Fast and Accurate Low-Rank Factorization of Compressively-Sensed Data

Vatsal Sharan*
Stanford University
vsharan@stanford.edu

Kai Sheng Tai*
Stanford University
kst@cs.stanford.edu

Peter Bailis
Stanford University
pbailis@cs.stanford.edu

Gregory Valiant
Stanford University
valiant@stanford.edu

Abstract

We consider the question of accurately and efficiently computing low-rank matrix or tensor factorizations given data compressed via random projections. This problem arises naturally in the many settings in which data is acquired via compressive sensing. We examine the approach of first performing factorization in the compressed domain, and then reconstructing the original high-dimensional factors from the recovered (compressed) factors. In both the tensor and matrix settings, we establish conditions under which this natural approach will provably recover the original factors. We support these theoretical results with experiments on synthetic data and demonstrate the practical applicability of our methods on real-world gene expression and EEG time series data.

1 Introduction

We consider the setting where we are given data that has been compressed via random projections. This setting frequently arises when data is acquired via compressive measurements [1, 2], or when high-dimensional data is projected to lower dimension in order to reduce storage and bandwidth costs [3, 4]. In the former case, the use of compressive measurement has enabled higher throughput in signal acquisition, more compact sensors, and reduced data storage costs [5, 2]. In the latter case, the use of random projections underlies many sketching algorithms for stream processing and distributed data processing applications [6].

Due to the computational benefits of working directly in the compressed domain, there has been significant interest in understanding which learning tasks can be performed on compressed data. For example, by learning SVMs directly on compressively sensed data, we can avoid the computational cost of performing sparse recovery for each example during training and inference [7]. The problem of learning from compressed data has also been considered for several other learning tasks, such as linear discriminant analysis [8], PCA [9, 10, 11], and regression [12, 13, 14].

Building off this line of work, we consider recovering low-rank matrix and tensor factorizations from compressed data. In particular, our goal is to recover the factors in their original (uncompressed) domain. We focus on the setting where the factors are sparse, and our results have implications for a variety of problems in this setting, including sparse PCA, nonnegative matrix factorization (NMF), and tensor decomposition.

This setting of compressed data with sparse factors arises in a number of important practical domains. To give two motivating examples, first consider the task of analyzing gene expression data. The gene expression levels are typically sparse, and this is a domain where compressive sensing has been successfully applied for efficient data acquisition [15]. Additionally, NMF has been widely used

*These authors contributed equally to the work.

in this domain (on uncompressed data) to cluster tissue samples corresponding to different forms of cancer, and yields sparse factors [16]. Another relevant setting is EEG data where compressive sensing has been useful for bandwidth-limited data transmission [17]. Tensor decomposition is commonly used in this domain to identify distinct modes of brain activity [18], and recovery of the factors in the original domain is essential for the interpretability of these factors.

Concretely, we model this problem in the matrix setting as follows: consider a high-dimensional data matrix $M \in \mathbb{R}^{n \times m}$ that has a rank r factorization $M = WH$, where $W \in \mathbb{R}^{n \times r}$, $H \in \mathbb{R}^{r \times m}$, and W is sparse. We are given the compressed measurements $\tilde{M} = PM$ for a known measurement matrix $P \in \mathbb{R}^{d \times n}$, where $d \leq n$. Our goal is to approximately recover the original factors W and H given the compressed data $\tilde{M}$ as accurately and efficiently as possible.

A natural approach to estimating W and H is to first factorize the compressed matrix, $\tilde{M} = \tilde{W}\tilde{H}$, and then to subsequently estimate W from $\tilde{W}$ using a sparse recovery algorithm that leverages the sparsity of the factors. We refer to this procedure as FACTORIZE-RECOVER. This approach has clear computational benefits over the alternative naive approach of first recovering each column of M, and only then performing low-rank factorization on the recovered matrix. In particular, FACTORIZE-RECOVER requires only r calls to the sparse recovery algorithm, in contrast to $m \gg r$ calls for the alternative: this difference is significant in practice, for example when m is the number of samples in the dataset and r is a small constant. The central question, however, is understanding the relationship between the factors of the compressed data, and the original factors. *Under what conditions on the sparsity, and compression matrix P is it possible to recover the "correct" factorization $\tilde{M} = (PW)H$ of the compressed data, from which the original factors can be successfully recovered?*

Our contributions. In this work, we establish conditions under which FACTORIZE-RECOVER provably succeeds, in both the matrix and tensor factorization domains. We substantiate these theoretical results with empirical validation that demonstrates both the accuracy of the recovered factors, and the computational speedup resulting from FACTORIZE-RECOVER versus the naïve and more commonly employed approach of first recovering a representation in the original uncompressed domain, and then factorizing the result.

Our main theoretical guarantee for sparse matrix factorizations, formally stated in Section 4.1, provides a simple condition under which the factors of the compressed data are the compressed factors. While the result is intuitive, the proof is delicate, and involves characterizing the likely sparsity of linear combinations of sparse vectors, exploiting graph theoretic properties of expander graphs.

Theorem 1 (informal). *Consider a rank r matrix $M \in \mathbb{R}^{n \times m}$, where $M = WH$, $W \in \mathbb{R}^{n \times r}$ and $H \in \mathbb{R}^{r \times m}$, and W is sparse. Given the compressed measurements $\tilde{M} = PM$ for a measurement matrix $P \in \mathbb{R}^{d \times n}$, under suitable conditions on n, m, d and the sparsity, $\tilde{M} = (PW)H$ is the sparsest rank r factorization of $\tilde{M}$, and hence performing sparse recovery on the columns of (PW) will yield the true factors W.*

While Theorem 1 provides guarantees on the quality of the sparsest rank r factorization, it does not directly address the algorithmic question of how to find such a factorization efficiently. For some of the settings of interest, such as sparse PCA, efficient algorithms for recovering this sparsest factorization are known, under some mild assumptions on the data [19, 10, 20]. Given this, in such settings, Theorem 1 guarantees that we can efficiently recover the correct factorization. For other matrix factorization problems, such as NMF, the current algorithmic understanding of how to recover the factorization, even on uncompressed data, is incomplete and guarantees for provable recovery require strong assumptions such as separability [21]. For this reason, one should not expect an analog of Theorem 1 to guarantee efficient recovery in general. Despite this, we do have algorithms that perform well in practice on many instances of NMF, and produce sparse factors. For example, NMF has been demonstrated to yield sparse dictionaries on real-world data [22, 23] and there is significant work on explicitly inducing sparsity in regularized variants of NMF [23, 24, 25, 26]. In light of this empirically demonstrated ability to compute sparse NMF, Theorem 1 provides theoretical grounding for why FACTORIZE-RECOVER should yield accurate reconstructions of the original factors.

We empirically verify that FACTORIZE-RECOVER succeeds for NMF by evaluating it on several real-world gene expression datasets—a domain where compressive sensing has emerged as a promising approach for high-throughput data acquisition [15, 27, 28]. We find that performing factorization

on compressed data achieves reconstruction accuracy comparable to or better than factorizing the recovered (uncompressed) data.

In addition to our results on matrix factorization, we show the following analog to Theorem 1 for tensor decomposition on compressed tensors:

Proposition 1 (informal). *Consider a rank r tensor $T \in \mathbb{R}^{n \times m_1 \times m_2}$ with factorization $T = \sum_{i=1}^{r} A_i \otimes B_i \otimes C_i$, where A is sparse. Under suitable conditions on the dimensions of the tensor, the projection dimension and the sparsity, $\tilde{T} = \sum_{i=1}^{r} (PA_i) \otimes B_i \otimes C_i$ is the unique factorization of the compressed tensor $\tilde{T}$, and hence performing sparse recovery on the columns of PA will yield the true factors A.*

As in the case of sparse PCA, there is an efficient algorithm for finding this unique tensor factorization, as tensor decomposition can be computed efficiently when the factors are linearly independent (see e.g. [29]). We also empirically validate our approach for tensor decomposition on a real-world EEG dataset, another domain where compressed sensing has been applied [17]. We demonstrate that factorizations from compressed measurements can yield interpretable factors that are useful for the efficient detection of the onset of seizures.

2 Related Work and Background

Learning from Compressed Data. As mentioned in the introduction, learning from compressed data has been widely considered, yielding strong results for many learning tasks, including linear classification [7, 8] and regression [12, 13]. However, in most of these settings the goal is to obtain a good predictive model in the compressed space itself, instead of recovering the model in the original space. A notable exception to this is previous work on performing PCA on compressed data [9, 11]; we extend this line of work by considering sparse matrix decompositions like sparse PCA and NMF.

From a technical perspective, the most relevant work is the paper of Spielman et al. [30], who consider the problem of sparse coding. Although their setting is different than ours, the technical cores of both analyses involve characterizing the sparsity patterns of linear combinations of random sparse vectors.

There is also an enormous body of algorithmic work on computing matrix and tensor decompositions more efficiently using random projections, usually by speeding up the linear algebraic routines that arise in the computation of these factorizations. This includes work on randomized SVD [31, 32], NMF [33] and tensor decomposition [34]. This work is rather different in spirit, as it leverages projections to accelerate certain components of the algorithms, but still requires repeated accesses to the original uncompressed data. In contrast, our methods apply in the setting where we are only given access to the compressed data.

Background on Compressive Sensing. Since we will repeatedly leverage a sparse recovery subroutine, it will be helpful to briefly summarize the main ideas in this domain. In the compressive sensing or sparse recovery framework, there is a sparse signal $v \in \mathbb{R}^n$ for which we are given $d \ll n$ linear measurements Pv, where $P \in \mathbb{R}^{d \times n}$ is a known measurement matrix. The goal is to recover v using the measurements Pv, and the property that v is sparse. Seminal results in compressive sensing [1, 35, 36] show that if the original solution is k-sparse then it can be exactly recovered with $d = O(k \log n)$ via a linear programming (LP) formulation. More efficient recovery algorithms than the LP for solving the problem are also known (such as SSMP [37, 38, 39]). However, these algorithms typically require more measurements in the compressed domain to achieve the same reconstruction accuracy as the LP formulation [39].

3 Factorization from Compressed Data

Notation. Let $[n]$ denote the set $\{1, 2, \ldots, n\}$. For any matrix A, we denote its ith column as A_i. Throughout, we denote the projection matrix by P. For a matrix $P \in \mathbb{R}^{d \times n}$ such that $d < n$, define:

$$\mathcal{R}_P(w) = \underset{x: Px = w}{\operatorname{argmin}} \|x\|_1 \tag{1}$$

as the sparse recovery operator on $w \in \mathbb{R}^n$. We omit the subscript P when it is clear from context.

Low-Rank Matrix Factorization. We assume that each sample is an n-dimensional column vector in uncompressed form. Hence, the uncompressed matrix $M \in \mathbb{R}^{n \times m}$ has m columns corresponding to m samples, and we assume that it has some rank-r factorization: $M = WH$, where $W \in \mathbb{R}^{n \times r}$,

$H \in \mathbb{R}^{r \times m}$, and the columns of W are k-sparse. We are given the compressed matrix $\tilde{M} = PM$ corresponding to the d-dimensional projection Pv for each sample $v \in \mathbb{R}^n$. The algorithm is as follows:

Algorithm 1: Compressed Matrix Factorization using FACTORIZE-RECOVER

Input: Compressed matrix $\tilde{M} = PM$, projection matrix P

Algorithm: *Outputs estimates* $(\hat{W}, \hat{H})$ *of* (W, H)

Compute rank-r factorization of $\tilde{M}$ to obtain factorization $\tilde{M} = \tilde{W}\tilde{H}$ and set $\hat{H} \to \tilde{H}$

Solve r sparse recovery problems (Eq. 1) for the r columns of W, set $\hat{W}_i \to \mathcal{R}(\tilde{W}_i)$

CP Tensor Decomposition. As above, we assume that each sample is n-dimensional and k-sparse. The samples are now indexed by two coordinates $y \in [m_1]$ and $z \in [m_2]$, and hence can be represented by a tensor $T \in \mathbb{R}^{n \times m_1 \times m_2}$. We assume that T has some rank-r factorization $T = \sum_{i=1}^{r} A_i \otimes B_i \otimes C_i$, where the columns of A are k-sparse. Here $\otimes$ denotes the outer product: if $a \in \mathbb{R}^n, b \in \mathbb{R}^{m_1}, c \in \mathbb{R}^{m_2}$ then $a \otimes b \otimes c \in \mathbb{R}^{n \times m_1 \times m_2}$ and $(a \otimes b \otimes b)_{ijk} = a_i b_j c_k$. This model, CP decomposition, is the most commonly used model of tensor decomposition. For a projection matrix $P \in \mathbb{R}^{d \times n}$, we are given a projected tensor $\tilde{T} \in \mathbb{R}^{d \times m_1 \times m_2}$ corresponding to a d dimensional projection Pv for each sample v.

Algorithm 2: Compressed Tensor Decomposition (TD) using FACTORIZE-RECOVER

Input: Compressed tensor $\tilde{T}$, projection matrix P

Algorithm: *Outputs estimates* $(\hat{A}, \hat{B}, \hat{C})$ *of* (A, B, C)

Compute rank-r TD of $\tilde{T}$: $\tilde{T} = \sum_{i=1}^{r} \tilde{A}_i \otimes \tilde{B}_i \otimes \tilde{C}_i$. Set $\hat{B} \to \tilde{B}$, $\hat{C} \to \tilde{C}$

Solve r sparse recovery problems (Eq. 1) for the r columns of A, set $\hat{A}_i \to \mathcal{R}(\tilde{A}_i)$

Projection Matrices. In this work, we consider sparse, binary projection (or measurement) matrices $P \in \{0, 1\}^{d \times n}$ with exactly p non-zero entries in each column. For our theoretical results, we set $p = O(\log n)$. Although the first results on compressive sensing only held for dense matrices [1, 36, 35], subsequent work has shown that sparse, binary matrices can also be used for compressive sensing [40] (we discuss this further in Section 4.1). In practice, sparse binary projection matrices can arise due to physical limitations in sensor design (e.g., where measurements are sparse and can only be performed additively) or in applications of non-adaptive group testing.

We now describe our formal results for matrix and tensor factorization.

4 Theoretical Guarantees for Factorization from Compressed Data

In this section, we establish conditions under which FACTORIZE-RECOVER will provably succeed for matrix and tensor decomposition on compressed data.

4.1 Sparse Matrix Factorization

The main idea is to show that with high probability, $\tilde{M} = (PW)H$ is the *sparsest* possible factorization of $\tilde{M}$ in the following sense: for any other factorization $\tilde{M} = W'H'$, W' has strictly more non-zero entries than (PW). It follows that the factorization $(PW)H$ is the optimal solution for a sparse matrix factorization of $\tilde{M}$ that penalizes non-zero entries of $\tilde{W}$. To show this uniqueness property, we require that the projection matrices satisfy certain structural conditions, namely that they correspond to adjacency matrices of *bipartite expander* graphs [41], which we subsequently define.

We also require that the projection matrix P is suitable for sparse recovery: given a good factorization $\tilde{M} = (PW)H$, we still need to recover the original factors W from PW. This property follows from existing work on sparse recovery: Theorem 3 of Berinde et al. [40] shows that the recovery procedure succeeds with high probability for P satisfying the previously-mentioned expander property.

We now formally state our theorem:

Theorem 1. *Consider a rank r matrix $M \in \mathbb{R}^{n \times m}$ which has factorization $M = WH$, for $H \in \mathbb{R}^{r \times m}$ and $W \in \mathbb{R}^{n \times r}$. Assume H has full row rank and $W = B \odot Y$, where each column of B has k non-zero entries chosen uniformly at random, each entry of Y is an independent $N(0, 1)$ random*

variable,[2] *and* $\odot$ *denotes the elementwise product. Assume* $k > C$*, where* C *is a fixed constant. Consider the projection matrix* $P \in \{0,1\}^{d \times n}$ *where each column of* P *has* $p = O(\log n)$ *non-zero entries chosen independently and uniformly at random. Assume* $d = \Omega((r+k)\log n)$*. Let* $\tilde{M} = PM$*. Note that* $\tilde{M}$ *has one possible factorization* $\tilde{M} = \tilde{W}H$ *where* $\tilde{W} = PW$*. For some fixed* $\beta > 0$*, with failure probability at most* $(r/n)e^{-\beta k} + (1/n^5)$*,* $\tilde{M} = \tilde{W}H$ *is the sparsest possible factorization in terms of the left factors: for any other rank* r *factorization* $\tilde{M} = W'H'$*,* $\|\tilde{W}\|_0 < \|W'\|_0$*.*

Theorem 1 shows that if the columns of W are k-sparse, then projecting into $\Omega((r+k)\log n)$ dimensions preserves uniqueness, with failure probability at most $(r/n)e^{-\beta k} + (1/n)^5$, for some constant $\beta > 0$. As real-world matrices have been empirically observed to be typically close to low rank, the (r/n) term is usually small for practical applications. Note that the requirement for the projection dimension being at least $\Omega((r+k)\log n)$ is close to optimal, as even being able to uniquely recover a k-sparse n-dimensional vector x from its projection Px requires the projection dimension to be at least $\Omega(k \log n)$; we also cannot hope for uniqueness for projections to dimensions below the rank r. We emphasize that the assumption on the nonzero entries of W being Gaussian is stated for simplicity and any other continuous distribution for the non-zero entries will also suffice. We provide a proof sketch below, with the full proof deferred to Appendix B.

Proof sketch. We first show a simple Lemma that for any other factorization $\tilde{M} = W'H'$, the column space of W' and $\tilde{W}$ must be the same (Lemma 5 in the Appendix). Using this, for any other factorization $\tilde{M} = W'H'$, the columns of W' must lie in the column space of the columns of $\tilde{W}$, and hence our goal will be to prove that the columns of $\tilde{W}$ are the sparsest vectors in the column space of $\tilde{W}$, which implies that for any other alternative factorization $\tilde{M} = W'H'$, $\|\tilde{W}\|_0 < \|W'\|_0$.

The outline of the proof is as follows. It is helpful to think of the matrix $\tilde{W} \in \mathbb{R}^{d \times r}$ as corresponding to the adjacency matrix of an unweighted bipartite graph G with r nodes on the left part U_1 and d nodes on the right part U_2, and an edge from a node $u \in U_1$ to a node $v \in U_2$ if the corresponding entry of $\tilde{W}$ is non-zero. For any subset S of the columns of $\tilde{W}$, define $N(S)$ to be the subset of the rows of $\tilde{W}$ which have a non-zero entry in at least one of the columns in S. In the graph representation G, $N(S)$ is simply the neighborhood of a subset S of vertices in the left part U_1. In part (a) we argue that the if we take any subset S of the columns of $\tilde{W}$, $|N(S)|$ will be large. This implies that taking a linear combination of all the S columns will result in a vector with a large number of non-zero entries—unless the non-zero entries cancel in many of the columns. In part (b), by using the properties of the projection matrix P and the fact that the non-zero entries of the original matrix W are drawn from a continuous distribution, we show this happens with zero probability.

The property of the projection matrix that is key to our proof is that it is the adjacency matrix of a bipartite expander graph, defined below.

Definition 1. *Consider a bipartite graph* R *with* n *nodes on the left part and* d *nodes on the right part such that every node in the left part has degree* p*. We call* R *a* $(\gamma n, \alpha)$ *expander if every subset of at most* $t \leq \gamma n$ *nodes in the left part has at least* $\alpha t p$ *neighbors in the right part.*

It is not difficult to show that a randomly chosen matrix P with p non-zero entries per column is the adjacency matrix of a $(\gamma n, 4/5)$ expander for $\gamma n = d/(pe^5)$ with failure probability $(1/n^5)$, if $p = O(\log n)$. We prove this in Lemma 6 in the Appendix. Note that part (a) is a requirement on the graph G for the matrix $\tilde{W}$ being a bipartite expander. In order to show that G is a bipartite expander, we show that with high probability P is a bipartite expander, and the matrix B corresponding to the non-zero entries of W is also a bipartite expander. G is a cascade of these bipartite expanders, and hence is also a bipartite expander.

For part (b), we need to deal with the fact that the entries of $\tilde{W}$ are no longer independent because the projection step leads to each entry of $\tilde{W}$ being the sum of multiple entries of W. However, the structure of P lets us control the dependencies, as each entry of W appears at most p times in $\tilde{W}$. Note that for a linear combination of any subset of S columns, $|N(S)|$ rows have non-zero entries in at least one of the S columns, and $|N(S)|$ is large by part (a). Since each entry of W appears at most

[2] In fact, the result also holds when the entries of Y are drawn from any other continuous distribution. In particular, it holds in the NMF setting where H is non-negative and W is the absolute value of independent standard Gaussians.

p times in $\tilde{W}$, it follows that with high probability, at most $|S|p$ out of the $|N(S)|$ rows with non-zero entries be zeroed out in any linear combination of the S columns. Therefore, if $|N(S)| - |S|p$ is large enough, then any linear combination of S columns has a large number of non-zero entries and is not sparse. This implies that the columns of $\tilde{W}$ are the sparsest columns in their column space. We defer the full proof to the Appendix.

4.2 Tensor Decomposition

It is easy to show uniqueness for tensor decomposition after random projection since tensor decomposition is unique under mild conditions on the factors [42, 43]. Formally:

Proposition 1. *Consider a rank r tensor $T \in \mathbb{R}^{n \times m_1 \times m_2}$ which has factorization $T = \sum_{i=1}^{r} A_i \otimes B_i \otimes C_i$, for $A \in \mathbb{R}^{n \times r}$, $B \in \mathbb{R}^{m_1 \times r}$ and $C \in \mathbb{R}^{m_1 \times r}$. Assume B and C have full column rank and $A = X \odot Y$, where each column of X has exactly k non-zero entries chosen uniformly at random, and each entry of Y is an independent $N(0,1)$ random variable. Assume $k > C$, where C is a fixed constant. Consider a projection matrix $P \in \{0,1\}^{d \times n}$ with $d = \Omega((r+k)\log n)$ where each column of P has exactly $p = O(\log n)$ non-zero entries chosen independently and uniformly at random. Let $\tilde{T}$ be the projection of T obtained by projecting the first dimension. Note that $\tilde{T}$ has one possible factorization $\tilde{T} = \sum_{i=1}^{r} (PA_i) \otimes B_i \otimes C_i$. For some fixed $\beta > 0$, with failure probability at most $(r/n)e^{-\beta k} + (1/n^5)$, PA has full column rank, and hence this is a unique factorization of $\tilde{T}$.*

Note that efficient algorithms are known for recovering tensors with linearly independent factors [29], hence factorizations in the compressed domain can provably recover projections of the original factors under the conditions of Proposition 1. We also show that we can provably recover factorizations in the compressed space using variants of the popular alternating least squares algorithm for tensor decomposition, though these algorithms require stronger assumptions on the tensor such as incoherence. These guarantees are stated in Section C in the Appendix.

The proof of Proposition 1 is direct given the results established in Theorem 1. We use the fact that tensors have a unique decomposition whenever the underlying factors $(PA), B, C$ are full column rank. By our assumption, B and C are given to be low rank. The key step is that by the proof of Theorem 1, the columns of PA are the sparsest columns in their column space. Therefore, they must be linearly independent, as otherwise the all zero vector will lie in their column space. Therefore, (PA) has full column rank, and Proposition 1 follows.

5 Experiments

We support our theoretical uniqueness results with experiments on real and synthetic data. On synthetically generated matrices where the ground-truth factorizations are known, we show that standard algorithms for computing sparse PCA and NMF converge in practice to the desired solutions in the compressed space. We then demonstrate the practical applicability of performing direct factorization on compressed data with experiments on gene expression and EEG time series datasets.

5.1 Synthetic Data

We provide empirical evidence that standard algorithms for sparse PCA and NMF converge in practice to the desired sparse factorization $\tilde{M} = (PW)H$. For sparse PCA, we use alternating minimization with LARS [44], and for NMF, we use projected gradient descent [45].[3] Additionally, we evaluate the quality of the factors obtained after sparse recovery by measuring the approximation error of the recovered factors $\hat{W}$ relative to the true factors W.

We generate synthetic data following the conditions of Theorem 1. For sparse PCA, we sample matrices $W = B \odot Y$ and H, where each column of $B \in \{0,1\}^{n \times r}$ has k non-zero entries chosen uniformly at random, $Y_{ij} \overset{\text{iid}}{\sim} N(0,1)$, and $H_{ij} \overset{\text{iid}}{\sim} N(0,1)$. For NMF, an elementwise absolute value function is applied to the values sampled from this distribution. The noisy data matrix is $M = WH + \mathcal{E}$, where the noise term $\mathcal{E}$ is a dense random Gaussian matrix scaled such that $\|\mathcal{E}\|_F / \|WH\|_F = 0.1$. We observe $\tilde{M} = PM$, where P has 5 non-zero entries per column.

Figure 1 shows our results on synthetic data with $m = 2000$, $n = 2000$, and $r = 10$ for projection dimension $d \in \{200, 400, 800\}$. For small column sparsities k relative to the projection dimension

[3] For sparse PCA, we report results for the setting of the ℓ_1 regularization parameter that yielded the lowest approximation error. We did not use an ℓ_1 penalty for NMF.

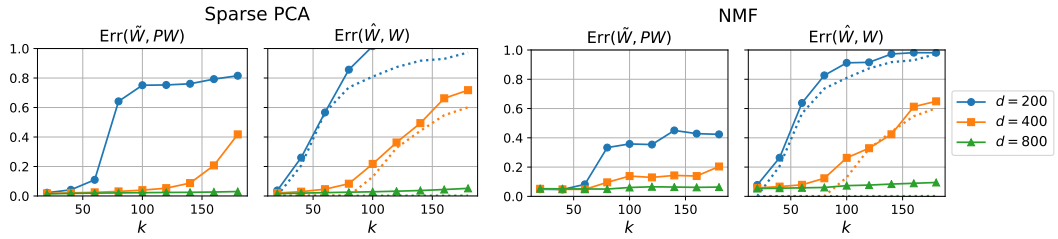

Figure 1: Approximation errors on synthetic data with varying column sparsity k of W and projection dimension d, where $\mathrm{Err}(X, X_*) := \|X - X_*\|_F / \|X_*\|_F$ is the normalized Frobenius error. Sparse PCA and NMF successfully recover the desired factorization $\tilde{M} = (PW)H$ when d is sufficiently large relative to the sparsity. The right panels show approximation errors of W relative to the recovered $\hat{W}$. The dotted lower bounds show recovery error given (exact) PW.

d, the estimated compressed left factors $\tilde{W}$ are good approximations to the desired solutions PW. Encouragingly, we find that the recovered solutions $\hat{W} = \mathcal{R}(\tilde{W})$ are typically only slightly worse in approximation error than $\mathcal{R}(PW)$, the solution recovered when the projection of W is known exactly. Thus, we perform almost as well as the idealized setting where we are *given* the correct factorization $(PW)H$.

5.2 NMF on Gene Expression Data

NMF is a commonly-used method for clustering gene expression data, yielding interpretable factors in practice [16, 46]. In the same domain, compressive sensing techniques have emerged as a promising approach for efficiently measuring the (sparse) expression levels of thousands of genes using compact measurement devices [15, 27, 28].[4] We evaluated our proposed NMF approach on gene expression datasets targeting three disease classes: embryonal central nervous system tumors [47], lung carcinomas [48], and leukemia [49] (Table 1). Each dataset is represented as a real-valued matrix where the ith row denotes expression levels for the ith gene across each sample.

Experimental Setup. For all datasets, we fixed a rank of $r = 10$ following previous clustering analyses in this domain [16, 46]. For each data matrix $M \in \mathbb{R}^{n \times m}$, we simulated compressed measurements $\tilde{M} \in \mathbb{R}^{d \times m}$ by projecting the feature dimension: $\tilde{M} = PM$. We ran projected gradient descent [45] for 250 iterations, which was sufficient to reach convergence on our datasets.

Computation Time. Computation time for NMF on all 3 datasets (Table 1) is dominated by the cost of solving instances of the ℓ_1-minimization problem (1). As a result, FACTORIZE-RECOVER achieves much lower runtime as it requires a factor of m/r fewer calls to the sparse recovery procedure. While fast iterative recovery procedures such as SSMP [39] achieve faster recovery times, we found that they require approximately $2\times$ the number of measurements to achieve comparable accuracy to LP-based sparse recovery.

Reduction in Approximation Error. For a fixed number of measurements d, we observe that the FACTORIZE-RECOVER procedure achieves lower approximation error than the alternative method of recovering prior to factorizing (Figure 2). While this phenomenon is perhaps counter-intuitive, it can be understood as a consequence of the sparsifying effect of NMF. Recall that for NMF, we model each column of the compressed data $\tilde{M}$ as a nonnegative linear combination of the columns of $\tilde{W}$. Due to the nonnegativity constraint on the entries of $\tilde{W}$, we expect the average sparsity of the columns of $\tilde{W}$ to be at least that of the columns of $\tilde{M}$. Therefore, if $\tilde{W}$ is a good approximation of PW, we should expect that the sparse recovery algorithm will recover the columns of W at least as accurately as the columns of M, given a fixed number of measurements.

5.3 Tensor Decomposition on EEG Time Series Data

EEG readings are typically organized as a collection of time series, where each series (or channel) is a measurement of electrical activity in a region of the brain. Order-3 tensors can be derived from this data by computing short-time Fourier transforms (STFTs) for each channel, yielding a tensor where each slice is a time-frequency matrix. Tensor decomposition can then be applied to cluster patterns of

[4] The measurement matrices for these devices can be modeled as sparse binary matrices since each dimension of the acquired signal corresponds to the measurement of a small set of gene expression levels.

Table 1: Summary of DNA microarray gene expression datasets, along with runtime (seconds) for each stage of the NMF pipeline on compressed data. FACTORIZE-RECOVER runs only r instances of sparse recovery, as opposed to the m instances used by the alternative, RECOVER-FACTORIZE.

Dataset	# Samples	# Features	RECOVER-FAC.		FAC.-RECOVER	
			Recovery	NMF	NMF	Recovery
CNS tumors	266	7,129	76.1	2.7	0.6	5.4
Lung carcinomas	203	12,600	78.8	4.0	0.8	9.3
Leukemia	435	54,675	878.4	39.6	6.9	55.0

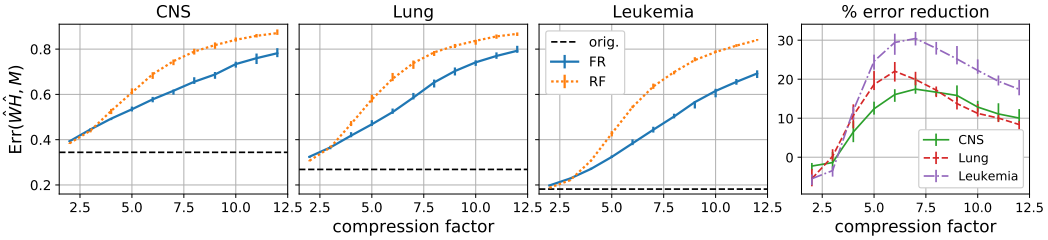

Figure 2: Normalized reconstruction errors $\|\hat{W}\hat{H} - M\|_F / \|M\|_F$ for NMF on gene expression data with varying compression factors n/d. **FR** (blue, solid) is FACTORIZE-RECOVER, **RF** (orange, dotted) is the alternative of recovering and then factorizing. The horizontal dashed line is the approximation error when M is decomposed in the original space. Perhaps surprisingly, when $n/d > 3$, we observe a reduction in approximation error when compressed data is first factorized. This phenomenon is a consequence of the compressed factors $\tilde{W}$ having higher sparsity than the compressed matrix $\tilde{M}$ (see the text for further discussion).

neural activity [18]. We experimented with tensor decomposition on a compressed tensor derived from the CHB-MIT Scalp EEG Database [50]. In the original space, this tensor has dimensions $27804 \times 303 \times 23$ (time $\times$ frequency $\times$ channel), corresponding to 40 hours of data (see Appendix A for further preprocessing details). The tensor was randomly projected along the temporal axis. We computed a rank-10 non-negative decomposition of this tensor using projected CP-ALS [29].

Factorization Accuracy. At $5\times$ compression, the recovered temporal factors match the temporal factors obtained by tensor decomposition on the uncompressed data with a median Pearson correlation of 0.90. This decreases to 0.78 and 0.53 correlation at $10\times$ and $20\times$ compression respectively. The median correlation for the other (unprojected) factors is at least 0.98 for all three compression levels. Thus, at $5\times$ compression, we still achieve accurate recovery of the compressed factor.

Factor Interpretability for Detecting Seizures. The EEG time series dataset was recorded from patients suffering from epileptic seizures [50]. We found that a tensor decomposition of the time series yields a factor that correlates with the onset of seizures (Figure 3). At $5\times$ compression, the recovered factor qualitatively retains the interpretability of the factor obtained by decomposing the tensor in the original space.

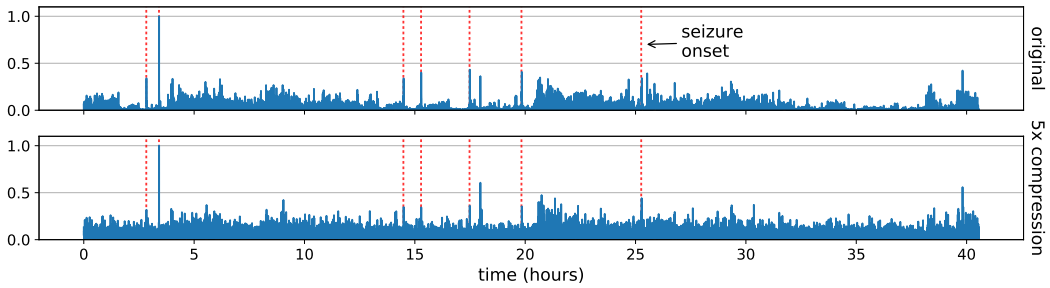

Figure 3: Visualization of a factor from the tensor decomposition of EEG data that correlates with the onset of seizures in a patient (red dotted lines). The factor recovered from a $5\times$ compressed version of the tensor (bottom) retains the peaks that are indicative of seizures.

6 Conclusion

In this work, we analyzed low-rank matrix and tensor decomposition on compressed data. Our main theoretical contribution is a novel uniqueness result for the matrix factorization case that relates sparse solutions in the original and compressed domains. We provided empirical evidence on real and synthetic data that accurate recovery can be achieved in practice. More generally, our results in this setting can be interpreted as the unsupervised analogue to previous work on supervised learning on compressed data.

Acknowledgments

This research was supported in part by affiliate members and other supporters of the Stanford DAWN project—Google, Intel, Microsoft, NEC, Teradata, and VMware, industrial gifts and support from Toyota Research Institute, Juniper Networks, Keysight Technologies, Hitachi, Facebook, Northrop Grumman, and NetApp. The authors were also supported by DARPA under No. FA8750-17-2-0095 (D3M), ONR Award N00014-17-1-2562, NSF Award CCF-1704417, and a Sloan Research Fellowship.

References

[1] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4): 1289–1306, 2006.

[2] Emmanuel J Candès and Michael B Wakin. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2):21–30, 2008.

[3] Jarvis Haupt, Waheed U Bajwa, Michael Rabbat, and Robert Nowak. Compressed sensing for networked data. *IEEE Signal Processing Magazine*, 25(2):92–101, 2008.

[4] Amir M Abdulghani, Alexander J Casson, and Esther Rodriguez-Villegas. Compressive sensing scalp eeg signals: implementations and practical performance. *Medical & biological engineering & computing*, 50(11):1137–1145, 2012.

[5] Marco F Duarte, Mark A Davenport, Dharmpal Takhar, Jason N Laska, Ting Sun, Kevin F Kelly, and Richard G Baraniuk. Single-pixel imaging via compressive sampling. *IEEE signal processing magazine*, 25(2):83–91, 2008.

[6] Graham Cormode, Minos Garofalakis, Peter J Haas, and Chris Jermaine. Synopses for massive data: Samples, histograms, wavelets, sketches. *Foundations and Trends in Databases*, 4(1–3): 1–294, 2012.

[7] Robert Calderbank, Sina Jafarpour, and Robert Schapire. Compressed learning: Universal sparse dimensionality reduction and learning in the measurement domain. *preprint*, 2009.

[8] Robert J Durrant and Ata Kabán. Compressed fisher linear discriminant analysis: Classification of randomly projected data. In *KDD*, pages 1119–1128. ACM, 2010.

[9] James E Fowler. Compressive-projection principal component analysis. *IEEE Transactions on Image Processing*, 18(10):2230–2242, 2009.

[10] Tianyi Zhou and Dacheng Tao. Godec: Randomized low-rank & sparse matrix decomposition in noisy case. In *International conference on machine learning*. Omnipress, 2011.

[11] Wooseok Ha and Rina Foygel Barber. Robust pca with compressed data. In *NIPS*, pages 1936–1944, 2015.

[12] Shuheng Zhou, John Lafferty, and Larry Wasserman. Compressed and privacy-sensitive sparse regression. *IEEE Transactions on Information Theory*, 55(2):846–866, 2009.

[13] Odalric Maillard and Rémi Munos. Compressed least-squares regression. In *NIPS*, pages 1213–1221, 2009.

[14] Ata Kabán. New bounds on compressive linear least squares regression. In *Artificial Intelligence and Statistics*, pages 448–456, 2014.

[15] Farzad Parvaresh, Haris Vikalo, Sidhant Misra, and Babak Hassibi. Recovering sparse signals using sparse measurement matrices in compressed dna microarrays. *IEEE Journal of Selected Topics in Signal Processing*, 2(3):275–285, 2008.

[16] Yuan Gao and George Church. Improving molecular cancer class discovery through sparse non-negative matrix factorization. *Bioinformatics*, 21(21):3970–3975, 2005.

[17] Zhilin Zhang, Tzyy-Ping Jung, Scott Makeig, and Bhaskar D Rao. Compressed sensing of eeg for wireless telemonitoring with low energy consumption and inexpensive hardware. *IEEE Transactions on Biomedical Engineering*, 60(1):221–224, 2013.

[18] Fengyu Cong, Qiu-Hua Lin, Li-Dan Kuang, Xiao-Feng Gong, Piia Astikainen, and Tapani Ristaniemi. Tensor decomposition of eeg signals: a brief review. *Journal of neuroscience methods*, 248:59–69, 2015.

[19] Massih Amini, Nicolas Usunier, and Cyril Goutte. Learning from multiple partially observed views-an application to multilingual text categorization. In *NIPS*, pages 28–36, 2009.

[20] Yash Deshpande and Andrea Montanari. Sparse pca via covariance thresholding. In *Advances in Neural Information Processing Systems*, pages 334–342, 2014.

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

[22] Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[23] Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *JMLR*, 5(Nov): 1457–1469, 2004.

[24] Stan Z Li, Xin Wen Hou, Hong Jiang Zhang, and Qian Sheng Cheng. Learning spatially localized, parts-based representation. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–I. IEEE, 2001.

[25] Jingu Kim and Haesun Park. Sparse nonnegative matrix factorization for clustering. Technical report, Georgia Institute of Technology, 2008.

[26] Robert Peharz and Franz Pernkopf. Sparse nonnegative matrix factorization with l0-constraints. *Neurocomputing*, 80:38–46, 2012.

[27] Wei Dai, Mona A Sheikh, Olgica Milenkovic, and Richard G Baraniuk. Compressive sensing dna microarrays. *EURASIP journal on bioinformatics and systems biology*, 2009(1):162824, 2008.

[28] Brian Cleary, Le Cong, Anthea Cheung, Eric S Lander, and Aviv Regev. Efficient generation of transcriptomic profiles by random composite measurements. *Cell*, 171(6):1424–1436, 2017.

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

[30] Daniel A Spielman, Huan Wang, and John Wright. Exact recovery of sparsely-used dictionaries. In *Conference on Learning Theory*, pages 37–1, 2012.

[31] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53 (2):217–288, 2011.

[32] Kenneth L Clarkson and David P Woodruff. Low rank approximation and regression in input sparsity time. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 81–90. ACM, 2013.

[33] Fei Wang and Ping Li. Efficient nonnegative matrix factorization with random projections. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 281–292. SIAM, 2010.

[34] Casey Battaglino, Grey Ballard, and Tamara G Kolda. A practical randomized cp tensor decomposition. *arXiv preprint arXiv:1701.06600*, 2017.

[35] Emmanuel J Candes and Terence Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE transactions on information theory*, 52(12):5406–5425, 2006.

[36] Emmanuel J Candes. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Mathematique*, 346(9-10):589–592, 2008.

[37] Radu Berinde, Piotr Indyk, and Milan Ruzic. Practical near-optimal sparse recovery in the l1 norm. In *Allerton*. IEEE, 2008.

[38] Piotr Indyk and Milan Ruzic. Near-optimal sparse recovery in the l1 norm. In *FOCS*, pages 199–207. IEEE, 2008.

[39] Radu Berinde and Piotr Indyk. Sequential sparse matching pursuit. In *Allerton*, pages 36–43. IEEE, 2009.

[40] Radu Berinde, Anna C Gilbert, Piotr Indyk, Howard Karloff, and Martin J Strauss. Combining geometry and combinatorics: A unified approach to sparse signal recovery. In *Allerton*, pages 798–805. IEEE, 2008.

[41] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006.

[42] Joseph B Kruskal. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear algebra and its applications*, 18(2): 95–138, 1977.

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

[44] Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006.

[45] Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

[46] Hyunsoo Kim and Haesun Park. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, 23(12): 1495–1502, 2007.

[47] Scott L Pomeroy, Pablo Tamayo, Michelle Gaasenbeek, Lisa M Sturla, Michael Angelo, Margaret E McLaughlin, John YH Kim, Liliana C Goumnerova, Peter M Black, Ching Lau, et al. Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, 415(6870):436, 2002.

[48] Arindam Bhattacharjee, William G Richards, Jane Staunton, Cheng Li, Stefano Monti, Priya Vasa, Christine Ladd, Javad Beheshti, Raphael Bueno, Michael Gillette, et al. Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. *Proceedings of the National Academy of Sciences*, 98(24):13790–13795, 2001.

[49] Ken I Mills, Alexander Kohlmann, P Mickey Williams, Lothar Wieczorek, Wei-min Liu, Rachel Li, Wen Wei, David T Bowen, Helmut Loeffler, Jesus M Hernandez, et al. Microarray-based classifiers and prognosis models identify subgroups with distinct clinical outcomes and high risk of aml transformation of myelodysplastic syndrome. *Blood*, 114(5):1063–1072, 2009.

[50] Ali H Shoeb and John V Guttag. Application of machine learning to epileptic seizure detection. In *ICML*, pages 975–982, 2010.

[51] Vatsal Sharan and Gregory Valiant. Orthogonalized ALS: A theoretically principled tensor decomposition algorithm for practical use. *arXiv preprint arXiv:1703.01804*, 2017.

[52] Animashree Anandkumar, Rong Ge, and Majid Janzamin. Guaranteed non-orthogonal tensor decomposition via alternating Rank-1 updates. *arXiv preprint arXiv:1402.5180*, 2014.

[53] Daniel M Kane and Jelani Nelson. Sparser johnson-lindenstrauss transforms. *JACM*, 61(1):4, 2014.

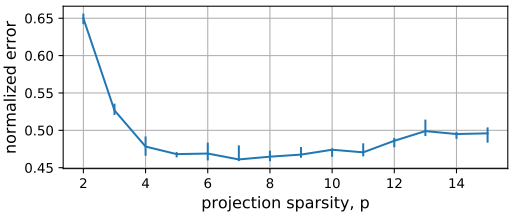

Figure 4: NMF reconstruction error vs. projection matrix column sparsity.

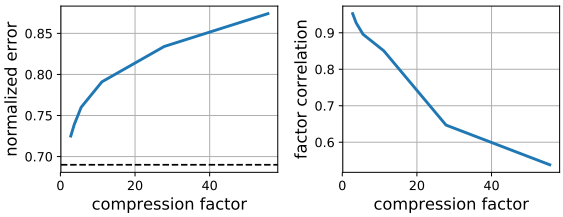

Figure 5: Accuracy of tensor decomposition on compressed EEG data. **Left:** Normalized reconstruction error; dashed line indicates baseline reconstruction error on original data. **Right:** Median Pearson correlations between recovered factors and factors computed from original data.

A Supplementary Experimental Results

A.1 NMF Reconstruction Error and Projection Matrix Column Sparsity

We investigated the trade-off between reconstruction error (as measured by normalized Frobenius loss) and the sparsity parameter p of the binary random projections P. Recall that $P \in \{0,1\}^{d \times n}$ is a randomly sampled sparse binary matrix where p distinct entries in each column are selected uniformly at random and set to 1. In Figure 4, we plot the normalized reconstruction error achieved by NMF using FACTORIZE-RECOVER on the lung carcinoma gene expression dataset [48] at a fixed compression level of 5. Since we observed that the cost of sparse recovery increases roughly linearly with p, we aimed to select a small value of p that achieves good reconstruction accuracy. We found that the setting $p = 5$ was a reasonable choice for our experiments.

A.2 Tensor Decomposition of EEG data

Reconstruction Error. In Figure 5, we show normalized Frobenius errors of the recovered factorization and median Pearson correlations between the columns of the recovered temporal factor and those computed on the original, uncompressed data (paired via maximum bipartite matching). Due to the sparsity of the data, we can achieve over $10\times$ compression at the cost of a 14% increase in reconstruction error. The alternative of recovering before factorizing (RF-TD) achieves slightly lower error at a given projection dimension: at $d = 1000$, RF-TD achieves a normalized error of 0.819 vs. 0.834 when the compressed data is factorized directly. However, RF-TD is three orders of magnitude slower than FACTORIZE-RECOVER on this dataset due to the large number of sparse recovery instances required ($303 \times 23 = 6969$) to fully recover the data tensor.

Preprocessing. Each channel is individually whitened with a mean and standard deviation estimated from segments of data known to not contain any periods of seizure. The spectrogram is computed with a Hann window of size 512 (corresponding to two seconds of data). The window overlap is set to 64. In order to capture characteristic sequences across time windows, we transform the spectrogram by concatenating groups of sequential windows, following Shoeb and Guttag [50]. We concatenate groups of size three.

B Proof of Theorem 1: Uniqueness for NMF

We follow the outline from the proof sketch. Recall that our goal will be to prove that the columns of $\tilde{W}$ are the sparsest vectors in the column space of $\tilde{W}$. For readability proofs of some auxiliary lemmas appears later in Section B.1

As mentioned in the proof sketch, the first step is part (a)—showing that if we take any subset S of the columns of $\tilde{W}$, then the number of rows which have non-zero entries in at least one of the columns in

S is large. Lemma 1 shows that the number of rows which are have at least one zero entry in a subset S of the columns of W columns proportionately with the size of S. The proof proceeds by showing that choosing B such that each column has k randomly chosen non-zero entries ensures expansion for B with high probability, and we have already ensured expansion for P with high probability.

Lemma 1. *For any subset S of the columns of $\tilde{W}$, define $N(S)$ to be the subset of the rows of $\tilde{W}$ which have a non-zero entry in at least one of the columns in S. Then for every subset S of columns of $\tilde{W}$, $|N(S)| \geq \min\{16|S|kp/25, d/200\}$ with failure probability $re^{-\beta k}/n + (1/n^5)$.*

We now prove the second part of the argument—that any linear combination of columns in S cannot have much fewer non-zero entries than $N(S)$, as the probability that many of the non-zero entries get canceled is zero. Lemma 2 is the key to showing this. Define a vector x as *fully dense* if all its entries are non-zero.

Lemma 2. *For any subset S of the columns of $\tilde{W}$, let U be the submatrix of $\tilde{W}$ corresponding to the S columns and $N(S)$ rows. Then with probability one, every subset of the rows of U of size at least $|S|p$ does not have any fully dense vector in its right null space.*

Proof. Without loss of generality, assume that S corresponds to the first $|S|$ columns of $\tilde{W}$, and $N(S)$ corresponds to the first $|N(S)|$ rows of $\tilde{W}$. We will partition the rows of U into t groups $\{\mathcal{G}_1, \ldots, \mathcal{G}_t\}$. Each group will have size at most p. To select the first group, we choose any entry y_1 of Y which appears in the first row of U. For example, if the first column of W has a one in its first row, and $P(1,1) = 1$, then the random variable $Y_{1,1}$ appears in the first row of U. Say we choose $y_1 = Y_{1,1}$. We then choose $\mathcal{G}_1$ to be the set of all rows where y_1 appears. We then remove the set of rows $\mathcal{G}_1$ from U. To select the second group, we pick any one of the remaining rows, and choose any entry y_2 of Y which appears in that row of U. $\mathcal{G}_2$ is the set of all rows where y_2 appears. We repeat this procedure to obtain t groups, each of which will have size at most p as every variable appears in p columns. Hence any subset of rows of size at least $|S|p$ must correspond to at least $|S|$ groups.

Let $\mathcal{N}_j$ be the right null space of the first j groups of rows. We define $\mathcal{N}_0 = \mathbb{R}^{|S|}$. We will now show that either $\text{rank}(\mathcal{N}_i) = |S| - i$ or $\mathcal{N}_i$ does not contain a fully dense vector. We prove this by induction. Consider the jth step, at which we have j groups $\{\mathcal{G}_1, \ldots, \mathcal{G}_j\}$. By the induction hypothesis, either $\mathcal{N}_j$ does not contain any fully dense vector, or $\text{rank}(\mathcal{N}_j) = |S| - j$. If $\mathcal{N}_j$ does not contain any fully dense vector, then we are done as this implies that $\mathcal{N}_{j+1}$ also does not contain any fully dense vector. Assume that $\mathcal{N}_j$ contains a fully dense vector x. Choose any row U_c which has not been already been assigned to one of the sets. By the following elementary proposition, the probability that x is orthogonal to U_c is zero. We provide a simple proof in Section B.1.

Lemma 3. *Let $v = (v_1, \ldots, v_n) \in \mathbb{R}^n$ be a vector of n independent $N(0,1)$ random variables. For any subset $S \subseteq \{1, \ldots, n\}$, let $v(S) \in \mathbb{R}^{|S|}$ refer to the subset of v corresponding to the indices in S. Consider t such subsets $S_1, \ldots, S_t$. Let each set S_i defines some linear relation $\alpha_{S_i}^T v(S_i) = 0$, for some $\alpha_{S_i} \in \mathbb{R}^{|S_i|}$. Assume that the variable v_i appear in the set S_i. Then the probability distribution of the set of variables $\{v_{t+1}, \ldots, v_n\}$ conditioned on the linear relations defined by $S_1, \ldots, S_t$ is still continuous and its density function has full support. In particular, any linear combination of the set of variables $\{v_{t+1}, \ldots, v_n\}$ has zero probability of being zero.*

If $\mathcal{N}_j$ contains a fully dense vector, then with probability one, $\text{rank}(\mathcal{N}_{j+1}) = \text{rank}(\mathcal{N}_j) - 1 = n - j - 1$. This proves the induction argument. Therefore, with probability one, for any $t \geq |S|$, either $\text{rank}(\mathcal{N}_t) = 0$ or $\mathcal{N}_t$ does not contain a fully dense vector and Lemma 2 follows. □

We now complete the proof of Theorem 1. Note that the columns of $\tilde{W}$ have at most kp non-zero entries, as each column of P has p-sparse. Consider any set S of columns of $\tilde{W}$. Consider any linear combination $v \in \mathbb{R}^d$ of the set S columns, such that all the combination weights $x \in \mathbb{R}^{|S|}$ are non-zero. By Lemma 1, $|N(S)| \geq \min\{16|S|kp/25, d/200\}$ with failure probability $re^{-\beta k}/n + (1/n^5)$. We claim that v has more than $|N(S)| - |S|p$ non zero entries. We prove by contradiction. Assume that v has $|N(S)| - |S|$ or fewer non zero entries. Consider the submatrix U of $\tilde{W}$ corresponding to the S columns and $N(S)$ rows. If v has $|N(S)| - |S|p$ or fewer non zero entries, then there must be a subset S' of the $|N(S)|$ rows of U with $|S'| = |S|p$, such that each of the rows in S' has at least one non-zero entry, and the fully dense vector x lies in the right null space of S'. But by Lemma 2, the probability of this happening is zero. Hence v has more than $|N(S)| - |S|p$ non zero entries. Lemma 4 obtains a lower bound on $|N(S)| - |S|p$ using simple algebra.

Lemma 4. *$|N(S)| - |S|p \geq 6kp/5$ for $|S| > 1$ for $d \geq 400p(r+k)$.*

Hence any linear combination of more than one column of $\tilde{W}$ has at least $6kp/5$ non-zero entries with failure probability $re^{-\beta k}/n$. Hence the columns of $\tilde{W}$ are the sparsest vectors in the column space of $\tilde{W}$ with failure probability $re^{-\beta k}/n + (1/n^5)$.

B.1 Additional Proofs for Uniqueness of NMF

Lemma 5. *If H is full row rank, then the column spaces of $\tilde{W}$ and W' are equal.*

Proof. We will first show that the column space of $\tilde{M}$ equals the column space of $\tilde{W}$. Note that the column space of $\tilde{M}$ is a subspace of the column space of $\tilde{W}$. As H is full row rank, the rank of the column space of $\tilde{M}$ equals the rank of the column space of $\tilde{W}$. Therefore, the column space of $\tilde{M}$ equals the column space of $\tilde{W}$.

By the same argument, for any alternative factorization $\tilde{M} = W'H'$, the column space of W' must equal the column space of $\tilde{M}$—which equals the column space of $\tilde{W}$. As the column space of W' equals the column space of $\tilde{W}$, therefore W' must lie in the column space of $\tilde{W}$. □

Lemma 1. *For any subset S of the columns of $\tilde{W}$, define $N(S)$ to be the subset of the rows of $\tilde{W}$ which have a non-zero entry in at least one of the columns in S. Then for every subset S of columns of $\tilde{W}$, $|N(S)| \geq \min\{16|S|kp/25, d/200\}$ with failure probability $re^{-\beta k}/n + (1/n^5)$.*

Proof. We first show a similar property for the columns of W, and will then extend it to the columns of $\tilde{W} = PW$. We claim that for every subset of S columns of W, $|N(S)| \geq \min\{4|S|k/5, n/200\}$ with failure probability $re^{-\beta k}/n$.

To verify, consider a bipartite graph T with r nodes on the left part U_1 corresponding to the r columns of W, and n nodes on the right part V corresponding to the n rows or indices of each factor. The ith node in U_1 has an edge to k nodes in V corresponding to the non-zero indices of the ith column of W. Note that $|N(S)|$ is the neighborhood of the set of nodes S in G. From Part 1 of Lemma 6, the graph G is a $(\gamma_1 r, 4/5)$ expander with failure probability $re^{-\beta k}/n$ for $\gamma_1 = n/(rke^5)$ and a fixed constant $\beta > 0$.

Lemma 6. *Randomly choose a bipartite graph G with n_1 vertices on the left part U and n_2 vertices on the right part V such that every vertex in U has degree D. Then,*

1. *For every $n_1, n_2, n_1 < n_2$, G is a $(\gamma n_1, 4/5)$ expander for $D \geq c$ for some fixed constant c and $\gamma n_1 = \frac{n_2}{De^5}$ except with probability $n_1 e^{-\beta D}/n_2$ for a fixed constant $\beta > 0$.*
2. *For every $n_1, n_2, n_2 < n_1$, G is a $(\gamma n_1, 4/5)$ expander for $D \geq c \log n_1$ for some fixed constant c and $\gamma n_1 = \frac{n_2}{De^5}$ except with probability $(1/n_1)^5$.*

As G is a $(\gamma_1 r, 4/5)$ expander, every set of $|S| \leq \gamma_1 r$ nodes has at least $4|S|k/5$ neighbors. A set of size $|S| > \gamma_1 r$ nodes, must include a subset of size $\gamma_1 r$ which has $4n/(5e^5) \geq n/200$ neighbours, and hence every set of size $|S| > \gamma_1 r$ has at least $n/200$ neighbors. Therefore, for every subset of S columns, $|N(S)| \geq \min\{4|S|k/5, n/200\}$ with failure probability $re^{-\beta k}/n$.

We will now extend the proof to show the necessary property for $\tilde{W}$. After the projection step, the n indices are projected to d dimensions, and the projection matrix is a $(\gamma_2 n, 4/5)$ expander with $\gamma_2 = d/(nke^5)$. We can now consider a tripartite graph, by adding a third set U_2 with d nodes. We add an edge from a node i in V to node j in U_2 if $P(j,i) = 1$. For any subset S of columns of $\tilde{W}$, $N(S)$ are the set of nodes in U_2 which are reachable from the nodes S in U_1.

With failure probability $(1/n^5)$, the projection matrix P is a $(\gamma_2 n, 4/5)$ expander with $\gamma_2 = d/(npe^5)$. Therefore every subset of size t in V has at least $\min\{4tp/5, d/200\}$ neighbors in W. By combining this argument with the fact that every set of S nodes in U, has at least $\min\{4|S|k/5, n/200\}$ neighbors with failure probability $re^{-\beta k}/n$, it follows that for every subset of S columns of $\tilde{W}$, $|N(S)| \geq \min\{16|S|kp/25, d/200\}$ with failure probability $re^{-\beta k}/n + (1/n^5)$. □

Lemma 3. *Let $v = (v_1, \ldots, v_n) \in \mathbb{R}^n$ be a vector of n independent $N(0,1)$ random variables. For any subset $S \subseteq \{1, \ldots, n\}$, let $v(S) \in \mathbb{R}^{|S|}$ refer to the subset of v corresponding to the indices in S. Consider t such subsets $S_1, \ldots, S_t$. Let each set S_i defines some linear relation $\alpha_{S_i}^T v(S_i) = 0$, for some $\alpha_{S_i} \in \mathbb{R}^{|S_i|}$. Assume that the variable v_i appear in the set S_i. Then the probability distribution of the set of variables $\{v_{t+1}, \ldots, v_n\}$ conditioned on the linear relations defined by $S_1, \ldots, S_t$ is still continuous and its density function has full support. In particular, any linear combination of the set of variables $\{v_{t+1}, \ldots, v_n\}$ has zero probability of being zero.*

Proof. We prove by induction. For the base case, note that without any linear constraints, the set of n random variables $\{v_1, \cdots, v_n\}$ is continuous and has full support as the random variables v_i are independent Gaussian. Consider the jth step, when linear constraints defined by the sets $S_1, \cdots, S_j$ have been imposed on the variables. We claim that the distribution of the set of random variables $\{v_{j+1}, \cdots, v_n\}$ is continuous and has full support after imposition of the constraints $S_1, \cdots, S_j$. By the induction hypothesis, the distribution of the set of random variables $\{v_j, \cdots, v_n\}$ is continuous and has full support after imposition of the constraints $S_1, \cdots, S_{j-1}$. Note that the linear constraint S_j can be satisfied for any assignment to the subset of variables $\{v_{j+1}, \cdots, v_n\}$ which appear in the constraint S_j, as v_j can be chosen appropriately because by the induction hypothesis it has full support conditioned on the previous constraints $S_1, \cdots, S_{j-1}$. Hence the probability distribution of the set of variables $\{v_{j+1}, \cdots, v_n\}$ is still continuous and has full support after adding the constraint S_j. □

Lemma 4. $|N(S)| - |S|p \geq 6kp/5$ *for* $|S| > 1$ *for* $d \geq 400p(r+k)$.

Proof. For $2 \leq |S| \leq d/(128kp)$,

$$\begin{aligned} |N(S)| - |S|p &\geq (16kp/25)|S| - p|S| = 30kp/25 + kp(16|S| - 30)/25 - p|S| \\ &\geq 6kp/5 + p\Big(k(16|S| - 30) - |S|\Big) \end{aligned}$$

For $|S| \geq 2$ and $k \geq 2$, $k(16|S| - 30) - |S| \geq 0$, hence $|N(S)| - |S|p \geq 6kp/5$ for $2 \leq |S| \leq d/(128kp)$. For $|S| > d/(128kp)$, $|N(S)| \geq d/200$. Therefore, $|N(S)| - |S|p \geq d/200 - rp \geq 2kp$ for $d \geq 400p(r+k)$. □

Lemma 6. *Randomly choose a bipartite graph G with n_1 vertices on the left part U and n_2 vertices on the right part V such that every vertex in U has degree D. Then,*

1. *For every $n_1, n_2, n_1 < n_2$, G is a $(\gamma n_1, 4/5)$ expander for $D \geq c$ for some fixed constant c and $\gamma n_1 = \frac{n_2}{De^5}$ except with probability $n_1 e^{-\beta D}/n_2$ for a fixed constant $\beta > 0$.*

2. *For every $n_1, n_2, n_2 < n_1$, G is a $(\gamma n_1, 4/5)$ expander for $D \geq c \log n_1$ for some fixed constant c and $\gamma n_1 = \frac{n_2}{De^5}$ except with probability $(1/n_1)^5$.*

Proof. Consider any subset $S \subset U$ with $|S| \leq \gamma n_1$. Let $\mathbb{P}(N(S) \subseteq M)$ denote the probability of the event that the neighborhood of S is entirely contained in $M \subset V$. $\mathbb{P}(N(S) \subseteq M) \leq \left(\frac{|M|}{n_2}\right)^{D|S|}$. We will upper bound the probability of G not being an expander by upper-bounding the probability of each subset $S \subset U$ with $|S| \leq \gamma n_1$ not expanding. Let $\mathbb{P}(S)$ denote the probability of the neighborhood of S being entirely contained in a subset $M \subset V$ with $M < \alpha|S|D$. By a union

bound,

$$\begin{aligned}
\mathbb{P}(G \text{ is not a } (\gamma n_1, \alpha) \text{ expander}) &\leq \sum_{\substack{S \subset U \\ |S| \leq \gamma n_1}} \mathbb{P}(\bar{S}) \\
&\leq \sum_{\substack{S \subset U \\ |S| \leq \gamma n_1}} \sum_{\substack{M \subset V \\ M = \alpha |S| D}} \mathbb{P}(N(S) \subseteq M) \\
&\leq \sum_{s=1}^{\gamma n_1} \sum_{\substack{S \subset U \\ |S| = s}} \sum_{\substack{M \subset V \\ M = \alpha |S| D}} \Big(\frac{\alpha |S| D}{n_2}\Big)^{D|S|} \\
&\leq \sum_{s=1}^{\gamma n_1} \binom{n_1}{s} \binom{n_2}{\alpha D s} \Big(\frac{\alpha D s}{n_2}\Big)^{Ds}
\end{aligned}$$

Using the bound $\binom{n}{k} \leq (ne/k)^k$, we can write,

$$\begin{aligned}
\mathbb{P}(G \text{ is not a } (\gamma n_1, \alpha) \text{ expander}) &\leq \sum_{s=1}^{\gamma n_1} \Big(\frac{n_1 e}{s}\Big)^{\alpha s} \Big(\frac{n_2 e}{\alpha D s}\Big)^{\alpha D s} \Big(\frac{\alpha D s}{n_2}\Big)^{Ds} \\
&\leq \sum_{s=1}^{\gamma n_1} \left[\Big(\frac{n_1 e}{s}\Big)^{\alpha} \Big(\frac{n_2 e}{\alpha D s}\Big)^{\alpha D} \Big(\frac{\alpha D s}{n_2}\Big)^{D} \right]^s \\
&\leq \sum_{s=1}^{\gamma n_1} x_s^s
\end{aligned}$$

where $x_s = \Big(\frac{n_1 e}{s}\Big) \Big(\frac{n_2 e}{\alpha D s}\Big)^{\alpha D} \Big(\frac{\alpha D s}{n_2}\Big)^{D}$. x_s can be bounded as follows-

$$\begin{aligned}
x_s &= \Big(\frac{n_1 e}{s}\Big) \Big(\frac{\alpha D s e^{1/(1-\alpha)}}{n_2}\Big)^{(1-\alpha)D} \\
&\leq \Big(\frac{e}{\gamma}\Big) \Big(\frac{\alpha D \gamma n_1 e^{1/(1-\alpha)}}{n_2}\Big)^{(1-\alpha)D} \\
&\leq \Big(\frac{n_1 e^{1+1/(1-\alpha)}}{n_2}\Big) D \alpha^{(1-\alpha)D} \\
&\leq \Big(\frac{n_1 e^6}{n_2}\Big) D e^{-D/25} = x
\end{aligned}$$

where in the last step we set $\alpha = 4/5$. Hence we can upper bound the probability of G not being an expander as follows—

$$\mathbb{P}(G \text{ is not a } (\gamma n_1, \alpha) \text{ expander}) \leq \sum_{s=1}^{\infty} x^s \leq \frac{x}{1-x}$$

The two parts of Lemma 6 follow by plugging in the respective values for n_1, n_2 and D. □

C Guaranteed Recovery for Tensors in the Projected Space

We can prove a stronger result for symmetric, incoherent tensors and guarantee accurate recovery in the compressed space using the tensor power method. The tensor power method is the tensor analog of the matrix power method for finding eigenvectors. It is equivalent to finding a rank 1 factorization using the Alternating Least Squares (ALS) algorithm. Incoherent tensors are tensors for which the factors have small inner products with other. We define the incoherence $\mu = \max_{i \neq j} \{|A_i^T A_j|\}$. Our guarantees for tensor decomposition follow from the analysis of the tensor power method by Sharan and Valiant [51]. Proposition 2 shows guarantees for recovering one of the true factors, multiple random initializations can then be used for the tensor power method to recover back all the factors (see Anandkumar et al. [52]).

Proposition 2. *Consider a n-dimensional rank r tensor $T = \sum_{i=1}^{r} w_i A_i \otimes A_i \otimes A_i$. Let $c_{\max} = \max_{i \neq j} |A_i^T A_j|$ be the incoherence between the true factors and $\gamma = \frac{w_{\max}}{w_{\min}}$ be the ratio of the largest and smallest weight. Assume γ is a constant and $\mu \leq o(r^{-2})$. Consider a projection matrix $P \in \{0, \pm 1\}^{n \times d}$ where every row has exactly p non-zero entries, chosen uniformly and independently at random and the non-zero entries have uniformly and independently distributed signs. We take $d = O(r^4 \log r)$ and $p = O(r^2 \log r)$. Let $\tilde{A} = AP$ and $\tilde{T}$ be the d dimensional projection of T, hence $\tilde{T} = \sum_{i=1}^{k} w_i \tilde{A}_i \otimes \tilde{A}_i \otimes \tilde{A}_i$. Then for the projected tensor decomposition problem, if the initialization $x_0 \in \mathbb{R}^d$ is chosen uniformly at random from the unit sphere, then with high probability the tensor power method converges to one of the true factors of $\tilde{T}$ (say the first factor $\tilde{A}_1$) in $O(r(\log r + \log \log d))$ steps, and the estimate $\tilde{A}'$ satisfies $\|\tilde{A}_1 - \tilde{A}_1'\|_2^2 \leq O(r \max\{\mu^2, 1/d^2\})$.*

Proof. Our proof relies on Theorem 3 of Sharan and Valiant [51] and sparse Johnson Lindenstrauss transforms due to Kane and Nelson [53]. To show Claim 2 we need to ensure that the incoherence parameter in the projected space is small. We use the Johnson Lindenstrauss property of our projection matrix to ensure this. A matrix M is regarded as a Johnson Lindenstrauss matrix if it preserves the norm of a randomly chosen unit vector x up to a factor of $(1 \pm \epsilon)$, with failure probabilty δ:

$$\mathbb{P}_x[(1 - \epsilon) < \|Mx\|_2^2 < (1 + \epsilon)] > 1 - \delta.$$

We use the results of Kane and Nelson [53] who show that with high probability a matrix $P \in \{0, \pm 1\}^{n \times d}$ where every row has p non-zero entries, chosen uniformly and independently at random and the non-zero entries have uniformly and independently distributed signs, preserves pairwise distances to within a factor ϵ for $d = O(\epsilon^{-2} \log(1/\delta))$ and $p = \Theta(\epsilon^{-1} \log(1/\delta))$.

It is easy to verify that inner-products are preserved to within an additive error ϵ if the pairwise distances are preserved to within a factors of $(1 \pm \epsilon)$. By choosing $\delta = 1/r^3$ and doing a union bound over all the r^2 pairs of factors, the factors are $(\mu \pm \epsilon)$ incoherent in the projected space with high probability if they were μ incoherent in the original space. Setting $\epsilon = r^{-2} \log^{-1} r$ ensures that $\mu + \epsilon = o(r^{-2})$. Claim 2 now again follows from Theorem 3 of Sharan and Valiant [51]. □