vec}(\mathbf{Y})$ (not to be confused with our earlier use of $\underline{\mathbf{y}}$ for $\text{vec}(\underline{\mathbf{Y}})$) and the linear operator $\mathcal{T}(\underline{\mathbf{D}}^\pi) \triangleq \text{vec}(\mathbf{D}\mathbf{X}) = \widetilde{\mathbf{X}}^T\mathbf{\Pi}^T\,\text{vec}(\underline{\mathbf{D}}^\pi)$, where $\widetilde{\mathbf{X}} = \mathbf{X} \otimes \mathbf{I}_m$ and $\mathbf{\Pi}$ is a permutation matrix such that $\text{vec}(\underline{\mathbf{D}}^\pi) = \mathbf{\Pi}\,\text{vec}(\mathbf{D})$. The procedure to find $\mathbf{\Pi}$ is explained in Appendix A.

ADMM Update Rules: Each iteration τ of ADMM consists of the following steps [37]:

$$\underline{\mathbf{D}}^\pi(\tau) = \operatorname*{argmin}_{\underline{\mathbf{D}}^\pi} \mathcal{L}_\gamma(\underline{\mathbf{D}}^\pi, \underline{\mathbf{W}}_n(\tau-1), \underline{\mathbf{A}}_n(\tau-1)), \tag{22}$$

$$\underline{\mathbf{W}}_n(\tau) = \operatorname*{argmin}_{\underline{\mathbf{W}}_n} \mathcal{L}_\gamma(\underline{\mathbf{D}}^\pi(\tau), \underline{\mathbf{W}}_n, \underline{\mathbf{A}}_n(\tau-1)), \tag{23}$$

$$\underline{\mathbf{A}}_n(\tau) = \underline{\mathbf{A}}_n(\tau-1) - \gamma\left(\underline{\mathbf{D}}^\pi(\tau) - \underline{\mathbf{W}}_n(\tau)\right), \tag{24}$$

for all $n \in [N]$. The solution to (22) can be obtained by taking the gradient of $\mathcal{L}_\gamma(\cdot)$ w.r.t. $\underline{\mathbf{D}}^\pi$ and setting it to zero. Suppressing the iteration index τ for ease of notation, we have

$$\frac{\partial \mathcal{L}_\gamma}{\partial \underline{\mathbf{D}}^\pi} = \mathcal{T}^*(\mathcal{T}(\underline{\mathbf{D}}^\pi) - \mathbf{y}) - \sum_{n=1}^{N} \underline{\mathbf{A}}_n + \sum_{n=1}^{N} \gamma\left(\underline{\mathbf{D}}^\pi - \underline{\mathbf{W}}_n\right),$$

where $\mathcal{T}^*(\mathbf{v}) = \text{vec}^{-1}\left(\mathbf{\Pi}\widetilde{\mathbf{X}}\mathbf{v}\right)$ is the *adjoint* of the linear operator $\mathcal{T}$ [39]. Setting the gradient to zero results in

$$\mathcal{T}^*(\mathcal{T}(\underline{\mathbf{D}}^\pi)) + \gamma N\ \underline{\mathbf{D}}^\pi = \mathcal{T}^*(\mathbf{y}) + \sum_{n=1}^{N}\left(\underline{\mathbf{A}}_n + \gamma\underline{\mathbf{W}}_n\right).$$

Equivalently, we have

$$\begin{aligned}\text{vec}^{-1}\left(\left[\mathbf{\Pi}\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^T\mathbf{\Pi}^T + \gamma N\mathbf{I}\right]\text{vec}(\underline{\mathbf{D}}^\pi)\right) \\ = \text{vec}^{-1}(\mathbf{\Pi}\widetilde{\mathbf{X}}\mathbf{y}) + \sum_{n=1}^{N}\left(\underline{\mathbf{A}}_n + \gamma\underline{\mathbf{W}}_n\right).\end{aligned} \tag{25}$$

Therefore, suppressing the index τ, the update rule for $\underline{\mathbf{D}}^\pi$ is

$$\begin{aligned}\underline{\mathbf{D}}^\pi =& \text{vec}^{-1}\left(\left[\mathbf{\Pi}^T\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^T\mathbf{\Pi} + \gamma N\mathbf{I}_{mp}\right]^{-1}\right. \\ &\left.\cdot\left[\mathbf{\Pi}^T\widetilde{\mathbf{X}}\mathbf{y} + \text{vec}\left(\sum_{n=1}^{N}\left(\underline{\mathbf{A}}_n + \gamma\underline{\mathbf{W}}_n\right)\right)\right]\right).\end{aligned} \tag{26}$$

To update $\{\underline{\mathbf{W}}_n\}$, we can further split (23) into N independent subproblems (suppressing the index τ):

$$\begin{aligned}\min_{\underline{\mathbf{W}}_n}\ \mathcal{L}_{\mathcal{W}} =& \lambda_1\left\|\mathbf{W}_n^{(n)}\right\|_{\text{tr}} - \langle\underline{\mathbf{A}}_n,\ \underline{\mathbf{D}}^\pi - \underline{\mathbf{W}}_n\rangle \\ &+ \frac{\gamma}{2}\left\|\ \underline{\mathbf{D}}^\pi - \underline{\mathbf{W}}_n\right\|_F^2.\end{aligned}$$

We can reformulate $\mathcal{L}_{\mathcal{W}}$ as

$$\begin{aligned}\mathcal{L}_{\mathcal{W}} =& \lambda_1\left\|\mathbf{W}_n^{(n)}\right\|_{\text{tr}} + \frac{\gamma}{2}\left\|\mathbf{W}_n^{(n)} - \left([\underline{\mathbf{D}}^\pi]^{(n)} - \frac{\mathbf{A}_n^{(n)}}{\gamma}\right)\right\|_F^2 \\ &+ \text{const.}\end{aligned}$$

The minimizer of $\mathcal{L}_{\mathcal{W}}$ with respect to $\mathbf{W}_n^{(n)}$ is $\text{shrink}\left([\mathbf{D}^\pi]^{(n)} - \frac{1}{\gamma}\mathbf{A}_n^{(n)},\ \frac{\lambda_1}{\gamma}\right)$ where $\text{shrink}(\mathbf{A}, z)$ applies soft thresholding at level z on the singular values of matrix $\mathbf{A}$ [40]. Therefore, suppressing the index τ,

$$\underline{\mathbf{W}}_n = \text{refold}\Big(\text{shrink}\left([\mathbf{D}^\pi]^{(n)} - \frac{1}{\gamma}\mathbf{A}_n^{(n)},\ \frac{\lambda_1}{\gamma}\right)\Big), \tag{27}$$

where $\text{refold}(\cdot)$ is the inverse of the unfolding operator. Algorithm 1 summarizes this discussion and provides pseudocode for the dictionary update stage in STARK.

B. TeFDiL: A Factorization-based LSR-DL Algorithm

While our experiments in Section VI validate good performance of STARK, the algorithm finds the dictionary $\mathbf{D} \in \mathbb{R}^{m\times p}$ and not the subdictionaries $\{\mathbf{D}_n \in \mathbb{R}^{m_n\times p_n}\}_{n=1}^N$. Moreover, STARK only allows indirect control over the separation rank of the dictionary through the regularization parameter λ_1. This motivates developing a factorization-based LSR-DL algorithm that can find the subdictionaries and allows for direct tuning of the separation rank to control the number of parameters of the model. To this end, we propose a factorization-based LSR-DL algorithm termed *Tensor Factorization-Based DL (TeFDiL)* in this section for solving Problem (9).

We discussed earlier in Section V-A that the error term $\|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2$ can be reformulated as $\|\mathbf{y} - \mathcal{T}(\underline{\mathbf{D}}^\pi)\|^2$ where $\mathcal{T}(\underline{\mathbf{D}}^\pi) = \widetilde{\mathbf{X}}^T\mathbf{\Pi}^T\,\text{vec}(\underline{\mathbf{D}}^\pi)$. Thus, the dictionary update objective in (9) can be reformulated as $\|\mathbf{y} - \mathcal{T}(\sum_{k=1}^r \mathbf{d}_N^k \circ \cdots \circ \mathbf{d}_1^k)\|^2$ where $\mathbf{d}_n^k \triangleq \text{vec}(\mathbf{D}_n^k)$. To avoid the complexity of solving this problem, we resort to first obtaining an inexact solution by minimizing $\|\mathbf{y} - \mathcal{T}(\underline{\mathbf{A}})\|^2$ over $\underline{\mathbf{A}}$ and then enforcing the low-rank structure by finding the rank-r approximation of the minimizer of $\|\mathbf{y} - \mathcal{T}(\underline{\mathbf{A}})\|^2$. TeFDiL employs CP decomposition (CPD) to find this approximation and thus enforce LSR structure on the updated dictionary.

Assuming the matrix of sparse codes $\mathbf{X}$ is full row-rank[9], then $\widetilde{\mathbf{X}}^T$ is full column-rank and $\underline{\mathbf{A}} = \mathcal{T}^+(\mathbf{y}) = \text{vec}^{-1}\left(\mathbf{\Pi}\big(\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^T\big)^{-1}\widetilde{\mathbf{X}}\mathbf{y}\right)$ minimizes $\|\mathbf{y} - \mathcal{T}(\underline{\mathbf{A}})\|^2$. Now, it remains to solve the following problem to update $\{\mathbf{d}_n^k\}$:

$$\min_{\{\mathbf{d}_n^k\}}\ \left\|\sum_{k=1}^{r}\mathbf{d}_N^k \circ \cdots \circ \mathbf{d}_1^k - \mathcal{T}^+(\mathbf{y})\right\|_F^2.$$

Although finding the best rank-r approximation (r-term CPD) of a tensor is ill-defined in general [41], various numerical al-

[9] In our experiments, we add $\delta\mathbf{I}$ to $\mathbf{X}\mathbf{X}^T$ with a small $\delta > 0$ at every iteration to ensure full-rankness.

gorithms exist in the tensor recovery literature to find a "good" rank-r approximation of a tensor [21], [41]. TeFDiL can employ any of these algorithms to find the r-term CPD, denoted by $\text{CPD}_r(\cdot)$, of $\mathcal{T}^+(\mathbf{y})$. At the end of each dictionary update stage, the columns of $\mathbf{D} = \sum \bigotimes \mathbf{D}_n^k$ are normalized. Algorithm 2 describes the dictionary update step of TeFDiL.

Algorithm 2 Dictionary Update in TeFDiL for LSR-DL

```
Require: $\mathbf{Y}$, $\mathbf{X}(t)$, $\mathbf{\Pi}$, $r$
Construct $\mathcal{T}^+(\mathbf{y}) = \text{vec}^{-1}\left(\mathbf{\Pi}\big(\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^T\big)^{-1}\widetilde{\mathbf{X}}\mathbf{y}\right)$
$\underline{\mathbf{D}}^\pi \leftarrow \text{CPD}_r(\mathcal{T}^+(\mathbf{y}))$
$\mathbf{D} \leftarrow \text{vec}^{-1}\left(\mathbf{\Pi}^T\,\text{vec}(\underline{\mathbf{D}}^\pi)\right)$
Normalize columns of $\mathbf{D}$
return $\mathbf{D}(t+1)$
```

C. OSubDil: An Online LSR-DL Algorithm

Both STARK and TeFDiL are batch methods in that they use the entire dataset for DL in every iteration. This makes them less scalable with the size of datasets due to high memory and per iteration computational cost and also makes them unsuitable for streaming data settings. To overcome these limitations, we now propose an online LSR-DL algorithm termed *Online SubDictionary Learning for structured DL (OSubDil)* that uses only a single data sample (or a small mini-batch) in every iteration (see Algorithm 3). This algorithm has better memory efficiency as it removes the need for storing all data points and has significantly lower per-iteration computational complexity. In *OSubDil*, once a new sample $\underline{\mathbf{Y}}(t+1)$ arrives, its sparse representation $\underline{\mathbf{X}}(t+1)$ is found using the current dictionary estimate $\mathbf{D}(t)$ and then the dictionary is updated using $\underline{\mathbf{Y}}(t+1)$ and $\underline{\mathbf{X}}(t+1)$. The dictionary update stage objective function after receiving the T-th sample is

$$J_T(\{\mathbf{D}_n^k\}) = \frac{1}{T}\sum\nolimits_{t=1}^{T} \|\mathbf{y}(t) - \big(\sum\nolimits_{k=1}^{r}\bigotimes\nolimits_{n=1}^{N}\mathbf{D}_n^k\big)\mathbf{x}(t)\|^2.$$

We can rewrite this objective as

$$\begin{aligned} J_T &= \sum\nolimits_{t=1}^{T} \|\mathbf{Y}^{(n)}(t) - \sum\nolimits_{k=1}^{r}\mathbf{D}_n^k\mathbf{X}^{(n)}(t)\mathbf{C}_n^k(t)\|_F^2 \\ &= \sum\nolimits_{t=1}^{T} \|\widehat{\mathbf{Y}}^{(n)}(t) - \mathbf{D}_n^k\mathbf{X}^{(n)}(t)\mathbf{C}_n^k(t)\|_F^2 \\ &= \text{Tr}\left([\mathbf{D}_n^k]^T\mathbf{D}_n^k\mathbf{A}_n^k(t)\right) - 2\,\text{Tr}\left([\mathbf{D}_n^k]^T\mathbf{B}_n^k(t)\right) + \text{const.}, \end{aligned}$$

where, dropping the iteration index t, the matrix $\mathbf{C}_n^k \triangleq \left(\mathbf{D}_N^k \otimes \cdots \otimes \mathbf{D}_{n+1}^k \otimes \mathbf{D}_{n-1}^k \cdots \otimes \mathbf{D}_1^k\right)^T$ and the estimate $\widehat{\mathbf{Y}}^{(n)} \triangleq \mathbf{Y}^{(n)} - \sum_{\substack{i=1\\i\neq k}}^{r}\mathbf{D}_n^i\mathbf{X}^{(n)}\mathbf{C}_n^i$. We can further define the matrices $\mathbf{A}_n^k(t) \triangleq \sum_{\tau=1}^{t}\mathbf{X}^{(n)}(t)\mathbf{C}_n^k(\tau)[\mathbf{C}_n^k(\tau)]^T[\mathbf{X}^{(n)}(\tau)]^T$ and $\mathbf{B}_n^k(t) \triangleq \sum_{\tau=1}^{t}\widehat{\mathbf{Y}}^{(n)}(\tau)[\mathbf{C}_n^k(\tau)]^T[\mathbf{X}^{(n)}(\tau)]^T$. To minimize J_T with respect to each $\mathbf{D}_n^k$, we take a similar approach as in Mairal et al. [6] and use a (block) coordinate descent algorithm with warm start to update the columns of $\mathbf{D}_n^k$ in a cyclic manner. Algorithm 3 describes the dictionary update step of OSubDil.

Algorithm 3 Dictionary Update in OSubDil for LSR-DL

```
Require: $\underline{\mathbf{Y}}(t)$, $\{\mathbf{D}_n^k(t)\}$, $\mathbf{A}_n^k(t)$, $\mathbf{B}_n^k(t)$, $\underline{\mathbf{X}}(t)$
for all $k \in [r]$ do
    for all $n \in [N]$ do
        $\mathbf{C}_n^k \leftarrow \left(\mathbf{D}_N^k \otimes \cdots \otimes \mathbf{D}_{n+1}^k \otimes \mathbf{D}_{n-1}^k \cdots \otimes \mathbf{D}_1^k\right)^T$
        $\widehat{\mathbf{Y}}^{(n)} \leftarrow \mathbf{Y}^{(n)} - \sum_{\substack{i=1\\i\neq k}}^{r}\mathbf{D}_n^i\mathbf{X}^{(n)}\mathbf{C}_n^i$
        $\mathbf{A}_n^k \leftarrow \mathbf{A}_n^k + \mathbf{X}^{(n)}\mathbf{C}_n^k[\mathbf{C}_n^k]^T[\mathbf{X}^{(n)}]^T$
        $\mathbf{B}_n^k \leftarrow \mathbf{B}_n^k + \widehat{\mathbf{Y}}^{(n)}[\mathbf{C}_n^k]^T[\mathbf{X}^{(n)}]^T$
        for $j = 1, \cdots, p_n$ do
            $[\mathbf{D}_n^k]_j \leftarrow \frac{1}{[\mathbf{A}_n^k]_{jj}}([\mathbf{B}_n^k]_j - \mathbf{D}_n^k[\mathbf{A}_n^k]_j) + [\mathbf{D}_n^k]_j$
        end for
    end for
end for
Normalize columns of $\mathbf{D} = \sum_{n=1}^{r}\bigotimes_{n=1}^{N}\mathbf{D}_n^k$
return $\{\mathbf{D}_n^k(t+1)\}$
```

VI. Numerical Experiments

We evaluate our algorithms on synthetic and real-world datasets to understand the impact of training set size and noise level on the performance of LSR-DL. In particular, we want to understand the effect of exploiting additional structure in representation accuracy and denoising performance. We compare the performance of our proposed algorithms with existing DL algorithms in each scenario and show that in almost every case our proposed LSR-DL algorithms outperform K-SVD [5]. Our results also offer insights into how the size and quality of training data can affect the choice of the proper DL model. Specifically, our experiments on image denoising show that when the noise level in data is high, TeFDiL performs best when the separation rank is 1 but in low noise regimes, its performance improves as we increase the separation rank. Furthermore, our synthetic experiments confirm that when the true underlying dictionary follows the KS (LSR) structure, our structured algorithms clearly outperform K-SVD, especially when the number of training samples is very small. This implies that our algorithms should perform well in applications where the true dictionary is close to being LSR-structured.

Remark. In all our experiments, hyperparameters λ_1 and γ (for STARK), r (for SubDil), and regularization parameter for sparsity, λ, have been selected using cross-validation on each training dataset based on representation error. The only exception is for r in cases where we specify its value in the KS-DL experiments ($r = 1$) in Table II and in the TeFDiL experiments reported in Table III.

Synthetic Experiments: We compare our algorithms to K-SVD (standard DL) as well as a simple block coordinate descent (BCD) algorithm that alternates between updating every subdictionary in problem (9). This BCD algorithm can be interpreted as an extension of the KS-DL algorithm [29] for the LSR model. We show how structured DL algorithms outperform the unstructured algorithm K-SVD [5] when the underlying dictionary is structured, especially when the training set is small. We focus on 3rd-order tensor data and we randomly generate a KS dictionary $\mathbf{D} = \mathbf{D}_1 \otimes \mathbf{D}_2 \otimes \mathbf{D}_3$ with dimensions $\mathbf{m} = [2, 5, 3]$ and $\mathbf{p} = [4, 10, 5]$. We select i.i.d samples from the standard Gaussian distribution, $\mathcal{N}(0, 1)$, for the subdictionary elements, and then normalize the columns of the subdictionaries. To generate $\mathbf{x}$, we select the locations of $s = 5$ nonzero elements uniformly at random. The values of those elements are sampled i.i.d. from $\mathcal{N}(0, 1)$. We assume

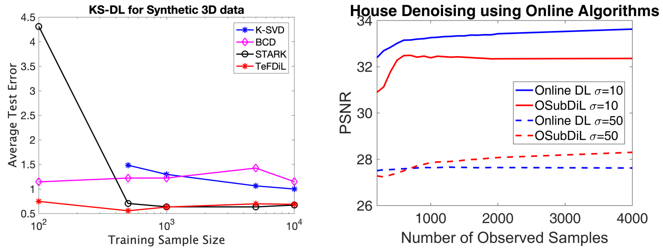
TABLE II: Performance of DL algorithms for image denoising in terms of PSNR

		Unstructured	KS-DL ($r=1$)			LSR-DL ($r>1$)		
Image	Noise	K-SVD [5]	SeDiL [11]	BCD [29]	TeFDiL	BCD	STARK	TeFDiL
House	$\sigma = 10$	35.6697	23.1895	31.6089	36.2955	32.2952	33.4002	37.1275
	$\sigma = 50$	25.4684	23.6916	24.8303	27.5412	21.6128	27.3945	26.5905
Castle	$\sigma = 10$	33.0910	23.6955	32.7592	34.5031	30.3561	37.0428	35.1000
	$\sigma = 50$	22.4184	23.2658	22.3065	24.6670	20.4414	24.4965	23.3372
Mushroom	$\sigma = 10$	34.4957	25.8137	33.2797	36.5382	32.2098	36.9443	37.7016
	$\sigma = 50$	22.5495	22.9464	22.8554	22.9284	21.7792	25.1081	22.8374
Lena	$\sigma = 10$	33.2690	23.6605	30.9575	34.8854	31.1309	33.8813	35.3009
	$\sigma = 50$	22.5070	23.4207	21.6985	23.4988	19.5989	24.8211	23.1658

TABLE III: Performance of TeFDiL with various ranks for image denoising in terms of PSNR

Image	Noise	$r=1$	$r=4$	$r=8$	$r=16$	$r=32$	K-SVD
Mushroom	$\sigma = 10$	36.5382	36.7538	37.4173	37.4906	37.7016	34.4957
	$\sigma = 50$	22.9284	22.8352	22.8384	22.8419	22.8374	22.5495
Number of parameters		265	1060	2120	4240	8480	147456

observations are generated according to $\mathbf{y} = \mathbf{D}\mathbf{x}$. In the initialization stage of the algorithms, $\mathbf{D}$ is initialized using random columns of $\mathbf{Y}$ for K-SVD and random columns of the unfoldings of $\mathbf{Y}$ for the structured DL algorithms. Sparse coding is performed using OMP [42]. Due to the invariance of DL to column permutations in the dictionary, we choose reconstruction error as the performance criteria. For $L = 100$, K-SVD cannot be used since $p > L$. Reconstruction errors are plotted in Figure 3a. It can be seen that for small number of samples, TeFDiL outperforms all three algorithms BCD, K-SVD, and STARK. As more samples become available, both TeFDiL and STARK (the proposed algorithms) outperform BCD and K-SVD.



(a) (b)

Fig. 3: (a) Normalized representation error of various DL algorithms for 3rd-order synthetic tensor data. (b) Performance of online DL algorithms for House.

Real-world Experiments: In this set of experiments, we evaluate the image denoising performance of different DL algorithms on four RGB images, House, Castle, Mushroom, and Lena, which have dimensions $256 \times 256 \times 3$, $480 \times 320 \times 3$, $480 \times 320 \times 3$, and $512 \times 512 \times 3$, respectively. We corrupt the images using additive white Gaussian noise with standard deviations $\sigma = \{10, 50\}$. To construct the training data set, we extract overlapping patches of size 8×8 from each image and treat each patch as a 3-dimensional data sample. We learn dictionaries with parameters $\mathbf{m} = [3, 8, 8]$ and $\mathbf{p} = [3, 16, 16]$. In the training stage, we perform sparse coding using FISTA [43] (to reduce training time) with regularization parameter $\lambda = 0.1$ for all algorithms. To perform denoising, we use OMP with $s = \lceil p/20 \rceil$. To evaluate the denoising performances of the methods, we use the resulting peak signal to noise ratio (PSNR) of the reconstructed images [44]. Table II demonstrates the image denoising results.

LSR-DL vs Unstructured DL: We observe that STARK outperforms K-SVD in every case when the noise level is high and in most cases when the noise level is low. Moreover, TeFDiL outperforms K-SVD in both low-noise and high-noise regimes for all four images while having considerably fewer parameters (one to three orders of magnitude).[10]

LSR-DL vs KS-DL: Our LSR-DL methods outperform SeDiL [11] and while BCD [29] has a good performance for $\sigma = 10$, its denoising performance suffers when noise level increases.[11]

Table III demonstrates the image denoising performance of TeFDiL for Mushroom based on the separation rank of TeFDiL. When the noise level is low, performance improves with increasing the separation rank. However, for higher noise level $\sigma = 50$, increasing the number of parameters has an inverse effect on the generalization performance.

Comparison of LSR-DL Algorithms: We compare LSR-DL algorithms BCD, STARK and TeFDiL. As for the merits of our LSR-DL algorithms over BCD, our experiments show that both TeFDiL and STARK outperform BCD in both noise regimes. In addition, while TeFDiL and STARK can be easily and efficiently used for higher separation rank dictionaries, when the separation rank is higher, BCD with higher rank does not perform well. While STARK has a better performance than TeFDiL for some tasks, it has the disadvantage that it does not output the subdictionaries and does not allow for direct tuning of the separation rank. Ultimately, the choice between these two algorithms will be application dependent. The flexibility in tuning the number of KS terms in the

[10] While the improvements in image denoising reported in DL papers are sometimes below 0.5 dB [5], [11], [29], [45]), we show that our algorithms provide 1–3 dB improvements over K-SVD in most scenarios.

[11] Note that SeDiL results may be improved by careful parameter tuning.

dictionary in TeFDiL (and indirectly in STARK, through parameter λ_1) allows selection of the number of parameters in accordance with the size and quality of the training data. When the training set is small and noisy, smaller separation rank (perhaps 1) results in a better performance. For training sets of larger size and better quality, increasing the separation rank allows for higher capacity to learn more complicated structures, resulting in a better performance.

OSubDil vs Online (Unstructured) DL: Figure 3b shows the PSNR for reconstructing House using OSubDil and Online DL in Mairal et al. [6] based on the number of observed samples. We observe that in the presence of high level of noise, our structured algorithm is able to outperform its unstructured counterpart with considerably fewer parameters.

VII. Conclusion

We studied the low separation rank model (LSR-DL) to learn structured dictionaries for tensor data. This model bridges the gap between unstructured and separable dictionary learning (DL) models. For the intractable rank-constrained and the tractable factorization-based LSR-DL formulations, we show that given $\Omega\big(r(\sum_n m_n p_n)p^2\rho^{-2}\big)$ data samples, the true dictionary can be locally recovered up to distance ρ. This is a reduction compared to the $\Omega(mp^3\rho^{-2})$ sample complexity of standard DL in Gribonval et al. [27]. However, a minimax lower bound scaling of $\Omega(p\sum_n m_n p_n \rho^{-2})$ in Shakeri et al. [15] for KS-DL ($r = 1$) has an $O(p)$ gap with our sample complexity upper bound. This gap suggests that the sample complexity bounds may be improved. Possible future directions in this regard include finding minimax bounds for the LSR-DL model and tightening the gap between the sample complexity lower bound (minimax bound) and the upper bounds for this model.

We also show in the regularization-based formulation that $\Omega(mp^3\rho^{-2})$ samples are sufficient for local identifiability of the true Kronecker-structured (KS) dictionary up to distance ρ. Improving this result and providing sample complexity results for when the true dictionary is LSR (and not just KS) is also another interesting future work.

Finally, we presented two LSR-DL algorithms and showed that they have better generalization performance for image denoising in comparison to unstructured DL algorithm K-SVD [5] and existing KS-DL algorithms SeDiL [11] and BCD [29]. We also present OSubDil that to the best our knowledge is the first online algorithm that results in LSR or KS dictionaries. We show that OSubDil results in a faster reduction in the reconstruction error in terms of number of observed samples compared to the state-of-the-art online DL algorithm [6] when the noise level in data is high.

Appendix A
The Rearrangement Procedure

To illustrate the procedure that rearranges a KS matrix into a rank-1 tensor, let us first consider $\mathbf{A} = \mathbf{A}_1 \otimes \mathbf{A}_2$. The elements of $\mathbf{A}$ can be rearranged to form $\mathbf{A}^\pi = \mathbf{d}_2 \circ \mathbf{d}_1$, where $\mathbf{d}_i = \operatorname{vec}(\mathbf{A}_i)$ for $i = 1, 2$ [10]. Figure 4 depicts this rearrangement for $\mathbf{A}$. Similarly, for $\mathbf{A} = \mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \mathbf{A}_3$, we can write $\underline{\mathbf{D}}^\pi = \mathbf{d}_3 \circ \mathbf{d}_2 \circ \mathbf{d}_1$, where each frontal slice[12] of the tensor $\underline{\mathbf{D}}^\pi$ is a scaled copy of $\mathbf{d}_3 \circ \mathbf{d}_2$. The rearrangement of $\mathbf{A}$ into $\underline{\mathbf{A}}^\pi$ is performed via a permutation matrix $\mathbf{\Pi}$ such that $\operatorname{vec}(\underline{\mathbf{A}}^\pi) = \mathbf{\Pi} \operatorname{vec}(\mathbf{A})$. Given index l of $\operatorname{vec}(\mathbf{A})$ and the corresponding mapped index l' of $\operatorname{vec}(\underline{\mathbf{A}}^\pi)$, our strategy for finding the permutation matrix is to define l' as a function of l. To this end, we first find the corresponding row and column indices (i, j) of matrix $\mathbf{A}$ from the lth element of $\operatorname{vec}(\mathbf{A})$. Then, we find the index of the element of interest on the Nth order rearranged tensor $\underline{\mathbf{A}}^\pi$, and finally, we find its location l' on $\operatorname{vec}(\underline{\mathbf{A}}^\pi)$. Note that the permutation matrix needs to be computed only once in an offline manner, as it is only a function of the dimensions of the factor matrices and not the values of elements of $\mathbf{A}$.

We now describe the rearrangement procedure in detail, starting with the more accessible case of KS matrices that are Kronecker product of $N = 3$ factor matrices and then extending it to the general case. Throughout this section, we define an n-th order "tile" to be a scaled copy of $\mathbf{A}_{N-n+1} \otimes \cdots \otimes \mathbf{A}_N$ for $N > 0$. A zeroth-order tile is just an element of a matrix. Moreover, we generalize the concept of slices of a 3rd-order tensor to "hyper-slices": an n-th order hyper-slice is a scaled copy of $\mathbf{d}_N \circ \mathbf{d}_{N-1} \circ \cdots \circ \mathbf{d}_{N-n+1}$.

A. Kronecker Product of 3 *Matrices*

In the case of 3rd-order tensors, we take the following steps:

i) Find index (i, j) in $\mathbf{A}$ that corresponds to the l-th element of $\operatorname{vec}(\mathbf{A})$.
ii) Find the corresponding index (r, c, s) on the third order tensor $\underline{\mathbf{A}}^\pi$.
iii) Find the corresponding index l' on $\operatorname{vec}(\underline{\mathbf{A}}^\pi)$.
iv) Set $\mathbf{\Pi}(l', l) = 1$.

Let $\mathbf{A} = \mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \mathbf{A}_3$, with $\mathbf{A} \in \mathbb{R}^{m \times p}$ and $\mathbf{A}_i \in \mathbb{R}^{m_i \times p_i}$ for $i \in \{1, 2, 3\}$. For the first operation, we have

$$(i, j) = \left(\left\lceil \frac{l}{m} \right\rceil,\ l - \left\lfloor \frac{l-1}{m} \right\rfloor m \right). \tag{28}$$

We can see from Figure 2 that the rearrangement procedure works in the following way. For each element indexed by (i, j) on matrix $\mathbf{A}$, find the 2nd-order tile to which it belongs. Let us index this 2nd-order tile by T_2. Then, find the 1st-order tile (within the 2nd-order tile indexed T_2) on which it lies and index this tile by T_1. Finally, index the location of the element (zeroth-order tile) within this first-order tile by T_0. After rearrangement, the location of this element on the rank-1 tensor is (T_0, T_1, T_2).

In order to find (T_0, T_1, T_2) that corresponds to (i, j), we first find T_2, then T_1, and then T_0. To find T_2, we need to find the index of the 2nd-order tile on which the element indexed by (i, j) lies:

$$T_2 = \underbrace{\left\lfloor \frac{j-1}{p_2 p_3} \right\rfloor}_{S_j^2} m_1 + \underbrace{\left\lfloor \frac{i-1}{m_2 m_3} \right\rfloor}_{S_i^2} + 1, \tag{29}$$

[12] A slice of a 3-dimensional tensor is a 2-dimensional section defined by fixing all but two of its indices. For example, a frontal slice is defined by fixing the third index.

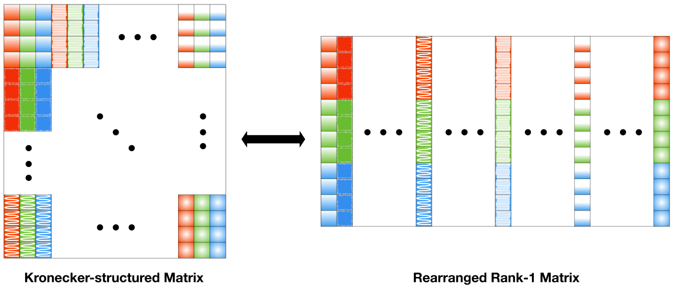


Fig. 4: Rearranging a Kronecker structured matrix ($N = 2$) into a rank-1 matrix.

where S_j^2 and S_i^2 are the number of the 2nd-order tiles on the left and above the tile to which the element belongs, respectively. Now, we find the position of the element in this 2nd-order tile:

$$i_2 = i - S_i^2 m_2 m_3 = i - \left\lfloor \frac{i-1}{m_2 m_3} \right\rfloor m_2 m_3,$$

$$j_2 = j - S_j^2 p_2 p_3 = j - \left\lfloor \frac{j-1}{p_2 p_3} \right\rfloor p_2 p_3. \qquad (30)$$

For the column index, T_1, we have

$$T_1 = \underbrace{\left\lfloor \frac{j_2 - 1}{p_3} \right\rfloor}_{S_j^1} m_2 + \underbrace{\left\lfloor \frac{i_2 - 1}{m_3} \right\rfloor}_{S_i^1} + 1. \qquad (31)$$

The location of the element on the 1st-order tile is

$$i_1 = i_2 - S_i^1 m_3 = i_2 - \left\lfloor \frac{i_2 - 1}{m_3} \right\rfloor m_3,$$

$$j_1 = j_2 - S_j^1 p_3 = j_2 - \left\lfloor \frac{j_2 - 1}{p_3} \right\rfloor p_3. \qquad (32)$$

Therefore, T_0 can be expressed as

$$T_0 = (j_1 - 1)\, m_3 + i_1. \qquad (33)$$

Finally, in the last step we find the corresponding index on vec($\underline{\mathbf{A}}^\pi$) using the following rule.

$$l' = (T_2 - 1) m_2 m_3 p_2 p_3 + (T_1 - 1) m_3 p_3 + T_0. \qquad (34)$$

This process is illustrated in Figure 2.

B. The General Case

We now extend our results to N-th order tensors. Vectorization and its adjoint operation are easy to compute for tensors of any order. We focus on rearranging elements of $\mathbf{A} = \mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \cdots \otimes \mathbf{A}_N$ to form the N-way rank-1 tensor $\underline{\mathbf{A}}^\pi$, where $\mathbf{A}_n \in \mathbb{R}^{m_n \times p_n}$ for $n \in [N]$, $\mathbf{A} \in \mathbb{R}^{m \times p}$, and $\underline{\mathbf{A}}^\pi \in \mathbb{R}^{m_N p_N \times m_{N-1} p_{N-1} \times \cdots \times m_1 p_1}$.

We first formally state the rearrangement and then we explain it. Similar to the case of $N = 3$ explained earlier, for each element of the KS matrix $\mathbf{A}$ indexed by (i, j), we first find the $(N-1)$th-order tile to which it belongs, then the $(N-2)$th-order tile, and so on. Let $T_{N-1}, T_{N-2}, \cdots, T_0$ denote the indices of these tiles, respectively. Then, after rearrangement, the element indexed (i, j) on KS matrix $\mathbf{A}$ becomes the element indexed $T_0, \cdots, T_{N-1}$ on the rearrangement tensor $\underline{\mathbf{A}}^\pi$.

Now, let us find the indices of the tiles of KS matrix $\mathbf{A}$ to which the element (i, j) belongs. In the following, we denote by (i_n, j_n) the index of this element within its nth-order tile. Note that since $\mathbf{A}$ is an Nth-order tile itself, we can use (i_N, j_N) instead of (i, j) to refer to the index of the element on $\mathbf{A}$ for consistency of notation. For the (i_N, j_N)-th element of $\mathbf{A}$ we have

$$T_{N-1} = \underbrace{\left\lfloor \frac{j_N - 1}{\Pi_{t=2}^N\, p_t} \right\rfloor}_{S_j^N} m_1 + \underbrace{\left\lfloor \frac{i_N - 1}{\Pi_{t=2}^N\, m_t} \right\rfloor}_{S_i^N} + 1,$$

$$i_{N-1} = i_N - S_i^N\ \Pi_{t=2}^N\, m_t,$$

$$j_{N-1} = j_N - S_j^N\ \Pi_{t=2}^N\, p_t,$$

where T_{N-1} is the index of the $(N-1)$-th order tile and (i_{N-1}, j_{N-1}) is the location of the given element within this tile. Similarly, we have

$$T_{N-n} = \underbrace{\left\lfloor \frac{j_{N-n+1} - 1}{\Pi_{t=n+1}^N\, p_t} \right\rfloor}_{S_j^{N-n+1}} m_n + \underbrace{\left\lfloor \frac{i_{N-n+1} - 1}{\Pi_{t=n+1}^N\, m_t} \right\rfloor}_{S_i^{N-n+1}} + 1,$$

$$i_{N-n} = i_{N-n+1} - S_i^n\ \Pi_{t=n+1}^N\, m_t,$$

$$j_{N-n} = j_{N-n+1} - S_j^n\ \Pi_{t=n+1}^N\, p_t,$$

for $N > n > 1$. Finally, we have

$$T_0 = (j_1 - 1) m_N + i_1.$$

It is now easy to see that the (i_N, j_N)-th element of $\mathbf{A}$ is the $(T_0,\ T_1,\ \cdots,\ T_{N-1})$-th element of $\underline{\mathbf{A}}^\pi$.

Intuitively, notice that N-th order KS matrix $\mathbf{A}$ is a tiling of $m_1 \times p_1$ KS tiles of order $N-1$. In rearranging $\mathbf{A}$ into $\underline{\mathbf{A}}^\pi$, the elements of each of these $(N-1)$-th order tiles construct a $(N-1)$-th order "hyper-slice". On matrix $\mathbf{A}$, these tiles consist of $m_2 \times p_2$ tiles, each of which is a $(N-2)$th-order KS matrix, whose elements are rearranged to a $(N-2)$-th hyper-slice of $\underline{\mathbf{A}}^\pi$, and so on. Hence, the idea is to use the correspondence between the nth-order tiles and nth-order hyper-slices: finding the index of the n-th order tile of $\mathbf{A}$ on which (i, j) lies is equivalent to finding the index of the nth-order hyper-slice of $\underline{\mathbf{A}}^\pi$ to which it is translated. Note that each entry of a tensor in indexed by an N-tuple and the index of an entry of a tensor on its nth hyper-slice is in fact its nth element in the index tuple of this entry. Therefore, we first find the $(N-1)$-th order KS tile of $\mathbf{A}$ on which the (i, j) element lies (equivalent to finding the $(N-1)$th-order hyper-slice to which (i, j) is translated), and then find the location (i_{N-1}, j_{N-1}) of this element on this tile. Next, the $(N-2)$-th order KS tile in which (i_{N-1}, j_{N-1}) lies is found as well as the location (i_{N-2}, j_{N-2}) of the element within this tile, and so on.

Appendix B
Proofs of Lemmas

Proof of Lemma 2. Proposition 4.1 in De Silva and Lim [41] shows that the space of tensors of order $N \geq 3$ and rank $r \geq 2$ is not closed. The fact that the rearrangement process preserves topological properties of sets means that the same result holds for the set $\mathcal{L}^{N,r}_{\mathbf{m},\mathbf{p}}$ with $N \geq 3$ and rank $r \geq 2$.

The proof for closedness of $\mathcal{L}^{N,1}_{\mathbf{m},\mathbf{p}}$ and $\mathcal{L}^{2,r}_{\mathbf{m},\mathbf{p}}$ follows from Propositions 4.2 and 4.3 in De Silva and Lim [41], which can be adopted here due to the relation between the sets of low-rank tensors and LSR matrices. □

Proof of Lemma 3. The rearrangement process allows us to borrow the results in Proposition 4.8 in De Silva and Lim [41] for tensors and apply them to LSR matrices. □

Proof of Lemma 6. Define $\mathcal{M}^r_{m\times p} = \{\mathbf{D} \in \mathcal{U} | \operatorname{rank}(\mathbf{D}) \leq r\}$ and $\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}} = \mathcal{L}^{2,r}_{\mathbf{m},\mathbf{p}} \cap \mathcal{U}$. Since the rearrangement operator is an isometry w.r.t. the Euclidean distance, the image of an ϵ-net of $\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}$ w.r.t. the Frobenius norm under this rearrangement operator is an ϵ-net of $\mathcal{M}^r_{m'\times p'}$ ($m' = m_2p_2$ and $p' = m_1p_1$) w.r.t the Frobenius norm. Thus, $\mathcal{N}_F(\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon) = \mathcal{N}_F(\mathcal{M}^r_{m'\times p'}, \epsilon)$. We also know that $\mathcal{N}_F(\mathcal{M}^r_{m'\times p'}, \epsilon) \leq (9/\epsilon)^{r(m'+p'+1)}$ [46]. This means that

$$\mathcal{N}_F(\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon) \leq (9/\epsilon)^{r(m_1p_1+m_2p_2+1)}. \tag{35}$$

On the other hand, for the oblique manifold we have $\mathcal{D}_{m\times p} \subset p\mathcal{U}$ and therefore, $\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}} \subset p\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}$. Hence, $\mathcal{N}_{2,\infty}(\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}) \leq \mathcal{N}_{2,\infty}(p\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon)$. Also, since $\|\mathbf{M}\|_{2,\infty} \leq \|\mathbf{M}\|_F$ for any $\mathbf{M}$, it follows that an ϵ-covering of any given set w.r.t. the Frobenius norm is also an ϵ-covering of that set w.r.t. the max-column-norm. Thus $\mathcal{N}_{2,\infty}(\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}) \leq \mathcal{N}_{2,\infty}(p\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon) \leq \mathcal{N}_F(p\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon)$. Moreover, it follows from the fact $\mathcal{N}_F(p\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon) = \mathcal{N}_F(\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon/p)$ that

$$\mathcal{N}_{2,\infty}(\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon) \leq \mathcal{N}_F(\widehat{\mathcal{L}}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon/p). \tag{36}$$

Thus, from (35) and (36) we see that $\mathcal{N}_{2,\infty}(\mathcal{K}^{2,r}_{\mathbf{m},\mathbf{p}}, \epsilon) \leq (9p/\epsilon)^{r(m_1p_1+m_2p_2+1)}$, which concludes the proof. □

Proof of Lemma 7. Each element $\mathbf{D} \in {}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$ can be written as a summation of at most r KS matrices $\bigotimes \mathbf{D}^k_n$ such that $\left\|\bigotimes \mathbf{D}^k_n\right\|_F \leq c$. This implies that ${}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}$ is a subset of the Minkowski sum (vector sum) of r copies of ${}^c\mathcal{K}^{N,1}_{\mathbf{m},\mathbf{p}}$, the set of KS matrices within the Euclidean ball of radius c. It is easy to show that the Minkowski sum of the ϵ-coverings of r sets is an $r\epsilon$-covering of the Minkowski sum of those sets in any norm. Therefore, we have

$$\mathcal{N}_{2,\infty}({}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}, \epsilon) \leq \left(\mathcal{N}_{2,\infty}({}^c\mathcal{K}^{N,1}_{\mathbf{m},\mathbf{p}}, \epsilon/r)\right)^r. \tag{37}$$

Moreover, we have ${}^c\mathcal{K}^{N,1}_{\mathbf{m},\mathbf{p}} \subset c\mathcal{K}^{N}_{\mathbf{m},\mathbf{p}}$. We also know from equation (16) that $\mathcal{N}(\mathcal{K}^N_{\mathbf{m},\mathbf{p}}, \epsilon) \leq (3/\epsilon)^{\sum_{i=1}^N m_ip_i}$. Putting all these facts together, we get

$$\begin{aligned}\mathcal{N}_{2,\infty}({}^c\mathcal{K}^{N,r}_{\mathbf{m},\mathbf{p}}, \epsilon) &\leq \left(\mathcal{N}_{2,\infty}(c\mathcal{K}^{N}_{\mathbf{m},\mathbf{p}}, \epsilon/r)\right)^r \\ &\leq (3rc/\epsilon)^{r\sum_{i=1}^N m_ip_i}.\end{aligned} \tag{38}$$

□

Proof of Lemma 8. According to Lemma 2 in Shakeri et al. [16], for any $\{\mathbf{A}_n\}$ and $\{\mathbf{B}_n\}$ we have

$$\begin{aligned}&\bigotimes_{n=1}^N \mathbf{A}_n - \bigotimes_{n=1}^N \mathbf{B}_n \\ &\quad= \sum_{n=1}^N \boldsymbol{\Gamma}_1 \otimes \cdots \otimes (\mathbf{A}_n - \mathbf{B}_n) \otimes \cdots \otimes \boldsymbol{\Gamma}_N,\end{aligned} \tag{39}$$

where $\boldsymbol{\Gamma}_n = \mathbf{A}_n$ or $\boldsymbol{\Gamma}_n = \mathbf{B}_n$ depending on n. Let $\epsilon^k_n \triangleq \|\mathbf{A}^k_n - \mathbf{B}^k_n\|_F$. Using equality (39), we have

$$\begin{aligned}&\Big\| \sum_{k=1}^r \bigotimes \mathbf{A}^k_n - \sum_{k=1}^r \bigotimes \mathbf{B}^k_n \Big\|_F \\ &= \Big\| \sum_{k=1}^r \sum_{n=1}^N \boldsymbol{\Gamma}^k_1 \otimes \cdots \otimes (\mathbf{A}^k_n - \mathbf{B}^k_n) \otimes \cdots \otimes \boldsymbol{\Gamma}^k_N \Big\|_F \\ &\leq \sum_{k=1}^r \sum_{n=1}^N \big\| \boldsymbol{\Gamma}^k_1 \otimes \cdots \otimes (\mathbf{A}^k_n - \mathbf{B}^k_n) \otimes \cdots \otimes \boldsymbol{\Gamma}^k_N \big\|_F \\ &= \alpha^{N-1} \sum_{k=1}^r \sum_{n=1}^N \epsilon^k_n \overset{(a)}{\leq} \alpha^{N-1}\sqrt{Nr}\epsilon,\end{aligned} \tag{40}$$

where the inequality (a) follows from $\|(\epsilon^k_n)\|_1 \leq \sqrt{Nr}\ \|(\epsilon^k_n)\|_2 \leq \sqrt{Nr}\epsilon$. □

Appendix C
Discussion on Convergence of the Algorithms

The batch algorithms proposed in Section V are essentially variants of alternating minimization (AM). Establishing the convergence of AM-type algorithms in general is challenging and only known for limited cases. Here, we first present a well-known convergence result for AM-type algorithms in Lemma 9 and discuss why our algorithms STARK and TeFDiL do not satisfy the requirements of this lemma. Then, we show a possible approach for proving convergence of STARK. We do not discuss convergence analysis of OSubDil here since it does not fall in the batch AM framework that we discuss here. We leave formal convergence results of our algorithms as open problems for future work.

First, let us state the following standard convergence result for alternating minimization-type algorithms.

Lemma 9 (Proposition 2.7.1, [47])**.** *Consider the problem*

$$\min_{\mathbf{x}=(\mathbf{x}_1,\ldots,\mathbf{x}_M)\in\mathcal{E}=\mathcal{E}_1\times\mathcal{E}_2\times\cdots\times\mathcal{E}_M} f(\mathbf{x}),$$

where $\mathcal{E}_i$ are closed convex subsets of the Euclidean space. Assume that $f(\cdot)$ is a continuous differentiable over the set $\mathcal{E}$. Suppose for each i and all $\mathbf{x} \in \mathcal{E}$, the minimum

$$\min_{\xi\in\mathcal{E}_i} f(\mathbf{x}_1, \cdots, \mathbf{x}_{i-1}, \xi, \mathbf{x}_{i+1}, \cdots, \mathbf{x}_M)$$

is uniquely attained. Then every limit point of the sequence $\{\mathbf{x}(t)\}$ generated by block coordinate descent method is a stationary point of $f(\cdot)$.

The result of Lemma 9 cannot be used for TeFDiL since its dictionary update stage does not have a unique minimizer (nonconvex minimization problem with multiple global minima)). Moreover, as discussed in Section V-B, TeFDiL only returns an inexact solution in the dictionary update stage.

Similarly, this result cannot be used to show convergence of STARK to a stationary point of Problem (8) since, as discussed in Section V-A, STARK returns an inexact solution in the dictionary update stage. However, we show next that dropping the unit column-norm constraint allows us to provide certain convergence guarantees. The unit column-norm constraint is essential in standard DL algorithms since in its absence, the ℓ_1 norm regularization term encourages undesirable solutions where $\|\mathbf{X}\|_F$ is very small while $\|\mathbf{D}\|_F$ is very large. However, in the regularization-based LSR-DL problem, the additional regularization term $\|\underline{\mathbf{D}}^\pi\|_{\mathrm{str}}$ ensures this does not happen. Therefore, dropping the unit column-norm constraint is sensible in this problem.

Let us discuss what guarantees we are able to obtain after relaxing the constraint set $\mathcal{D}_{m\times p}$. Consider the problem

$$\min_{\mathbf{D}\in\mathbb{R}^{m\times p},\mathbf{X}} \|\mathbf{Y}-\mathbf{D}\mathbf{X}\|_F^2 + \lambda_1 \|\underline{\mathbf{D}}^\pi\|_{\mathrm{str}} + \lambda\|\mathbf{X}\|_{1,1}. \qquad (41)$$

We show in Proposition 1 that under the following assumptions, STARK converges to a stationary point of Problem (41) (when the normalization step is not enforced). Then we discuss how this problem is related to Problem (8).

Assumption 4. *Consider the sequence* $\big(\mathbf{D}(t),\mathbf{X}(t)\big)$ *generated by STARK. We assume that for all* $t\geq 0$*:*

I) Classical optimality conditions for the lasso problem (see Tibshirani [48]) are satisfied.

II) $\mathbf{X}(t)$ *is full row-rank at all* t*.*

Proposition 1. *Under Assumption 4, STARK converges to a stationary point of problem* (41).

Proof. We invoke Lemma 9 to show the convergence of STARK. To use this lemma, the minimization problem w.r.t. each block needs to correspond to a closed convex constraint set and also needs to have a unique minimizer.

In the sparse coding stage, given Assumption 4-I, the minimizer of the lasso problem is unique. In the dictionary update stage of STARK, the objective of problem (41) is strongly convex w.r.t. $\mathbf{D}$ under Assumption 4-II and thus has a unique minimizer. Moreover, the constraint set $\mathbb{R}^{p\times L}$ is closed and convex. To utilize Lemma 9, it remains to show that this minimum is actually attained by ADMM. To this end, we restate Problem (20) as

$$\min_{\underline{\mathbf{D}}^\pi,\underline{\widetilde{\mathbf{W}}}} f_1(\underline{\mathbf{D}}^\pi) + f_2(\underline{\widetilde{\mathbf{W}}}) \qquad \text{s.t.} \quad \underline{\widetilde{\mathbf{W}}} = \mathcal{H}\underline{\mathbf{D}}^\pi, \qquad (42)$$

where $f_1(\underline{\mathbf{D}}^\pi) = \frac{1}{2}\|\mathbf{Y}-\mathbf{D}\mathbf{X}\|_F^2$ ($\mathbf{D}\mathbf{X}$ is a linear function of $\underline{\mathbf{D}}^\pi$) and $f_2(\underline{\widetilde{\mathbf{W}}}) = \lambda_1\sum_{n=1}^N \big\|(\underline{\mathbf{W}}_n)_{(n)}\big\|_*$. It is clear that $\mathcal{H}\mathcal{H}^*$ is convertible. Therefore, according to [49][Chapter 3, Proposition 4.2], the ADMM algorithm converges to the unique minimizer of Problem (20).

□

So far we discussed convergence of STARK to Problem (41) while our identifiability results are for problem (8). There is, however, a strong connection between minimization Problems (8) and (41): for each local minimum $\widehat{\mathbf{D}}$ of problem (8), there exists an $\widehat{\mathbf{X}}$ such that $(\widehat{\mathbf{D}},\widehat{\mathbf{X}})$ is a local minimum of (41). Define $\ell_{\mathbf{Y}}^{\mathrm{reg}}(\mathbf{D},\mathbf{X}) = \frac{1}{L}\|\mathbf{Y}-\mathbf{D}\mathbf{X}\|_F^2 + \lambda_1\|\underline{\mathbf{D}}^\pi\|_{\mathrm{str}} + \frac{\lambda}{L}\|\mathbf{X}\|_{1,1}$. Consider any $\widehat{\mathbf{D}}$ that is a local minimum of (8) and let $\widehat{\mathbf{X}} = \mathrm{argmin}_{\mathbf{X}\in\mathbb{R}^{p\times L}} \ell_{\mathbf{Y}}^{\mathrm{reg}}(\widehat{\mathbf{D}},\mathbf{X})$. We have $\ell_{\mathbf{Y}}^{\mathrm{reg}}(\widehat{\mathbf{D}},\widehat{\mathbf{X}}) = F_{\mathbf{Y}}^{\mathrm{reg}}(\widehat{\mathbf{D}})$. Since $\widehat{\mathbf{D}}$ is a local minimizer of $F_{\mathbf{Y}}^{\mathrm{reg}}(\mathbf{D})$, $F_{\mathbf{Y}}^{\mathrm{reg}}(\widehat{\mathbf{D}}) \leq F_{\mathbf{Y}}^{\mathrm{reg}}(\mathbf{D})$ for any $\mathbf{D}$ in the local neighborhood of $\widehat{\mathbf{D}}$. Also by definition, $F_{\mathbf{Y}}^{\mathrm{reg}}(\mathbf{D}) \leq \ell_{\mathbf{Y}}^{\mathrm{reg}}(\mathbf{D},\mathbf{X})$ for any $\mathbf{X}$. Thus, $\ell_{\mathbf{Y}}^{\mathrm{reg}}(\widehat{\mathbf{D}},\widehat{\mathbf{X}}) \leq \ell_{\mathbf{Y}}^{\mathrm{reg}}(\mathbf{D},\mathbf{X})$ for any $(\mathbf{D},\mathbf{X})$ in the local neighborhood of $(\widehat{\mathbf{D}},\widehat{\mathbf{X}})$, meaning that $(\widehat{\mathbf{D}},\widehat{\mathbf{X}})$ is a local minimizer of (41). Since we showed in Section IV that a local minimum $\mathbf{D}^*$ of (8) is close to the true dictionary $\mathbf{D}^0$, we can say there is a local minimum $(\mathbf{D}^*,\mathbf{X}^*)$ of (41) close to $\mathbf{D}^0$. So our recovery result for (8) can apply to our proposed algorithm for solving (41) as well.

References

[1] M. Ghassemi, Z. Shakeri, A. D. Sarwate, and W. U. Bajwa, "STARK: Structured dictionary learning through rank-one tensor recovery," in *Proc. IEEE 7th Int. Workshop Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, 2017, pp. 1–5.

[2] M. Ghassemi, Z. Shakeri, W. U. Bajwa, and A. D. Sarwate, "Sample complexity bounds for low-separation-rank dictionary learning," in *Proc. 2019 IEEE Int. Symp. Inf. Theory*, July 2019.

[3] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T.-W. Lee, and T. J. Sejnowski, "Dictionary learning algorithms for sparse representation," *Neural Computation*, vol. 15, no. 2, pp. 349–396, 2003.

[4] I. Tosic and P. Frossard, "Dictionary learning," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 27–38, March 2011.

[5] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, November 2006.

[6] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Machine Learning Research*, vol. 11, pp. 19–60, 2010.

[7] L. R. Tucker, "Implications of factor analysis of three-way matrices for measurement of change," *Prob. Meas. Change*, pp. 122–137, 1963.

[8] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an explanatory multi-modal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, pp. 1–84, 1970.

[9] Z. Shakeri, A. D. Sarwate, and W. U. Bajwa, "Sample complexity bounds for dictionary learning from vector- and tensor-valued data," in *Information Theoretic Methods in Data Science*. Cambridge, UK: Cambridge University Press, 2019, ch. 5.

[10] C. F. Van Loan, "The ubiquitous Kronecker product," *J. Computational and Appl. Math.*, vol. 123, no. 1, pp. 85–100, 2000.

[11] S. Hawe, M. Seibert, and M. Kleinsteuber, "Separable dictionary learning," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition (CVPR)*, 2013, pp. 438–445.

[12] F. Roemer, G. Del Galdo, and M. Haardt, "Tensor-based algorithms for learning multidimensional separable dictionaries," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Process.*, 2014, pp. 3963–3967.

[13] C. F. Dantas, M. N. da Costa, and R. da Rocha Lopes, "Learning dictionaries as a sum of Kronecker products," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 559–563, March 2017.

[14] S. Zubair and W. Wang, "Tensor dictionary learning with sparse Tucker decomposition," in *Proc. IEEE 18th Int. Conf. Digital Signal Process. (DSP)*, 2013, pp. 1–6.

[15] Z. Shakeri, W. U. Bajwa, and A. D. Sarwate, "Minimax lower bounds on dictionary learning for tensor data," *IEEE Trans. Inf. Theory*, vol. 64, no. 4, pp. 2706–2726, April 2018.

[16] Z. Shakeri, A. D. Sarwate, and W. U. Bajwa, "Identifiability of Kronecker-structured dictionaries for tensor data," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 5, pp. 1047 – 1062, 2018.

[17] N. Cressie and H.-C. Huang, "Classes of nonseparable, spatio-temporal stationary covariance functions," *J. American Statistical Association*, vol. 94, no. 448, pp. 1330–1339, 1999.

[18] G. Beylkin and M. J. Mohlenkamp, "Numerical operator calculus in higher dimensions," *Proceedings of the National Academy of Sciences*, vol. 99, no. 16, pp. 10 246–10 251, 2002.

[19] T. Tsiligkaridis and A. O. Hero, "Covariance estimation in high dimensions via Kronecker product expansions," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5347–5360, 2013.

[20] J. Håstad, "Tensor rank is NP-complete," *J. Algorithms*, vol. 11, no. 4, pp. 644–654, 1990.

[21] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, August 2009.

[22] I. V. Oseledets, "Tensor-train decomposition," *SIAM J. Scientific Computing*, vol. 33, no. 5, pp. 2295–2317, 2011.

[23] A. Novikov, D. Podoprikhin, A. Osokin, and D. P. Vetrov, "Tensorizing neural networks," in *Proc. Advances in Neural Inform. Process. Syst.*, 2015, pp. 442–450.

[24] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, 2017.

[25] S. Arora, R. Ge, and A. Moitra, "New algorithms for learning incoherent and overcomplete dictionaries," in *Proc. 25th Annu. Conf. Learning Theory*, ser. JMLR: Workshop and Conf. Proc., vol. 35, 2014, pp. 1–28.

[26] A. Agarwal, A. Anandkumar, P. Jain, P. Netrapalli, and R. Tandon, "Learning sparsely used overcomplete dictionaries," in *Proc. 27th Annu. Conf. Learning Theory*, ser. JMLR: Workshop and Conf. Proc., vol. 35, no. 1, 2014, pp. 1–15.

[27] R. Gribonval, R. Jenatton, and F. Bach, "Sparse and spurious: Dictionary learning with noise and outliers," *IEEE Trans. Inf. Theory*, vol. 61, no. 11, pp. 6298–6319, 2015.

[28] K. Schnass, "On the identifiability of overcomplete dictionaries via the minimisation principle underlying K-SVD," *Appl. and Computational Harmonic Anal.*, vol. 37, no. 3, pp. 464–491, 2014.

[29] C. F. Caiafa and A. Cichocki, "Multidimensional compressed sensing and their applications," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 3, no. 6, pp. 355–380, 2013.

[30] E. Schwab, B. Haeffele, N. Charon, and R. Vidal, "Separable dictionary learning with global optimality and applications to diffusion MRI," *arXiv preprint arXiv:1807.05595*, 2018.

[31] C. F. Dantas, J. E. Cohen, and R. Gribonval, "Learning fast dictionaries for sparse representations using low-rank tensor decompositions," in *Proc. Int. Conf. Latent Variable Analysis and Signal Separation*. Springer, 2018, pp. 456–466.

[32] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2121–2130, 2010.

[33] E. Dohmatob, A. Mensch, G. Varoquaux, and B. Thirion, "Learning brain regions via large-scale online structured sparse dictionary learning," in *Proc. Advances Neural Inf. Process. Syst.*, 2016, pp. 4610–4618.

[34] W. Rudin, *Principles of mathematical analysis*. McGraw-hill New York, 1964, vol. 3.

[45] Z. Zhang and S. Aeron, "Denoising and completion of 3d data via multidimensional dictionary learning," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*. AAAI Press, 2016, pp. 2371–2377.

[35] R. Gribonval, R. Jenatton, F. Bach, M. Kleinsteuber, and M. Seibert, "Sample complexity of dictionary learning and other matrix factorizations," *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3469–3486, 2015.

[36] K. Wimalawarne, M. Sugiyama, and R. Tomioka, "Multitask learning meets tensor factorization: Task imputation via convex optimization," in *Proc. Advances in Neural Inform. Process. Syst.*, 2014, pp. 2825–2833.

[37] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[38] B. Romera-Paredes, H. Aung, N. Bianchi-Berthouze, and M. Pontil, "Multilinear multitask learning," in *Proc. 30th Int. Conf. Mach. Learn. (ICML)*, vol. 28, no. 3, Atlanta, Georgia, USA, 2013, pp. 1444–1452.

[39] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, no. 2, p. 025010, January 2011.

[40] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.

[41] V. de Silva and L. Lim, "Tensor rank and the ill-posedness of the best low-rank approximation problem," *SIAM J. Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1084–1127, 2008.

[42] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Asilomar Conf. Signals, Syst. and Comput.*, vol. 1, 1993, pp. 40–44.

[43] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[44] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. IEEE Int. Conf. Pattern recognition (ICPR)*, 2010, pp. 2366–2369.

[46] E. J. Candes and Y. Plan, "Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements," *IEEE Trans. Inf. Theor.*, vol. 57, no. 4, pp. 2342–2359, Apr. 2011.

[47] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific Belmont, 1999.

[48] R. J. Tibshirani, "The lasso problem and uniqueness," *Electron. J. Statist.*, vol. 7, pp. 1456–1490, 2013.

[49] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Prentice Hall Englewood Cliffs, NJ, 1989, vol. 23.