

LOW-RANK APPROXIMATION AND COMPLETION OF POSITIVE TENSORS*

ANIL ASWANI†

Abstract. Unlike the matrix case, computing low-rank approximations of tensors is NP-hard and numerically ill-posed in general. Even the best rank-1 approximation of a tensor is NP-hard. In this paper, we use convex optimization to develop polynomial-time algorithms for low-rank approximation and completion of positive tensors. Our approach is to use algebraic topology to define a new (numerically well-posed) decomposition for positive tensors, which we show is equivalent to the standard tensor decomposition in important cases. Though computing this decomposition is a nonconvex optimization problem, we prove it can be exactly reformulated as a convex optimization problem. This allows us to construct polynomial-time randomized algorithms for computing this decomposition and for solving low-rank tensor approximation problems. Among the consequences is that best rank-1 approximations of positive tensors can be computed in polynomial time. Our framework is next extended to the tensor completion problem, where noisy entries of a tensor are observed and then used to estimate missing entries. We provide a polynomial-time algorithm that for specific cases requires a polynomial (in tensor order) number of measurements, in contrast to existing approaches that require an exponential number of measurements. These algorithms are extended to exploit sparsity in the tensor to reduce the number of measurements needed. We conclude by providing a novel interpretation of statistical regression problems with categorical variables as tensor completion problems, and numerical examples with synthetic data and data from a bioengineered metabolic network show the improved performance of our approach on this problem.

Key words. tensor completion, tensor approximation, categorical regression

AMS subject classifications. 90C25, 62F12, 05E45, 60B20

1. Introduction. Tensors generalize matrices by describing a multidimensional array of numbers. More formally, a tensor ψ of order p is given by $\psi \in \mathbb{R}^{r_1 \times \dots \times r_p}$, where r_i is the *dimension* of the tensor in the i -th index, for $i = 1, \dots, p$. When we would like to refer to a specific entry in the tensor, we use the notation $\psi_x := \psi_{x_1, \dots, x_p}$, where $x = (x_1, \dots, x_p)$, $x_i \in [r_i]$ denotes the value of the i -th index, and $[s] := \{1, \dots, s\}$. Also let $r = \max_i r_i$. The reasons for choosing this notation will become more clear when discussing our novel interpretation of statistical regression with categorical variables as tensor completion.

The similarity between tensors and matrices is misleading because many problems that are routine and polynomial-time computable for matrices are NP-hard for tensors. For instance, it is NP-hard to compute the rank of a tensor [27], which is defined as the minimal number of rank-1 components needed to represent the tensor: $\text{rank}_{\otimes}(\psi) = \min\{q \mid \psi = \sum_{j=1}^q v_1^j \otimes \dots \otimes v_p^j, \text{ where } v_i^j \in \mathbb{R}^{r_i}\}$, where \otimes is the *tensor product* [33, 27]. Tensor analogs of the matrix singular value decomposition (e.g., CANDECOMP/PARAFAC or CP) are also NP-hard to compute [27]. Furthermore, determining the best low-rank approximations for tensors is an ill-posed problem in general [16], and computing the best rank-1 approximation is NP-hard in general [27].

In this paper, we attack the computational challenges posed by tensor problems by showing that positive tensors are amenable to polynomial-time algorithms with strong guarantees. A new tensor decomposition called a hierarchical decomposition is defined in §2 using a structure from algebraic topology. This decomposition is shown to exist, be numerically well-posed, and coincide with the usual tensor CP

*This work was supported in part by NSF grant CMMI-1450963.

†Industrial Engineering and Operations Research, University of California, Berkeley, CA 94720 (aaswani@berkeley.edu).

decomposition [31, 27] in specific cases. Section 3 develops a randomized algorithm to compute the hierarchical decomposition in polynomial time that depends on the degrees-of-freedom of the tensor rather than on the total number of tensor entries, which can be exponentially larger than the degrees-of-freedom. This algorithm can compute a best rank-1 approximation of positive tensors in polynomial time.

Our approach differs from existing methods in a number of ways. An iterative descent algorithm was proposed for decomposition of positive tensors in [58], but no convergence guarantees were provided. Matrix nuclear norm approaches are popular for tensor completion [56, 55, 22, 37, 42, 62], though these approaches have exponentially slow statistical convergence. Using the tensor nuclear norm (which is NP-hard to compute [21]) also gives exponentially slow statistical convergence [60, 61]. For orthogonal third-order tensors (i.e., $p = 3$), alternating minimization provides polynomial statistical convergence [28]; unfortunately, this guarantee requires using a large number of randomized initializations, and so the results cannot be naturally generalized to higher order tensors without losing polynomial-time computability. Recently, Lasserre hierarchy approaches have been proposed for best rank-1 approximations [46] and tensor completion [50]. These have found success on specific numerical examples, but conditions to guarantee global optimality are currently unavailable.

After presenting our algorithm for computing a hierarchical decomposition, §4 extends this framework to the problem of tensor completion [56, 55, 22, 37, 42, 62, 40], in which a subset of tensors entries are observed and then used to estimate missing entries. We provide an algorithm that for specific cases requires a polynomial (in tensor order) number of measurements, which is much lower than the exponential number of measurements required by tensor completion methods using the matrix nuclear norm [56, 55, 22, 37, 42, 62]. In the case of a rank-1 tensor, the number of needed measurements of our approach $O((rp^2)^{1+\zeta})$, for any $\zeta > 0$, is essentially a quadratic factor away from the information-theoretic lower bound of $O(rp)$. Section 5 shows how the algorithms can be improved to exploit sparsity in the tensor. Numerical examples with synthetic data in §6 show our approach outperforms other tensor completion algorithms. We conclude by providing in §7 a novel interpretation of statistical regression problems with categorical variables as tensor completion problems. Data from a bioengineered metabolic network is used to show the improved performance of our approach for categorical regression.

2. Hierarchical Decomposition of Positive Tensors. A structure from algebraic topology [17, 18] is used to parametrize our new decomposition. Following the definition of [18]: A *simplicial complex* is a set $\Gamma \subseteq 2^{[p]}$ such that $F \in \Gamma$ and $S \subset F$ implies that $S \in \Gamma$. The elements of Γ are called *faces* of Γ and the inclusion-maximal faces are the *facets* of Γ . We will assume the facets have been arbitrarily assigned an order, so that we can represent the simplicial complex as $\text{facets}(\Gamma) = \{F_1, F_2, \dots, F_{m(\Gamma)}\}$, where $m(\Gamma)$ is the number of facets. We will drop the argument notation (Γ) when clear from the context. Roughly speaking, a simplicial complex is a graph with higher-order connections between vertices. Whereas edges in a graph can only connect two vertices, facets in a simplicial complex can simultaneously connect an arbitrary number of vertices. An example of a simplicial complex with zero-, one-, two-, and three-dimensional facets is shown in Figure 1.

This section begins with the definition of a hierarchical decomposition for positive tensors, and this decomposition is parametrized by a simplicial complex. Hierarchical decompositions are shown to always exist and be numerically well-posed for positive tensors. Next, we define an important special case of the hierarchical decomposition,

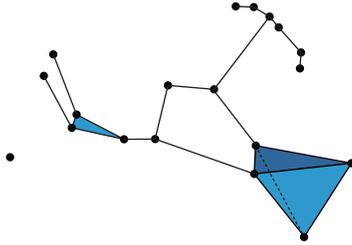


FIG. 1. Example of a Simplicial Complex

which we call a partition decomposition. This is used to provide instances in which these decompositions exactly coincide with the typical tensor CP decomposition. The following notation (adapted from [18]) will be needed to write subindices. Recall the set notation $[s] := \{1, \dots, s\}$, and define $\mathcal{R} = [r_1] \times \dots \times [r_p]$. If $x = (x_1, \dots, x_p) \in \mathcal{R}$ and $F = \{f_1, f_2, \dots\} \subseteq [p]$, then $\mathcal{X}_F = (x_{f_1}, x_{f_2}, \dots)$, and this vector has the state space $\mathcal{R}_F = [r_{f_1}] \times [r_{f_2}] \times \dots$. We use the notation $\mathcal{X}_k = \mathcal{X}_{F_k}$ and $\mathcal{R}_k = \mathcal{R}_{F_k}$ to reduce the number of indices in our equations.

2.1. Definition. Motivated by hierarchical log-linear models used in statistics to construct hypothesis tests for contingency tables [18], we define a *hierarchical decomposition* of a positive tensor to be

$$(2.1) \quad \psi_x = \prod_{k=1}^m \theta_{\mathcal{X}_k}^{(k)}$$

where Γ is a simplicial complex with facets $\Gamma = \{F_1, \dots, F_m\}$, and $\theta^{(k)} \in \mathbb{R}^{r_{f_1} \times r_{f_2} \times \dots}$ are constants indexed by the different values of $\mathcal{X}_k \in \mathcal{R}_k$. When Γ is such that (2.1) is satisfied, we say Γ is *correct*; on the other hand, if Γ is such that (2.1) does not hold, then we say Γ is *incorrect*. To simplify notation, we drop the superscript in $\theta_{\mathcal{X}_k}^{(k)}$ and write this as $\theta_{\mathcal{X}_k}$ when clear from the context. Also, $\Theta = \{\theta^{(k)} : k = 1, \dots, m\}$ refers to the set of all parameters.

2.2. Existence and Representational Complexity. Existence (and well-posedness) of the hierarchical decomposition of a positive tensor can be shown under a mild boundedness assumption:

A1. The tensor is bounded $M^{-1} \leq \psi_x \leq M$ by some constant $M > 1$.

Our results generalize to the case $M_1 \leq \psi_x \leq M_2$, where $0 < M_1 < M_2$; we keep the above assumption to simplify stating the results. Relaxing the lower bound to zero is more delicate: In practice, we can choose M sufficiently large such that the lower bound is arbitrarily close to zero. In theory, relaxing the lower bound to exactly zero requires additional analysis because the loss function we will use, though continuously differentiable, does not have a bounded derivative at zero.

PROPOSITION 2.1. *If ψ satisfies **A1**, then a hierarchical decomposition of ψ with a correct Γ exists.*

Proof. The result follows by choosing a simplicial complex: facets $\Gamma = \{F_1\}$, where $F_1 = \{1, \dots, p\}$, and then setting $\theta_{\mathcal{X}_1} := \psi_x$. \square

Note there is a lack of uniqueness of the parametrizing Γ because we can always choose a simplicial complex with a single facet, as in the above proof, to specify

a valid hierarchical decomposition. Because of this nonuniqueness, it is useful to define a notation of complexity. We define the *effective dimension* of a hierarchical decomposition for a specific choice of Γ to be $\rho(\Gamma) = \sum_{k=1}^m \prod_{j \in F_k} r_j$. The effective dimension is the number of coefficients used in the hierarchical decomposition of the tensor. In many cases, a tensor of low rank can be represented by a hierarchical decomposition with low effective dimension. Specific examples are given in the next subsection. Moreover, a counting argument implies that the tensor rank must be upper bounded by the effective dimension: $\text{rank}_{\otimes}(\psi) \leq \rho$. It is for these reasons we use low effective dimension as a surrogate for low tensor rank when we study the problems of tensor approximation and completion.

2.3. Numerical Well-Posedness. Beyond existence, hierarchical decompositions are also well-posed. One of the reasons that computing the best low-rank approximation of a tensor is an ill-posed problem in general [16] is that though the entries of the tensor might be bounded, the coefficients of the tensor decomposition can be unbounded. (This can occur because the unbounded nature of the coefficients cancel each other out.) This leads to unique phenomenon such as having a sequence of tensors of rank two that converge to a tensor of rank three [16, 33]. Fortunately, the situation for nonnegative tensors is better because the approximation problem is well-posed [36, 49]. As we show with the next proposition, the hierarchical decomposition is also well-posed in a particular way that will be important for formulating optimization problems.

PROPOSITION 2.2. *If ψ satisfies **A1** and Γ is correct, then there exists Θ such that $M^{-2} \leq \theta_{\mathcal{X}_k} \leq M^2$, for all $\mathcal{X}_k \in \mathcal{R}_k$ and $k = 1, \dots, m$.*

Proof. We successively construct a set of parameters and show these satisfy the proposition. One set of parameters can be defined by performing the following steps:

1. Set $\mathcal{I} = \mathcal{R}$;
2. For $j = 1, \dots, m$:
 - (a) Set $\mathcal{H}_j = \{\mathcal{X}_j \in \mathcal{R}_j : x \in \mathcal{I}\}$;
 - (b) While $\mathcal{H}_j \neq \emptyset$:
 - i. Select an arbitrary element $u \in \mathcal{I}$;
 - ii. Set $\theta_{\mathcal{U}_k} = 1$ for all $k = (j+1), \dots, m$;
 - iii. Set $\theta_{\mathcal{X}_j} = \psi_x / \prod_{k=1}^{j-1} \theta_{\mathcal{X}_k}$, for all $x \in \mathcal{I}$ such that $\mathcal{X}_k = \mathcal{U}_k, \forall k = (j+1), \dots, m$;
 - iv. Set $\mathcal{I} = \mathcal{I} \setminus \{x \in \mathcal{I} : \mathcal{X}_k = \mathcal{U}_k, \forall k = (j+1), \dots, m\}$;
 - v. Set $\mathcal{H}_j = \mathcal{H}_j \setminus \{\mathcal{X}_j \in \mathcal{H}_j : x \in \mathcal{I} \text{ such that } \mathcal{X}_k = \mathcal{U}_k, \forall k = (j+1), \dots, m\}$.

Observe \mathcal{I} lists the subset of indices of \mathcal{R} for which the decomposition is undefined, and \mathcal{H}_j lists the subset of indices of \mathcal{R}_j for which the decomposition is undefined. The intuition behind this algorithm is we successively specify the parameters of the decomposition until there are no indices for which the decomposition is undefined. The inner loop ensures \mathcal{H}_j becomes empty, and the set \mathcal{I} becomes empty at the end of the algorithm because $\mathcal{I} \equiv \{v \in \mathcal{I} : \mathcal{V}_k = \mathcal{U}_k, \forall k = (j+1), \dots, m\}$ when $j = m$.

Next, note that the parameters $\theta_{\mathcal{U}_k}$ trivially satisfy $M^{-2} \leq \theta_{\mathcal{U}_k} \leq M^2$ since $\theta_{\mathcal{U}_k} = 1$, and so we only need to show that the remaining parameters satisfy the bounds of the proposition. For any $j > 1$, suppose that $M^{-1} \leq \prod_{k=1}^{j-1} \theta_{\mathcal{X}_k} \leq M$. If this condition holds, then two consequences follow from step 2.b.iii: (i) $M^{-2} \leq \theta_{\mathcal{X}_j} \leq M^2$, and (ii) $M^{-1} \leq \prod_{k=1}^j \theta_{\mathcal{X}_k} \leq M$. In fact, for $j = 1$ we have that $M^{-1} \leq \theta_{\mathcal{X}_1} \leq M$, since $\theta_{\mathcal{X}_1} = \psi_x / \prod_{k=1}^0 \theta_{\mathcal{X}_k} = \psi_x$. This inductively shows that the bounds of the

proposition hold for all the remaining parameters. \square

This result implies that the parameters Θ of the decomposition are bounded by an amount that is independent of p and Γ as long as the individual entries of the tensor are bounded as in **A1**. This will allow us to define constraints in our optimization problems that ensure the numerical scaling of different parameters is controlled. For numerical reasons, we would like to avoid scalings in which some parameters are very large and other parameters are very small. This proposition allows us to define constraints that control the scaling.

2.4. Partition Decomposition. An important special case of a hierarchical decomposition is when the facets of the simplicial complex Γ are a partition of the set $[p]$. We refer to this instance as a partition decomposition. The partition decomposition can be written as $\psi_x = \prod_{k=1}^m \theta_{\mathcal{X}_k} = (\theta^{(1)} \otimes \cdots \otimes \theta^{(m)})_{P(x)}$, where the middle equation is the partition decomposition, \otimes is the tensor product, and $P(\cdot)$ is an appropriately-defined permutation of the indices. The partition decomposition is of note because it can be written as the product of tensors with smaller order than ψ , and because it exactly coincides in specific cases with a low-rank CP decomposition of tensors. The CP decomposition is defined as $\psi = \sum_{j=1}^q v_1^j \otimes \cdots \otimes v_p^j$, where $v_i^j \in \mathbb{R}^{r_i}$ and $q = \text{rank}_{\otimes}(\psi)$ is the tensor rank [31], and it is a typical tensor decomposition and an analog of the matrix singular value decomposition [31, 27].

The simplest case in which the partition decomposition coincides with the CP decomposition is when the partition is given by facets $(\Gamma) = \{\{1\}, \dots, \{p\}\}$. In this case, both the partition and CP decompositions represent a rank-1 tensor: $\psi_x = \prod_{k=1}^p \theta_{x_k} = (\theta^{(1)} \otimes \cdots \otimes \theta^{(p)})_x$, where the $\theta^{(k)} \in \mathbb{R}^{r_k}$ are vectors, the middle equation is the partition decomposition, and the right equation is the CP decomposition. The decompositions coincide in this case because they are equivalent.

Another instance where the partition and CP decompositions coincide is when the $\theta^{(k)}$ are either vectors or matrices of full rank. Assume the partitions are arranged so $\theta^{(1)}, \dots, \theta^{(s)}$ are matrices and $\theta^{(s+1)}, \dots, \theta^{(m)}$ are vectors. Also, let a matrix decomposition of $\theta^{(k)}$ be given by $\theta^{(k)} = \sum_{j=1}^{q_k} u_k^j \otimes v_k^j$, where q_k is the matrix rank of $\theta^{(k)}$, and u_k^j, v_k^j are vectors of appropriate dimensions. Then we have $\psi_x = \prod_{k=1}^m \theta_{\mathcal{X}_k} = \left(\sum_{j_1 \times \cdots \times j_s \in [q_1] \times \cdots \times [q_s]} u_1^{j_1} \otimes v_1^{j_1} \otimes \cdots \otimes u_s^{j_s} \otimes v_s^{j_s} \otimes \theta^{(s+1)} \otimes \cdots \otimes \theta^{(m)} \right)_{P(x)}$, where the middle and right equations are the partition and CP decompositions, respectively. The decompositions coincide in this case because the partition decomposition can be used to compute the CP decomposition by computing the matrix singular value decomposition of $\theta^{(k)}$; similarly, the CP decomposition can be used to compute the partition decomposition by computing $\theta^{(k)} = \sum_{j=1}^{q_k} u_k^j \otimes v_k^j$.

3. Randomized Algorithm for Decompositions and Approximations.

The algorithm in Proposition 2.2 implies a hierarchical decomposition can be computed in steps that are polynomial in the number of tensor entries. However, this computational complexity can be improved with a randomized algorithm that will only need a polynomial in effective dimension ρ number of arithmetic calculations. This can be a significant improvement because the effective dimension can be much smaller than the number of tensor entries: For instance, a rank-1 tensor has effective dimension $\rho = \sum r_i = O(rp)$ while it has $\prod r_i = O(r^p)$ entries.

Our approach to developing a randomized algorithm for computing a hierarchical decomposition is to randomly sample entries of the tensor. With enough samples, the decomposition will have low error with high probability. In anticipation of generalizing to the tensor completion problem, we allow the sampled entries to be measured with

noise. This noise could be deterministically interpreted as the approximation error of a hierarchical decomposition, meaning the hierarchical decomposition for a specified Γ that is closest (as measured by some loss function) to the tensor ψ . As a result, the statistical consequences have deterministic interpretations.

This section begins by describing the noise and measurement model for sampling entries of the tensor, and then attention turns towards choosing the loss function that will be used to measure the discrepancy between the computed decomposition and the sampled entries. Specific computational and statistical challenges with choosing the loss function are discussed, and this precludes the use of a squared loss function or of taking the logarithm of the data. We propose an alternative loss function: This loss has the same minimizer in specific cases as that of the squared loss function, and we show it is majorized and minorized by the squared loss function. Furthermore, we show this loss function can be minimized in polynomial time by exactly reformulating the optimization problem as a convex program.

Next, we use this reformulation to show an equivalence result between our loss function and the decomposition error as measured by the squared loss. This equivalence result allows us to study approximation properties using our loss function and then apply the approximation properties to the squared loss. We use the stochastic processes theory of Rademacher complexity [5, 29, 32, 7] to bound the approximation error induced by computing a decomposition using a sample of tensor entries (rather than using all the tensor entries). And the section concludes by presenting a randomized algorithm, which uses the alternative loss function, and proving it has polynomial-time complexity in terms of effective dimension ρ .

3.1. Noise and Measurement Model. Note we use the indexing notation $\langle i \rangle$ to denote the i -th measurement. For a randomly chosen set of indices $x\langle i \rangle \in \mathcal{R}$, suppose we make a noisy measurement of the corresponding tensor entry $y\langle i \rangle = (1 + z\langle i \rangle) \cdot \psi_{x\langle i \rangle}$, where $z\langle i \rangle$ is noise. A multiplicative noise model, as opposed to an additive noise model, is used here because this allows us to define a statistical model where measurements $y\langle i \rangle$ are positive-valued while the noise is independent of $x\langle i \rangle$. However, our results also apply to the case of additive zero-mean noise with the only changes being in the constants of the resulting bounds. Rather than complicating the presentation, we focus on the multiplicative noise model. We make the following assumption about the noise:

A2. The noise $z\langle i \rangle$ are iid random variables with a mean of zero $\mathbb{E}(z) = 0$, and they are bounded $\mu^{-1} \leq 1 + z \leq \mu$ by some constant $\mu > 1$.

The bounds on noise could be relaxed to be unbounded in both directions (i.e., positive and negative). This is appealing because many interesting noise distributions satisfying the property $\mathbb{E}(z) = 0$ are sub-gamma distributions [7]. We do not consider these cases because their consideration does not provide additional theoretical insights; the main difference is slower rates of convergence for heavier-tailed distributions. And so for simplicity, we assume the above boundedness condition; however, we will use the gamma distribution (which is unbounded) to generate noise for the synthetic data in our numerical examples.

Another note is the reason for choosing a model with $\mathbb{E}(z) = 0$ is so $\mathbb{E}[y|x] = \psi_x$ holds. This is a mild assumption because we are interested in computing a decomposition that best approximates ψ_x ; the Θ themselves do not have any particular meaning in our decomposition because they are nonunique up to a scaling factor.

We also make an assumption about the measurements available. For now, we will not impose any conditions on the distribution, except for requiring iid measurements.

A3. The data are iid measurements $(x^{(i)}, y^{(i)})$, for $i = 1, \dots, n$, where n is the number of measurements.

3.2. Challenges with Choosing Loss Function. The usual loss function is the squared loss $L(\Theta) = \mathbb{E}((y - \prod_{k=1}^m \theta_{\mathcal{X}_k})^2)$, and when Γ is correct the minimizer is given by Θ^* such that $\psi_x = \prod_{k=1}^m \theta_{\mathcal{X}_k}^*$ [4]. But numerically minimizing this loss is difficult because of nonconvexity of the squared loss in the parameters Θ . One common approach is to use a heuristic such as alternating least squares (ALS), but this only converges to local optimum [31].

Given the structure of the hierarchical decomposition, it is tempting to compute the decomposition by minimizing $\mathbb{E}((\log y - \sum_{k=1}^m \log \theta_{\mathcal{X}_k})^2)$, because this converts the optimization into a linear least squares problem. However, this is a problematic choice because the approach in [4] can be used to show that the minimizer of the above loss function is $\mathbb{E}[\log y|x] = \sum_{k=1}^m \log \theta_{\mathcal{X}_k} + \mathbb{E}(\log(1+z))$. This is nonideal because the solution will be incorrect by the amount $\mathbb{E}(\log(1+z)) \neq 0$. Jensen's inequality for concave functions implies $\mathbb{E}(\log(1+z)) \leq \log \mathbb{E}(1+z) = \log 1 = 0$; so the general case is the nuisance parameter $\mathbb{E}(\log(1+z))$ will be nonpositive. Taking the exponent $\exp(\mathbb{E}[\log y|x])$ does not resolve the problem because we still have a multiplicative error of $\exp(\mathbb{E}(\log(1+z))) \neq 1$.

3.3. Alternative Loss Function. So if we do not *a priori* know the value of the nuisance parameter $\mathbb{E}(\log(1+z))$, then we could devise a two step procedure that consistently estimates this nuisance parameter and then removes it from the least squares solution, in order to compute a best hierarchical decomposition of ψ_x . We can eliminate the need for considering this nuisance parameter by defining an alternative loss function. This choice will be subsequently justified by showing that it displays faithful error properties and is amenable to polynomial-time computation.

We use the following loss function

$$(3.1) \quad R(\Theta) = \mathbb{E}\left(-y \cdot \sum_{k=1}^m \log \theta_{\mathcal{X}_k} + \prod_{k=1}^m \theta_{\mathcal{X}_k}\right),$$

and the best approximate hierarchical decomposition

$$(3.2) \quad \hat{\Theta} = \arg \min \{ \hat{R}(\Theta) \mid \Theta \in \Omega \}.$$

is defined to be the minimizer of the empirical loss function

$$(3.3) \quad \hat{R}(\Theta) = \frac{1}{n} \sum_{i=1}^n \left(-y^{(i)} \cdot \sum_{k=1}^m \log \theta_{\mathcal{X}_k^{(i)}} + \prod_{k=1}^m \theta_{\mathcal{X}_k^{(i)}} \right)$$

subject to the constraint set

$$(3.4) \quad \Omega = \{ \Theta : M^{-1} \leq \prod_{k=1}^m \theta_{\mathcal{X}_k} \leq M, M^{-2} \leq \theta_{\mathcal{X}_k} \leq M^2, \forall x \in \mathcal{R} \}.$$

We justify this choice by first showing an equivalence to the usual squared loss function. Our second justification is it enables polynomial-time computation for specific approximation problems for positive tensors that are NP-hard in the case of a general (i.e., not necessarily positive) tensor, and this is shown by rewriting (3.2) as a convex optimization problem with a polynomial in ρ and n number of constraints.

3.4. Error Properties of Loss Function. We show the loss function (3.1) has favorable error properties. This loss function resembles the negative log-likelihood for a Poisson distribution: $\frac{1}{n} \sum_{i=1}^n (-y_i \log \mu + \mu)$, where $\mu > 0$ is the rate parameter of the distribution, and this is not surprising because this likelihood can be used to fit hierarchical log-linear models to contingency tables [18]. Furthermore, maximum likelihood decomposition of nonnegative tensors of count data using the Poisson distribution has been previously considered [13]. However, this is the wrong interpretation for our case because ψ_x can take continuous (non-integer) values and should not be interpreted as counts in general.

A better interpretation for the loss function (3.1) is as a Bregman divergence [4], or more specifically a generalized I-divergence (which is a generalization of the Kullback-Leibler divergence) [4, 39]. This is a more natural interpretation because of the following proposition that shows minimizing either our loss $R(\Theta)$ or the squared loss $L(\Theta)$ recovers the same solution when Γ is correct.

PROPOSITION 3.1 (Banerjee, et al., 2005 [4]). *If **A1, A2** hold and Γ is correct, then $\Theta^* \in \arg \min\{R(\Theta) \mid \Theta \in \Omega\} \Leftrightarrow \Theta^* \in \arg \min\{L(\Theta) \mid \Theta \in \Omega\}$. Moreover, the solution Θ^* has the property $\psi_x = \prod_{k=1}^m \theta_{\mathcal{X}_k}^*$.*

A further justification for using the loss (3.1) is that it is equivalent to the squared loss in the sense that it both majorizes and minorizes the squared loss.

PROPOSITION 3.2. *Under **A1, A2** and for any Γ , the loss function $R(\Theta)$ majorizes and minorizes the squared loss function $L(\Theta)$, meaning $a_l \cdot L(\Theta) + b_l \leq R(\Theta) \leq a_u \cdot L(\Theta) + b_u$, where constants $a_l, a_u > 0$ and b_l, b_u depend on μ, M .*

Proof. Define $\mathcal{M} = [(\mu M)^{-1}, \mu M]$, and consider the function $f(u) = -y \log(u) + u$ over the domain $u \in \mathcal{M}$. This function is strongly convex for $u \in \mathcal{M}$, and so we have $-y \log(u) + u \geq -y \log(v) + v + (-y/v + 1) \cdot (u - v) + y \cdot (u - v)/2(\mu M)^2$. Choosing $v = y$ gives $-y \log(u) + u \geq -\mu M \log(\mu M) + (\mu M)^{-1} + (u - y)^2/2(\mu M)^3$. The lower bound follows by setting $u = \prod_{k=1}^m \theta_{\mathcal{X}_k}$ and taking the expectation of both sides.

The upper bound is shown using the mean-value form of Taylor's theorem, which states that for any $u, v \in \mathcal{M}$: $-y \log(u) + u = -y \log(v) + v + (-y/v + 1) \cdot (u - v) + y \cdot (u - v)^2/2z^2$, for some $z \in \mathcal{M}$ between u and v . As a result, we have $-y \log(u) + u \leq -y \log(v) + v + (-y/v + 1) \cdot (u - v) + (\mu M)^2 \cdot y \cdot (u - v)^2/2$. Choosing $v = y$ gives $-y \log(u) + u \leq -y \log(y) + y + M^2 \cdot y \cdot (u - y)^2/2 \leq (\mu M)^{-1} \log(\mu M) + \mu M + (\mu M)^3 \cdot (u - y)^2/2$. The result follows by setting $u = \prod_{k=1}^m \theta_{\mathcal{X}_k}$ and taking the expectation of both sides. \square

3.5. Computational Properties. An equivalent reformulation of (3.2) can be defined using the following reparametrization of the loss function

$$(3.5) \quad R(U) = \mathbb{E} \left(-y \cdot \sum_{k=1}^m u_{\mathcal{X}_k} + \exp \left(\sum_{k=1}^m u_{\mathcal{X}_k} \right) \right),$$

and the relationship between parametrizations is that $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$. The loss function $R(U)$ is convex in $u_{\mathcal{X}_k}$, unlike the original parametrization (3.2) which is nonconvex in $\theta_{\mathcal{X}_k}$. Moreover, the $\prod r_i$ number of constraints in Ω can be reduced to a polynomial in ρ number of constraints by using a linear program (LP) lift [59]. Consider the set

$$(3.6) \quad \Phi = \left\{ U : \exists \eta_k, \nu_k \text{ s.t. } \eta_k \leq u_{\mathcal{X}_k} \leq \nu_k, -2 \log M \leq \eta_k, \nu_k \leq 2 \log M, \right. \\ \left. -\log M \leq \sum_{k=1}^m \eta_k, \sum_{k=1}^m \nu_k \leq \log M, \forall x \in \mathcal{R} \right\}.$$

We use this to define our reparametrized best approximate hierarchical decomposition as the minimizer to the following convex optimization problem

$$(3.7) \quad \hat{U} = \arg \min \{ \hat{R}(U) \mid U \in \Phi \},$$

where the reparametrized empirical loss function is

$$(3.8) \quad \hat{R}(U) = \frac{1}{n} \sum_{i=1}^n \left(-y\langle i \rangle \cdot \sum_{k=1}^m u_{\mathcal{X}_k\langle i \rangle} + \exp \left(\sum_{k=1}^m u_{\mathcal{X}_k\langle i \rangle} \right) \right).$$

The following proposition shows that (3.7) is equivalent to (3.2).

PROPOSITION 3.3. *Under **A1–A3** and for any Γ , the solution to (3.7) is equivalent to the solution of (3.2) with the invertible (under **A1**) mapping $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$.*

Proof. We have already argued above that $R(U)$ and $R(\Theta)$ are identical under $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$, and so to prove the first part we have to show Φ is equivalent to Ω when using the same mapping. Observe that for points belonging to Φ , we must have $\eta_k \leq \min_{\mathcal{X}_k \in \mathcal{R}_k} u_{\mathcal{X}_k}$ and $\max_{\mathcal{X}_k \in \mathcal{R}_k} u_{\mathcal{X}_k} \leq \nu_k$. Combining this with the other inequalities defining Φ leads to $-\log M \leq \min_{x \in \mathcal{R}} \sum_{k=1}^m u_k$ and $\max_{x \in \mathcal{R}} \sum_{k=1}^m u_k \leq \log M$, which is the same (under the equivalence) as $M^{-1} \leq \prod_{k=1}^m \theta_{\mathcal{X}_k} \leq M$ from Ω . A similar argument gives that $-2 \log M \leq \eta_k$ and $\nu_k \leq 2 \log M$ from Φ is the same as $M^{-2} \leq \theta_{\mathcal{X}_k} \leq M^2$ from Ω , under the equivalence. Because the objective and constraints of (3.7) and (3.2) are the same when equating $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$, we have that the solution to (3.7) is the same as the solution to (3.2). \square

We also have the following result about the polynomial-time solvability of (3.2). It is proved by considering the convex reparametrization (3.7) and explicitly defining a barrier function for a path-following interior-point method, and then using the methods of [44, 43] to conduct a complexity analysis for an interior-point method with this barrier function. The result is stated in terms of the complexity of computing an ϵ -solution, which is a solution x^ϵ of an optimization problem $f^* = \min\{f(x) \mid f_i(x) \leq 0, \forall i; x \in G\}$, such that (i) $f(x^\epsilon) - f^* \leq \epsilon$, (ii) $f_i(x^\epsilon) \leq \epsilon$ for all i , and (iii) $x^\epsilon \in G$.

PROPOSITION 3.4. *Under **A1–A3** and for any Γ , an ϵ -solution to the optimization problem (3.2) can be calculated with $O(1)(\rho^3 + n^3)\sqrt{\rho + n} \log(\frac{1}{\epsilon}(\mu\rho M \log M)(\rho + n))$ arithmetic steps, which is polynomial time in μ, ρ, M, n .*

Proof. We suitably modify the proof in [44] for the polynomial-time solvability of geometric programs: The first step is to reformulate the convex program (3.7) as following the convex program

$$(3.9) \quad \begin{aligned} \min \quad & \frac{1}{n} \sum_{i=1}^n \left(-y\langle i \rangle \cdot \sum_{k=1}^m u_{\mathcal{X}_k\langle i \rangle} + t_i \right) \\ \text{s.t.} \quad & \exp \left(\sum_{k=1}^m u_{\mathcal{X}_k\langle i \rangle} \right) - t_i \leq 0, \quad \forall i \in [n] \\ & \eta_k - u_{\mathcal{X}_k} \leq 0, \quad u_{\mathcal{X}_k} - \nu_k \leq 0, \quad \forall k \in [m], \mathcal{X}_k \in \mathcal{R}_k \\ & -\log M - \sum_{k=1}^m \eta_k \leq 0, \quad \sum_{k=1}^m \nu_k - \log M \leq 0 \\ & (u, \eta, \nu, t) \in G \end{aligned}$$

where $G = \{(u, \eta, \nu, t) : |\eta_k| \leq 2 \log M, |\nu_k| \leq 2 \log M, |u_{\mathcal{X}_k}| \leq 2 \log M, \forall k \in [m], \mathcal{X}_k \in \mathcal{R}_k; |t_i| \leq M, \forall i \in [n]\}$ is a bounded convex set. Note $x^+ = (0, 0, 0, 0)$ is the symmetry center of G , and so the asymmetry coefficient (see [44, 52]) of G with respect to x^+ is $\alpha(G : x^+) = 1$. From Propositions 5.1.3 and 5.4.1 of [44], it follows that

$$(3.10) \quad \begin{aligned} F = & -\sum_{k=1}^m \sum_{\mathcal{X}_k \in \mathcal{R}_k} \left\{ \log(2 \log M + u_{\mathcal{X}_k}) + \log(2 \log M - u_{\mathcal{X}_k}) \right\} + \\ & -\sum_{k=1}^m \left\{ \log(2 \log M + \eta_k) + \log(2 \log M - \eta_k) + \right. \\ & \left. \log(2 \log M + \nu_k) + \log(2 \log M - \nu_k) \right\} + \\ & -\sum_{i=1}^n \left\{ \log(M + t_i) + \log(M - t_i) \right\} \end{aligned}$$

is a $(2\rho + 4m + 2n)$ -self-concordant barrier for G .

The next step is to bound the objective and constraints of (3.9). Note **A1, A2** imply the absolute value of the objective is upper bounded by $\mu m M \log M + M$. Similarly, **A1, A2** imply the absolute value of the left hand side of the constraints are upper bounded by $2M, 4 \log M, 4 \log M, \log M + 2m \log M, \log M + 2m \log M$, respectively. Consequently, an upper bound on these upper bounds is $V = 4\mu m M \log M + 2M$.

The third step is to identify barrier functions for the epigraphs of each constraint in (3.9). Proposition 5.4.1 of [44] states $-\log(-\tau)$ is a 1-self-concordant barrier for the constraint $\tau \leq 0$. Similarly, Proposition 5.3.3 of [44] states $-\log(\log(\tau) - \xi) - \log(\tau)$ is a 2-self-concordant barrier for the constraint $\exp(\xi) \leq \tau$. Consequently, the following

$$(3.11) \quad -\log(3Vt/\epsilon - V - \frac{1}{n} \sum_{i=1}^n (-y\langle i \rangle \cdot \sum_{k=1}^m u_{\mathcal{X}_k\langle i \rangle} + t_i)) + \\ - \sum_{i=1}^n \{ \log(\log(t_i + t) - \sum_{k=1}^m u_{\mathcal{X}_k\langle i \rangle}) + \log(t_i + t) \} + \\ - \sum_{k=1}^m \sum_{\mathcal{X}_k \in \mathcal{R}_k} \{ \log(-\eta_k + u_{\mathcal{X}_k}) + \log(-u_{\mathcal{X}_k} + \nu_k) \} + \\ - \log(\log M + \sum_{k=1}^m \eta_k) - \log(\log M - \sum_{k=1}^m \nu_k) + F$$

is a $(3\rho + 4m + 4n + 3)$ -self-concordant barrier function by Proposition 5.1.3 of [44]. Note we have the bound $\rho \geq m$ from the definition of ρ . From the results of §6.1 of [44], an ϵ -solution to (3.7) can be found in $O(1)\sqrt{\rho + n} \log(\frac{1}{\epsilon}(\mu\rho M \log M)(\rho + n))$ steps of the path-following algorithm. The Newton system for one step is assembled in $O(nm^2 + \rho)$ arithmetic steps and can be solved in $O(\rho^3 + n^3)$ arithmetic steps. Consequently, an ϵ -solution to (3.7) can be found in $O(1)(\rho^3 + n^3)\sqrt{\rho + n} \log(\frac{1}{\epsilon}(\mu\rho M \log M)(\rho + n))$ arithmetic steps. The proof concludes by noting Proposition 3.3 implies an ϵ -solution to (3.2) can be calculated by applying the transformation $\theta_{\mathcal{X}_k} = \exp(u_{\mathcal{X}_k})$ to the ϵ -solution to (3.7). \square

This result immediately implies that the best rank-1 approximation of a positive tensor can be computed in polynomial time, which is in contrast to the general case where computing the best rank-1 approximation is NP-hard [27]. The approximation problem becomes easier when we restrict our focus to positive tensors.

COROLLARY 3.5. *The best rank-1 approximation, under the loss function (3.1) and satisfying **A1**, of a tensor ψ can be computed in polynomial time with a number of arithmetic steps that is polynomial in r, p, n, μ, M .*

Proof. The best rank-1 approximation corresponds to a partition decomposition with $\text{facets}(\Gamma) = \{\{1\}, \dots, \{p\}\}$, and so the result follows from Proposition 3.4. \square

3.6. Bound on Squared Error in Terms of Loss Function. Define the oracle parameters to be any $\Theta^* \in \arg \min\{R(\Theta) \mid \Theta \in \Omega\}$. Below, we provide a relationship between a squared error function involving Θ^* and the loss function (3.2). This relationship will serve as useful machinery for proving subsequent results.

PROPOSITION 3.6. *Under **A1, A2** and for any Γ , we have for any $\Theta \in \Omega$ that $\frac{1}{2M^3} \cdot \mathbb{E}((\prod_{k=1}^m \theta_{\mathcal{X}_k} - \prod_{k=1}^m \theta_{\mathcal{X}_k}^*)^2) \leq R(\Theta) - R(\Theta^*)$.*

Proof. We will use the equivalent (by Proposition 3.3) convex reparameterization in U to show the necessary bound. The first-order optimality condition [53] for the reparametrized optimization problem (3.7) is

$$(3.12) \quad D(U^*, U) = \nabla R(U^*) \cdot (U - U^*) \geq 0,$$

for all $U \in \Phi$. Since the probability space of $x \in \mathcal{R}$ is finite, we can interchange the order of differentiation and integration as shown below

$$(3.13) \quad \partial_{\mathcal{X}_j} R(U) = \sum_{x \in \mathcal{R}} f_x \cdot (\partial_{\mathcal{X}_j} (-\psi_x \cdot \sum_{k=1}^m u_{\mathcal{X}_k} + \exp(\sum_{k=1}^m u_{\mathcal{X}_k})))$$

$$(3.14) \quad = \sum_{x \in \mathcal{R}} f_x \cdot (-\psi_x + \exp(\sum_{k=1}^m u_{\mathcal{X}_k})) \cdot \mathbb{1}_{\mathcal{X}_j},$$

where $f_x = \mathbb{P}(x)$. Combining (3.12) and (3.14) leads to

$$(3.15) \quad D(U^*, U) = \sum_{\mathcal{X}_j} \sum_{x \in \mathcal{R}} f_x \cdot (-\psi_x + \exp(\sum_{k=1}^m u_{\mathcal{X}_k}^*)) \cdot \mathbb{1}_{\mathcal{X}_j} \cdot (u_{\mathcal{X}_j} - u_{\mathcal{X}_j}^*)$$

$$(3.16) \quad = \mathbb{E}((- \psi_x + \exp(\sum_{k=1}^m u_{\mathcal{X}_k}^*)) \cdot (\sum_{k=1}^m u_{\mathcal{X}_k} - \sum_{k=1}^m u_{\mathcal{X}_k}^*)).$$

Next, consider $f(u) = -yu + e^u$. Since $f''(u) \geq e^a$ for all $u \in [a, b]$, this function is strongly convex [8] and satisfies $-yu + e^u \geq -yv + e^v + (-y + e^v) \cdot (u - v) + e^a/2 \cdot (u - v)^2$, for all $u, v \in [a, b]$. Applying this inequality to $R(U)$ gives that for any $U \in \Phi$, $R(U) \geq R(U^*) + D(U^*, U) + \frac{1}{2M} \cdot \mathbb{E}((\sum_{k=1}^m u_{\mathcal{X}_k}^* - \sum_{k=1}^m u_{\mathcal{X}_k})^2)$, where we have used (3.16) to simplify the expression. Since $D(U^*, U) \geq 0$ from (3.12), we have that for any $U \in \Phi$, $R(U) - R(U^*) \geq \frac{1}{2M} \cdot \mathbb{E}((\sum_{k=1}^m u_{\mathcal{X}_k}^* - \sum_{k=1}^m u_{\mathcal{X}_k})^2)$. Because e^u is Lipschitz on bounded domains (i.e., $|e^u - e^v| \leq e^{\log M} \cdot |u - v|$, for all $-\log M \leq u, v \leq \log M$), we have that for any $U \in \Phi$, $R(U) - R(U^*) \geq \frac{1}{2M^3} \cdot \mathbb{E}((\exp(\sum_{k=1}^m u_{\mathcal{X}_k}^*) - \exp(\sum_{k=1}^m u_{\mathcal{X}_k}))^2)$. Inverting the mapping $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$, which is possible because of **A1**, gives $R(\Theta) - R(\Theta^*) \geq \frac{1}{2M^3} \cdot \mathbb{E}((\prod_{k=1}^m \theta_{\mathcal{X}_k}^* - \prod_{k=1}^m \theta_{\mathcal{X}_k})^2)$. \square

3.7. Risk Consistency via Rademacher Complexity. Having shown that the loss function (3.1) has promising properties, we next identify sufficient conditions for risk consistency [5, 26, 29, 32]. Our approach is to interpret the problem as a high-dimensional (though lower-dimensional than if we had not taken the low-rank tensor structure into consideration) linear regression under a Lipschitz loss function. The linear regression will not be with respect to the indices x , but will instead be defined using indicator functions. With this interpretation, we will use Rademacher averages [5, 29, 32, 7] to bound the complexity of our model (3.1).

PROPOSITION 3.7. *Under **A1–A3** and for any Γ , we have*

$$(3.17) \quad \mathbb{P}\left(\sup_{\Theta \in \Omega} |\hat{R}(\Theta) - R(\Theta)| < t\right) \geq 1 - \exp\left(-C_1 n \left(t - C_2 \sqrt{\frac{m\rho}{n}}\right)^2\right),$$

where constants $C_1, C_2 > 0$ depend on μ, M .

Proof. The proof proceeds similarly to [5, 32] by bounding the deviation of the supremum from the expectation of the supremum, and it will be easier to work in the reparametrized space. First, note that $\hat{R}(U)$ satisfies the bounded deviation condition with constant $(\mu M \log M + M)/n$ [7] because of **A1, A2**. As a result, McDiarmid's inequality [7] gives

$$(3.18) \quad \mathbb{P}\left(\sup_{U \in \Phi} |\Delta(U)| - \mathbb{E}(\sup_{U \in \Phi} |\Delta(U)|) > t\right) \leq \exp\left(\frac{-2nt^2}{(\mu M \log M + M)^2}\right),$$

where $\Delta(U) = \hat{R}(U) - R(U)$. And so the result follows if we can bound the quantity $\mathbb{E}(\sup_{U \in \Phi} |\Delta(U)|)$. Because the loss function $\phi(z) = -yz + e^z$ (for a fixed value of y and for $z \in [-\log M, \log M]$) is Lipschitz with respect to z with Lipschitz constant $L = \mu M + M$, structural results [34, 5] give that

$$(3.19) \quad \mathbb{E}\left(\sup_{U \in \Phi} |\Delta(U)|\right) \leq 4L \cdot R(F_W),$$

where $R(F_W)$ is the Rademacher complexity for an appropriate linear function class. In particular, we can define our empirical loss by taking the sample average of ϕ composed with the linear model $\sum_{k=1}^m \sum_{\mathcal{X}_k \in \mathcal{R}_k} \mathbb{1}_{\mathcal{X}_k = \mathcal{X}_k^{(i)}} \cdot u_{\mathcal{X}_k}$. We should interpret the terms $\mathbb{1}_{\mathcal{X}_k = \mathcal{X}_k^{(i)}}$ as pseudo-predictors, and the $u_{\mathcal{X}_k}$ are still the parameters. The key observation is that if we define $\chi \in \{0, 1\}^\rho$ to be the vector of pseudo-predictors,

then in fact $\|\chi\|_1 = m$, $\|\chi\|_2 = \sqrt{m}$, and $\|\chi\|_\infty = 1$. Recall that Φ is defined so that $\|u_{\mathcal{X}_k}\|_\infty \leq 2 \log M$. And so results from [29] imply that $R(F_W) \leq 2 \log M \cdot \sqrt{m\rho/n}$. The result follows by combining this with (3.18) and (3.19). \square

The above result can be used to show risk consistency of the ϵ -solution $\hat{\Theta}^\epsilon$ to the best approximate hierarchical decomposition problem (3.2):

THEOREM 3.8. *Under **A1–A3** and for any Γ , with probability at least $1 - c_1 n^{-1}$ we have $0 \leq R(\hat{\Theta}^\epsilon) - R(\Theta^*) \leq \sqrt{m\rho \log n/n} + \epsilon$, where constant $c_1 > 0$ depends on μ, M .*

Proof. The proof follows that in [26] with modifications to extend the argument for ϵ -solutions. The triangle inequality implies $|R(\hat{\Theta}^\epsilon) - R(\Theta^*)| \leq |R(\hat{\Theta}^\epsilon) - \hat{R}(\hat{\Theta}^\epsilon)| + |R(\Theta^*) - \hat{R}(\hat{\Theta}^\epsilon)|$, and so we need to bound these two terms. The first term $|R(\hat{\Theta}^\epsilon) - \hat{R}(\hat{\Theta}^\epsilon)|$ is bounded by Proposition 3.7, and so we only need to focus on the second term $|R(\Theta^*) - \hat{R}(\hat{\Theta}^\epsilon)|$. Because the quantity $\hat{\Theta}^\epsilon$ is an ϵ -solution to an optimization problem with objective function $\hat{R}(\cdot)$, we have $\hat{R}(\hat{\Theta}^\epsilon) \leq \hat{R}(\Theta^*) + \epsilon \Rightarrow \hat{R}(\hat{\Theta}^\epsilon) - R(\Theta^*) \leq \hat{R}(\Theta^*) - R(\Theta^*) + \epsilon$. Similarly, because Θ^* is the minimizer of $R(\cdot)$, we have $R(\Theta^*) \leq R(\hat{\Theta}^\epsilon) \Rightarrow \hat{R}(\hat{\Theta}^\epsilon) - R(\Theta^*) \geq \hat{R}(\hat{\Theta}^\epsilon) - R(\hat{\Theta}^\epsilon)$. The result follows from combining the above with Proposition 3.7. \square

3.8. Polynomial-Time Hierarchical Decompositions and Approximations. We are now in a position to provide a randomized algorithm that can compute a hierarchical decomposition with time that is polynomial in $\rho(\Gamma)$, as opposed to the algorithm given in the proof of Proposition 2.2 that is polynomial in $\prod r_i$. Let δ be a parameter that controls the approximation accuracy of the decomposition. Then given any Γ , the algorithm is as follows:

1. Set $n = \rho(\Gamma)/\delta$;
2. Sample indices $x\langle i \rangle \in \mathcal{R}$ and record the corresponding tensor entries $y\langle i \rangle$, for $i = 1, \dots, n$;
3. Compute \hat{U}^ϵ by solving (3.2);
4. Compute $\hat{\Theta}^\epsilon$ by using the mapping $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$.

We will use Proposition 3.6 and Theorem 3.8 to reason about errors measured using the squared loss function. Recall that δ controls the approximation accuracy of the decomposition, and the value ϵ controls the accuracy of the optimization solution.

THEOREM 3.9. *Suppose **A1–A3** hold, Γ is correct for ψ_x , and the indices $x\langle i \rangle$ are sampled uniformly from \mathcal{R} . Then with probability at least $1 - c_1 \delta/\rho$ the above algorithm computes a hierarchical decomposition $\hat{\Theta}^\epsilon$ with average approximation error*

$$(3.20) \quad \left(\prod r_i\right)^{-1} \sum_{x \in \mathcal{X}} \left(\psi_x - \prod_{k=1}^m \hat{\theta}_{\mathcal{X}_k}^\epsilon\right)^2 \leq 2M^3 \left(\sqrt{m\delta \log(\rho/\delta)} + \epsilon\right)$$

and has a polynomial-time arithmetic cost $O(1)(\rho/\delta)^{7/2} \log\left(\frac{1}{\delta\epsilon}(\mu\rho^2 M \log M)\right)$, where constant $c_1 > 0$ depends on μ, M .

Proof. We can replace $\prod_{k=1}^m \theta_{\mathcal{X}_k}^*$ with ψ_x because of Proposition 3.1. Combining Theorem 3.8 with Proposition 3.6 gives $\frac{1}{2M^3} \cdot \mathbb{E}\left(\left(\prod_{k=1}^m \hat{\theta}_{\mathcal{X}_k}^\epsilon - \psi_x\right)^2\right) \leq \sqrt{m\delta \log(\rho/\delta)} + \epsilon$ with probability at least $1 - c_1 \delta/\rho$. When $x\langle i \rangle$ is uniformly sampled, this expectation can be written as $\mathbb{E}\left(\left(\psi_x - \prod_{k=1}^m \hat{\theta}_{\mathcal{X}_k}^\epsilon\right)^2\right) = \left(\prod r_i\right)^{-1} \sum_{x \in \mathcal{X}} \left(\psi_x - \prod_{k=1}^m \hat{\theta}_{\mathcal{X}_k}^\epsilon\right)^2$. Proposition 3.4 states $\hat{\Theta}^\epsilon$ has arithmetic cost $O(1)(\rho/\delta)^{7/2} \log\left(\frac{1}{\epsilon}(\mu\rho^2 M \log M/\delta)\right)$. \square

When Γ is incorrect for ψ_x , only a weaker result is possible because it will be the case that $\psi_x \neq \prod_{k=1}^m \theta_{\mathcal{X}_k}^*$ for the oracle parameters Θ^* . In other words, the previous result states we can compute hierarchical decompositions (with a correct Γ) in polynomial time using an amount of data that depends on effective dimension ρ (rather than the number of tensor entries $\prod r_i$), whereas the following result states we

can compute best tensor approximations (where the approximation has a hierarchical decomposition given by a potentially incorrect Γ) in polynomial time using an amount of data that depends on effective dimension ρ .

THEOREM 3.10. *Suppose **A1–A3** hold, and the indices $x\langle i \rangle$ are sampled uniformly from \mathcal{R} . Then with probability at least $1 - c_1\delta/\rho$ the above algorithm computes an approximate hierarchical decomposition $\hat{\Theta}^\epsilon$ with average approximation error*

$$(3.21) \quad \left(\prod r_i\right)^{-1} \sum_{x \in \mathcal{X}} \left(\left(\prod_{k=1}^m \theta_{\mathcal{X}_k}^* - \prod_{k=1}^m \hat{\theta}_{\mathcal{X}_k}^\epsilon \right)^2 \right) \leq 2M^3 \left(\sqrt{m\delta \log(\rho/\delta)} + \epsilon \right)$$

and has a polynomial-time arithmetic cost $O(1)(\rho/\delta)^{7/2} \log\left(\frac{1}{\delta\epsilon}(\mu\rho^2 M \log M)\right)$, where constant $c_1 > 0$ depends on μ, M .

Proof. This result is proved in the proof of Theorem 3.9. \square

One of the implications of this is that we can compute the best (as measured by the loss (3.1)) approximate hierarchical decomposition of ψ in an amount of time that is polynomial in the effective dimension ρ induced by the simplicial complex Γ . We can also specialize these results to rank-1 approximations.

COROLLARY 3.11. *The best rank-1 approximation, under the loss function (3.1) and satisfying **A1**, of a tensor ψ can be computed in polynomial time with a number of arithmetic steps that is polynomial in r, p, μ, M .*

Proof. The best rank-1 approximation corresponds to a partition decomposition with facets $(\Gamma) = \{\{1\}, \dots, \{p\}\}$, and so the result follows from Theorem 3.10. \square

4. Tensor Completion. The tensor completion problem is almost the same as computing a hierarchical decomposition using data samples of tensor entries, and the only difference is we must also determine Γ using the measured data before computing the tensor approximation. A key assumption for formulating completion problems is the object being estimated has low rank, because this allows for statistically consistent estimation with a small number of measurements. The usual idea for solving the completion problem is to formulate an optimization problem in which the rank (or a rank surrogate) of the estimated object is minimized subject to the entries that have been observed being equal to the corresponding entries in the estimated object.

Matrix completion is well-studied [20, 63, 51, 9, 10, 38, 1, 54, 30, 12, 23], but tensor completion is still an open problem. Tensor rank is NP-hard to compute [27] and has poor continuity properties [16]. As a result, existing approaches use multilinear rank as a surrogate for tensor rank because it can be computed in polynomial time [42] and has better continuity properties than tensor rank [16]. The multilinear (or Tucker) rank of tensor ψ is the vector: $\text{rank}_{\boxplus}(\psi) = (\text{rank}(\psi_{(1)}), \dots, \text{rank}(\psi_{(p)}))$, where $\text{rank}(\cdot)$ is the standard matrix rank and $\psi_{(k)}$ is the unfolding of the tensor (into a matrix) along the k -th index [16, 42].

Several approaches to tensor completion [56, 55, 22, 37, 42, 62, 40] use soft-thresholding on the multilinear rank, because this converts the problem into the well-studied matrix completion problem. A canonical formulation [56, 22] is to solve $\min_{\hat{\psi}} \frac{1}{n} \sum_{i=1}^n (y\langle i \rangle - \hat{\psi}_{x\langle i \rangle})^2 + \sum_{k=1}^p \lambda_k \cdot \|\hat{\psi}_{(k)}\|_*$, where $\|\cdot\|_*$ denotes the nuclear norm and $\lambda_k > 0$ are weightings. There are open questions on the optimal weighting λ_k in this formulation [48, 42, 62]. Another class of approaches use power iteration, message passing, or alternating minimization algorithms [24, 40]; these are local approaches that are not guaranteed to provide statistical consistency in general, though they can empirically work well on specific problem instances.

However, there is a large gap with the statistical convergence rates achievable by the above methods [56, 55, 22, 37, 42, 62, 40]. If we define $\pi = \max_j \text{rank}_{\boxplus}(\psi)$, then

existing convex optimization-based algorithms need $O(\pi^{\lfloor p/2 \rfloor} r^{\lceil p/2 \rceil})$ measurements; this is substantially worse than the best rates achievable by an NP-hard formulation, which needs $O(\pi^p + r\pi p)$ measurements [42]. Another analysis focusing on rank-1 tensors found that existing local approaches like power iteration and message passing need a diverging signal-to-noise ratio to achieve statistical consistency [40].

Here, we use effective dimension $\rho(\Gamma)$ of a partition Γ as a surrogate for tensor rank. As discussed in §2.2, ρ majorizes tensor rank, and so a small ρ corresponds to a low tensor rank. We will focus on the case where Γ is a partition and leave open the more general case of a Γ that is a general simplicial complex. One advantage of developing tensor completion algorithms using effective dimension is that we will be able to show this leads to methods that achieve statistical consistency using only slightly more measurements than NP-hard formulations in specific cases.

The section begins by discussing which correct partition would be best for statistical purposes. Next, we define a test statistic that can distinguish whether two indices belong to the same or different facets of this partition. This statistic can be used to construct a partition Γ from the data, and we consequently use this test statistic to construct a polynomial-time algorithm for tensor completion. A theoretical analysis shows our algorithm needs exponentially less data for specific cases than existing tensor completion methods [56, 55, 22, 37, 42, 62, 40]. The section concludes by discussing how to modify the algorithm to handle a tradeoff between purposely choosing an incorrect partition (which increases statistical bias) in exchange for significant reduction in the effective dimension (which reduces statistical variance).

Before we begin, we make a note regarding the computational complexity of computing estimates by solving optimization problems. It is typical in the statistics literature [20, 63, 51, 9, 10, 38, 1, 54, 30, 12, 23, 56, 55, 22, 37, 42, 62, 40] to not make a distinction between ϵ -solutions and exact solutions to optimization problems. The reason is that the ϵ -solutions generally add only an ϵ term in upper bounds on error (see for instance Theorems 3.9 and 3.10). In this section, we follow this convention from statistics and do not distinguish between ϵ -solutions and exact solutions. Polynomial time computation is ensured in our case because of Proposition 3.4.

4.1. Specifying the Ideal Partition. Despite the lack of uniqueness of correct Γ for a tensor ψ_x , some correct Γ are better than others. Statistically, a correct Γ with the smallest effective dimension $\rho(\Gamma)$ is the best choice because this reduces the number of parameters to estimate and leads to more efficient methods. As a result, we define the *ideal partition* Γ^* to be a partition such that $\rho(\Gamma^*) \leq \rho(\Gamma)$ for all correct partitions Γ . An ideal partition must always exist because $\text{facets}(\Gamma) = \{\{1, \dots, p\}\}$ is a correct partition, but an ideal partition may not always have low effective dimension: In a subsequent subsection, we will discuss low-rank approximations that can be used for this situation. However, if an ideal partition exists then it must be unique because otherwise we could use all ideal partitions to define a new correct partition Γ with a strictly smaller effective dimension $\rho(\Gamma)$. Additionally, because the loss function (3.1) is a Bregman divergence, it is known [4] that the minimal possible risk is

$$(4.1) \quad R(\psi_x) := \arg \min \{R(\Theta) \mid \Theta \in \Omega, \forall \text{ choices of } \Gamma\} = R(\Theta^*(\Gamma^*)),$$

where $R(\Theta^*(\Gamma^*))$ denotes the minimum risk under the ideal partition Γ^* .

4.2. Defining the Risk Gap. We begin by making a minor assumption about the distribution of the predictors:

A4. The $x\langle i \rangle$ are iid random variables with distribution such that $x_u\langle i \rangle$ is independent of $x_v\langle i \rangle$ whenever $u \in F_j$ and $v \in F_k$, where $F_j, F_k \in \text{facets}(\Gamma^*)$ and $F_j \neq F_k$.

This independence assumption is similar to assumptions typically made for low-rank matrix and tensor completion (e.g., [56, 12, 42]). The typical assumption is that entries of the matrix (or tensor) are sampled with uniform probability, which is equivalent to assuming the $x_u\langle i \rangle$ are jointly independent [33]. Here, we only require independence between indices that belong to different facets of Γ^* . It is useful to emphasize that uniform sampling of entries would satisfy our assumption.

This assumption could be generalized to include approximately independent distributions. For instance, consider the distribution on x given by: $f_x = (1 - \epsilon) \cdot \bigotimes_{k=1}^p V_k + \epsilon \cdot g_x$, where $V_k \in \mathbb{R}^{r_k}$ are nonnegative vectors that sum to one $\sum_j V_k^j = 1$, $\epsilon \ll 1$ is a small constant, and g_x is an arbitrary probability distribution on x . Because $\bigotimes_{k=1}^p V_k$ represents a distribution where each x_k is jointly independent [33], we can interpret the distribution f_x as having approximate independence between the x_k . Under such conditions, we can bound the error incurred by our estimators assuming **A4**. We do not consider the details of this generalization in this paper.

Our main idea is that we can determine structural properties of the ideal partition Γ^* by computing quantities that are statistically easy to estimate. In particular, we define the *risk gap* of two indices j, q to be the test statistic $\mathcal{G}_{jq} = \min\{\bar{R}_{jq}(\bar{B}) \mid \bar{B} \in \bar{\Phi}\} - \min\{R_{jq}(B) \mid B \in \Phi\}$, where $\bar{\Phi}$ is the set (3.4) for the partition $\{\{j\}, \{q\}\}$, Φ in this case is the set (3.4) for the partition $\{\{j, q\}\}$, $R_{jq}(B) = \mathbb{E}(-y \log(\beta_{x_j, x_q}) + \beta_{x_j, x_q})$, and $\bar{R}_{jq}(\bar{B}) = \mathbb{E}(-y \cdot (\log \bar{\beta}_{x_j} + \log \bar{\beta}_{x_q}) + \bar{\beta}_{x_j} \bar{\beta}_{x_q})$. Note that $\mathcal{G}_{jq} \geq 0$ because

$$(4.2) \quad \min\{\bar{R}_{jq}(\bar{B}) \mid \bar{B} \in \bar{\Phi}\} = \min\{R_{jq}(B) \mid B \in \Phi, \beta_{x_j, x_q} = \bar{\beta}_{x_j} \bar{\beta}_{x_q}, \bar{B} \in \bar{\Phi}\}.$$

The following structural characterization is essentially a corollary to results in [4].

PROPOSITION 4.1. *Suppose **A1**, **A2**, **A4** hold and that Γ^* is an ideal partition. If indices j, q are such that there is no $F_k \in \text{facets}(\Gamma^*)$ with $j, q \in F_k$, then $\mathcal{G}_{jq} = 0$.*

Proof. Because of **A4**, without loss of generality we have $F_1, F_2 \in \text{facets}(\Gamma^*)$ such that $j \in F_1, q \in F_2$, and $F_1 \neq F_2$. Next observe that $\beta_{x_j, x_q} = \mathbb{E}[y|x_j, x_q]$ minimizes $\min\{R_{jq}(B) \mid B \in \Phi\}$ [4]; this conditional expectation can be written as $\beta_{x_j, x_q} = \mathbb{E}[\prod_{k=1}^m \theta_{\mathcal{X}_k} | x_j, x_q] = \mathbb{E}[\theta_{\mathcal{X}_1} | x_j] \cdot \mathbb{E}[\theta_{\mathcal{X}_2} | x_q] \cdot \mathbb{E}[\prod_{k=3}^m \theta_{\mathcal{X}_k}]$. Defining $v_{x_j} = \mathbb{E}[\theta_{\mathcal{X}_1} | x_j]$, $w_{x_q} = \mathbb{E}[\theta_{\mathcal{X}_2} | x_q]$, and $\kappa = \mathbb{E}[\prod_{k=3}^m \theta_{\mathcal{X}_k}]$, we can write the conditional expectation as $\beta_{x_j, x_q} = \kappa \cdot v_{x_j} w_{x_q}$. Because **A1** holds and Γ^* is an ideal partition, this means the $\theta_{\mathcal{X}_k}$ are strictly positive. As a result, we have that (i) κ is a strictly positive constant, and (ii) the vectors u_{x_j}, v_{x_q} have strictly positive entries.

Next, observe that if we can choose $\bar{\beta}_{x_j}$ and $\bar{\beta}_{x_q}$ such that $\beta_{x_j, x_q} = \bar{\beta}_{x_j} \bar{\beta}_{x_q}$ and $\bar{B} \in \bar{\Phi}$, then the result follows because the minimizer to $\min\{R_{jq}(B) \mid B \in \Phi\}$ also gives the minimizer to (4.2). In fact, such a choice is guaranteed to exist by Proposition 2.2 applied to β_{x_j, x_q} , since we have the decomposition $\beta_{x_j, x_q} = \kappa \cdot u_{x_j} v_{x_q}$. \square

This is a useful result because it says that important structural information is encoded in an object \mathcal{G}_{jq} that is easy to estimate. Unfortunately, the converse is not true. Consider the counterexample with $p = 3, r_1 = r_2 = r_3 = 2$, and

$$(4.3) \quad (\psi_{x_1, x_2, 1} \mid \psi_{x_1, x_2, 2}) = \begin{pmatrix} 2 & 0 & 0 & 2 \\ 0 & 2 & 2 & 0 \end{pmatrix},$$

where the entries are measured uniformly. It can be shown that $\mathcal{G}_{jq} = 0$, but the hypothesis of Proposition 4.1 does not hold. Consequently, we will have to restrict

the class of low-rank tensors we consider by defining an incoherence condition:

A5. There exists $\alpha > 0$ such that $\mathcal{G}_{jq} \geq \alpha$ for all $j, q \in F_k$ and all $F_k \in \text{facets}(\Gamma^*)$.

Incoherence conditions are common in the matrix and tensor completion literature [51, 9, 10, 38, 1, 54, 30, 12, 23]. One interpretation of **A5** is it forces \mathcal{G}_{jq} to represent the difference in risk (flattened to just two variables) between keeping x_j, x_q coupled versus decoupled in the ideal partition Γ^* . The existence of tensors satisfying this condition can be seen by considering the second example from §2.4 in which $\psi_x = \prod_{k=1}^m \theta_{\mathcal{X}_k} = (\theta^{(1)} \otimes \dots \otimes \theta^{(m)})_x$, where $\theta^{(1)}, \dots, \theta^{(s)}$ are matrices and $\theta^{(s+1)}, \dots, \theta^{(m)}$ are vectors. Assume that (i) the entries of $\theta^{(k)}$ lie within the set $[M^{-1/m}, M^{1/m}]$, for all $k = 1, \dots, m$, (ii) we sample uniformly from the tensor, and (iii) there is a constant $\alpha > 0$ such that the singular values of each matrix (i.e., $k = 1, \dots, s$) satisfy $\sum_{\gamma \geq 2} \sigma_\gamma(\theta^{(k)}) \geq \sqrt{2}(rM)^{3/2}\sqrt{\alpha}$, where $\sigma_\gamma(\cdot)$ are the singular values sorted into decreasing order. Then Proposition 3.2 gives $\mathcal{G}_{jq} \geq \min\{\frac{1}{2M^3} \cdot \mathbb{E}((\bar{\beta}_{x_j}\bar{\beta}_{x_q} - \mathbb{E}[y|x_j, x_q])^2) \mid B \in \bar{\Phi}\} \geq \frac{1}{2(rM)^3} \cdot (\sum_{\gamma \geq 2} \sigma_\gamma(\theta^{(k)}))^2 = \alpha$, where we used that $\|A\|_* \leq \sqrt{r}\|A\|_F$ for a matrix A with dimensions upper bounded by r , and that the probability of a single entry being observed when entries are observed uniformly is lower bounded by $1/r^2$. Hence, these tensors satisfy **A5** by construction.

We lastly turn to the question of interpretation of the incoherence condition **A5**. There is a large amount of incoherence in the above class of tensors because either two indices j, q are decoupled because they lie in distinct facets or these indices jointly belong to the same facet that is decoupled from every other index. Interpreted in this way, we can see why the example (4.3) displays pathological behavior: The value of an entry in the tensor ψ is very sensitive to changes in x_3 , and so the indices 1, 2 do not have sufficient incoherence from the index 3 for our property **A5** to hold.

4.3. Tensor Completion Algorithm for Low-Rank Ideal Partitions. As we have shown above, when **A1–A5** are satisfied, the risk gap \mathcal{G}_{jq} is zero (non-zero) when the indices j, q are decoupled (coupled) in the ideal partition Γ^* . The idea of our algorithm is that we will use estimates of the risk gap $\hat{\mathcal{G}}_{jq}$ to construct an estimate of the ideal partition $\hat{\Gamma}$, and this will lead to a consistent estimation procedure because estimates of the risk gap converge significantly faster than estimates of the completed tensor. Let t_n be a threshold. The steps are:

1. Define the initial partition to be $\text{facets}(\hat{\Gamma}) = \{\{1\}\}$. The remaining variables will be subsequently added to the partition.
2. For the variables indicated by $j = 2, \dots, p$, do the following:
 - (a) For the partitions represented by $k = 1, \dots, \#\text{facets}(\hat{\Gamma})$
 - i. Let $q = (F_k)_1$ and compute the empirical version of the risk gap: $\hat{\mathcal{G}}_{jq} = \min\{\hat{R}_{jq}(\bar{B}) \mid \bar{B} \in \bar{\Phi}\} - \min\{\hat{R}_{jq}(B) \mid B \in \Phi\}$, where $\hat{R}_{jq}(B) = \frac{1}{n} \sum_{i=1}^n (-y^{(i)}) \cdot \log(\beta_{x_j^{(i)}, x_q^{(i)}}) + \beta_{x_j^{(i)}, x_q^{(i)}}$ and also $\hat{R}_{jq}(\bar{B}) = \frac{1}{n} \sum_{i=1}^n (-y^{(i)}) \cdot (\log \bar{\beta}_{x_j^{(i)}} + \log \bar{\beta}_{x_q^{(i)}}) + \bar{\beta}_{x_j^{(i)}} \bar{\beta}_{x_q^{(i)}}$.
 - ii. If $\hat{\mathcal{G}}_{jq} > t_n$, then add j to the k -th facet ($\hat{F}_k = \hat{F}_k \cup j$) and break this inner loop.
 - (b) If j was not added to any facet, then add j as its own facet ($\text{facets}(\hat{\Gamma}) = \text{facets}(\hat{\Gamma}) \cup \{j\}$).
3. Compute $\hat{\Theta}$ by solving (3.7) with the partition $\hat{\Gamma}$ and then inverting the mapping $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$.

Our first result on the consistency of this estimation procedure applies to cases in which we know the value of α and set the threshold to $t_n = \alpha/2$. Technically, this result applies to any threshold $t_n = \alpha/\eta$ for any $\eta \in (0, 1)$.

THEOREM 4.2. *If **A1–A5** are satisfied and $t_n = \alpha/2$, then with probability at least $1 - c_1 n^{-1} - 2p^2 \cdot \exp(-c_2 n(\alpha/4 - c_3 r/\sqrt{n})^2)$ we have $0 \leq R(\hat{\Theta}) - R(\psi_x) \leq \sqrt{m\rho \log n/n}$, where constants $c_1, c_2, c_3 > 0$ depend on μ, M .*

Proof. Two types of mistakes can occur when estimating the ideal partition Γ^* using the values $\hat{\mathcal{G}}_{jq}$: Either (i) $\mathcal{G}_{jq} = 0$ but $\hat{\mathcal{G}}_{jq} > \alpha/2$, or (ii) $\mathcal{G}_{jq} \geq \alpha$ but $\hat{\mathcal{G}}_{jq} \leq \alpha/2$. Restated, a type (i) error does not occur if $|\hat{\mathcal{G}}_{jq} - \mathcal{G}_{jq}| < \alpha/2$, and a type (ii) error does not occur if $|\hat{\mathcal{G}}_{jq} - \mathcal{G}_{jq}| < \alpha/2$. And because the estimation procedure is constructed such that the maximum number of $\hat{\mathcal{G}}_{jq}$ estimates that will be computed is $p(p-1)/2$, Proposition 3.7 implies $\mathbb{P}(\max_{(j,q) \in \mathcal{J}} |\hat{\mathcal{G}}_{jq} - \mathcal{G}_{jq}| < \alpha/2) \geq 1 - 2p^2 \cdot \exp(-c_2 n(\alpha/4 - c_3 r/\sqrt{n})^2)$, where \mathcal{J} is the set of indices (j, q) for which $\hat{\mathcal{G}}_{jq}$ is computed, and $c_2, c_3 > 0$ are constants that depend on μ, M . This expression is the probability that the estimated partition $\hat{\Gamma}$ is equal to the ideal partition Γ^* .

Let \mathcal{A} be the event that $\hat{\Gamma} = \Gamma^*$, and let \mathcal{B} be the event that $0 \leq R(\hat{\Theta}) - R(\Theta^*) \leq \sqrt{m\rho \log n/n}$. Then $\mathbb{P}(\mathcal{B}) \geq \mathbb{P}[\mathcal{B}|\mathcal{A}] \cdot \mathbb{P}(\mathcal{A}) \geq (1 - 2p^2 \cdot \exp(-c_2 n(\alpha/4 - c_3 r/\sqrt{n})^2)) \cdot (1 - c_1 n^{-1})$, which has the lower bound $1 - c_1 n^{-1} - 2p^2 \cdot \exp(-c_2 n(\alpha/4 - c_3 r/\sqrt{n})^2)$. The proof concludes by recalling that $R(\Theta^*) = R(\psi_x)$ by (4.1). \square

The value of α is not always known *a priori*, and so we consider an alternative threshold that does not use the value of α . The downside of this alternative is that the results must necessarily be asymptotic because when $t_n > \alpha$, we cannot lower bound the probability of choosing the correct partition using the bounds from Proposition 3.7, since these bounds only ensure that the estimation error lies within an interval.

THEOREM 4.3. *If **A1–A5** are satisfied, $t_n = c_4/\sqrt{\log n}$ where $c_4 > 0$ is a constant, $r = O(1)$, and $\log p = o(n/\log n)$; then $R(\hat{\Theta}) - R(\psi_x) = O_p(\sqrt{m\rho \log n/n})$.*

Proof. The proof is roughly the same as the proof of Theorem 4.2, and so we highlight the main differences. Since t_n is strictly decreasing, there is some N such that $t_n < \alpha/2$ for all $n \geq N$. For the remaining arguments in the proof, we will assume $n \geq N$. Next, note that the mistakes we can make are: Either (i) $\mathcal{G}_{jq} = 0$ but $\hat{\mathcal{G}}_{jq} > t_n$, or (ii) $\mathcal{G}_{jq} \geq \alpha$ but $\hat{\mathcal{G}}_{jq} \leq t_n$. Restated, a type (i) error does not occur if $|\hat{\mathcal{G}}_{jq} - \mathcal{G}_{jq}| < t_n$, and a type (ii) error does not occur if $|\hat{\mathcal{G}}_{jq} - \mathcal{G}_{jq}| < \alpha/2$. As a result, Proposition 3.7 implies $\mathbb{P}(\max_{(j,q) \in \mathcal{J}} |\hat{\mathcal{G}}_{jq} - \mathcal{G}_{jq}| < t_n) \geq 1 - 2p^2 \cdot \exp(-c_2 n(\frac{1}{2}c_4/\sqrt{\log n} - c_3 r/\sqrt{n})^2)$. And so $\mathbb{P}(\mathcal{B}) \geq \mathbb{P}[\mathcal{B}|\mathcal{A}] \cdot \mathbb{P}(\mathcal{A}) \geq (1 - 2p^2 \cdot \exp(-c_2 n(\frac{1}{2}c_4/\sqrt{\log n} - c_3 r/\sqrt{n})^2))(1 - c_1 n^{-1})$, which leads to the desired result. \square

4.4. Comparison of Statistical Convergence Rates. These results imply we need $O((m\rho)^{1+\zeta})$, for any $\zeta > 0$, measurements to ensure $|R(\hat{\Theta}) - R(\psi_x)| = O_p(1)$. Reshaping the tensor (as in [42]) may potentially improve this statistical rate, but this requires further study that is beyond the scope of the present paper. Recall that existing convex optimization-based methods need $O(\pi^{\lfloor p/2 \rfloor} r^{\lceil p/2 \rceil})$ measurements whereas an NP-hard formulation needs $O(\pi^p + r\pi p)$ points [42]; note $\rho \geq \text{rank}_{\oplus}(\cdot) \geq \pi = \max_j \text{rank}_{\boxplus}(\cdot)$ [16], but these bounds are not tight [33]. It is difficult to compare approaches because we use low effective dimension as a surrogate for tensor rank, while many existing methods use low multilinear rank [56, 55, 22, 37, 42, 62].

However, we can make a direct comparison in the special case of rank-1 tensors (where $\pi = \text{rank}_{\boxplus}(\cdot) = 1$). In this case, existing convex optimization-based approaches need $O(r^{\lfloor p/2 \rfloor})$ measurements while our approach only needs $O((rp^2)^{1+\zeta})$, for any $\zeta > 0$, measurements (since $m \leq p$). Restated, our approach requires a poly-

nomial in p number of measurements, while existing approaches need an exponential in p number of measurements. Also, our approach is essentially a quadratic factor away from the NP-hard formulation, which needs $O(rp)$ measurements in this case. Methods based on the tensor nuclear norm need $O(rp^{p/2})$ measurements [60, 61], while methods based on the higher-order power method [57] can achieve the information theoretic limit $O(rp)$ whenever the algorithm converges to a global optimum.

4.5. Approximate Low-Rank Structure. So far we have assumed the ideal partition Γ^* for ψ has low effective dimension $\rho(\Gamma^*)$ and consequently describes a tensor with low rank; however, it is common to study estimation procedures for instances with approximate low-rank or sparsity structures (e.g., [6, 12]). Unfortunately, it is unclear how to define approximate low-rank structure for the class of tensors we consider. The difficulty is that our procedure works by exactly estimating the ideal partition Γ^* , but if a tensor approximately has low effective dimension then we would need to estimate an approximate partition $\bar{\Gamma}$. However, partitions are discrete and so there is no clear notion of approximation.

Given these ambiguities with defining approximate low-rank structure, we consider a related notion: We will estimate tensors with low bias that are also low rank and have low effective dimension. There is a tradeoff inherent in this between the amount of bias and the rank of the tensor. Smaller bias will lead to higher rank tensors, while larger bias will lead to lower rank tensors. It is difficult to analytically answer the question of how to control this tradeoff, and so instead we describe a cross-validation approach that can be used to control this.

The challenge with cross-validation is that we will need to control our effective dimension ρ ; otherwise the cross-validation error will not be an accurate estimate of the actual loss. We will create a finite sequence of nested partitions $\Gamma^1 \subset \Gamma^2 \subset \dots \subset \Gamma^q$, where $\Gamma^j \subset \Gamma^{j+1}$ denotes that $F \in \text{faces}(\Gamma^{j+1})$ whenever $F \in \text{facets}(\Gamma^j)$. The nested partitions will be constructed using a set of thresholds $T = \{t_1, t_2, \dots, t_q\}$, and we will use cross-validation to pick the threshold. Note that in general some subset of partitions may be equivalent (i.e., there may be j such that $\Gamma^j = \Gamma^{j+1}$).

For simplicity, we will consider leave- k -out cross-validation with $k = n/2$. The corresponding tensor completion algorithm using cross-validation is:

1. For each $t_j \in T$, do the following:
 - (a) Apply the algorithm from §4.3 to the full data set $(x\langle i \rangle, y\langle i \rangle)$ for $i = 1, \dots, n$, to estimate the risk gaps $\hat{\mathcal{G}}_{jq}(t_j)$ and partitions $\Gamma^j = \hat{\Gamma}(t_j)$.
 - (b) Use the data $(x\langle i \rangle, y\langle i \rangle)$ for $i = (\lfloor n/2 \rfloor + 1), \dots, n$ to compute estimates $\hat{\Theta}(t)$ by solving (3.7) with the partition $\hat{\Gamma}(t)$ and then inverting the mapping $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$.
 - (c) Compute the empirical cross-validation error $\hat{V}(t)$, which is defined as
$$\hat{V}(t) = \frac{1}{\lfloor n/2 \rfloor} \sum_{i=1}^{\lfloor n/2 \rfloor} (-y\langle i \rangle) \cdot \sum_{k=1}^m \log \tilde{\theta}(t)_{\mathcal{X}_k\langle i \rangle} + \prod_{k=1}^m \tilde{\theta}(t)_{\mathcal{X}_k\langle i \rangle}.$$
2. Set $\hat{t} = \arg \min\{\hat{V}(t) \mid t \in T\}$ to be the threshold selected by cross-validation.
3. Use the full data set $(x\langle i \rangle, y\langle i \rangle)$ for $i = 1, \dots, n$, to compute the final estimate $\hat{\Theta}(\hat{t})$ by solving (3.7) with the partition $\hat{\Gamma}(\hat{t})$ and then inverting the mapping $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$.

Suppose that $t^* = \arg \min\{R(\hat{\Theta}(t)) \mid t \in T\}$ is the optimal threshold. The following theorem shows that we can achieve an oracle inequality [2] using leave- k -out cross-validation. Note that we do not assume **A4, A5** hold.

THEOREM 4.4. *If **A1–A3** are satisfied, then with probability at least $1 - 10c_1 n^{-1}$. ($\#T$) we have $R(\hat{\Theta}(\hat{t})) - R(\psi_x) \leq R(\hat{\Theta}(t^*)) - R(\psi_x) + (4\sqrt{2} + 2)\sqrt{m_q \rho_q \log n/n}$, where*

$m_q = m(\Gamma^q)$, $\rho_q = \rho(\Gamma^q)$, and constant $c_1 > 0$ depends on μ, M .

Proof. Observe that we must have

$$(4.4) \quad R(\hat{\Theta}(\hat{t})) - R(\psi_x) \leq R(\hat{\Theta}(t^*)) - R(\psi_x) + |R(\hat{\Theta}(\hat{t})) - R(\hat{\Theta}(t^*))|,$$

and so applying the triangle inequality to the second term gives

$$(4.5) \quad |R(\hat{\Theta}(\hat{t})) - R(\hat{\Theta}(t^*))| \leq |R(\hat{\Theta}(\hat{t})) - \hat{V}(\hat{t})| + |\hat{V}(\hat{t}) - R(\hat{\Theta}(t^*))|.$$

We will deal with the two terms on the right separately.

Applying the triangle inequality to the first term of (4.5) gives $|R(\hat{\Theta}(\hat{t})) - \hat{V}(\hat{t})| \leq |R(\hat{\Theta}(\hat{t})) - R(\Theta^*(\hat{t}))| + |R(\Theta^*(\hat{t})) - R(\tilde{\Theta}(\hat{t}))| + |R(\tilde{\Theta}(\hat{t})) - \hat{V}(\hat{t})|$. The first two terms are bounded by [Theorem 3.8](#), and the third term is bounded by [Proposition 3.7](#). So if we let $w_n = \sqrt{m_q \rho_q \log n/n}$, then using the union bound twice (once for having three terms and once for having multiple $t \in T$) gives

$$(4.6) \quad |R(\hat{\Theta}(\hat{t})) - \hat{V}(\hat{t})| < (2\sqrt{2} + 1) \cdot w_n,$$

with probability at least $1 - 5c_1 n^{-1} \cdot (\#T)$.

For the second term of (4.5), the triangle inequality gives $|\hat{V}(\hat{t}) - R(\hat{\Theta}(t^*))| \leq |\hat{V}(\hat{t}) - R(\tilde{\Theta}(t^*))| + |R(\tilde{\Theta}(t^*)) - R(\Theta^*(t^*))| + |R(\Theta^*(t^*)) - R(\hat{\Theta}(t^*))|$. The last two terms are bounded by [Theorem 3.8](#), and so we focus on the first term. Because \hat{t} minimizes $\hat{V}(t)$, we have

$$(4.7) \quad \hat{V}(\hat{t}) \leq \hat{V}(t^*) \Rightarrow \hat{V}(\hat{t}) - R(\tilde{\Theta}(t^*)) \leq \hat{V}(t^*) - R(\tilde{\Theta}(t^*)).$$

Similarly, because t^* is the minimizer of $R(\tilde{\Theta}(t))$, we have

$$(4.8) \quad R(\tilde{\Theta}(t^*)) \leq R(\tilde{\Theta}(\hat{t})) \Rightarrow \hat{V}(\hat{t}) - R(\tilde{\Theta}(\hat{t})) \leq \hat{V}(\hat{t}) - R(\tilde{\Theta}(t^*)).$$

Combining (4.7) and (4.8) leads to $|\hat{V}(\hat{t}) - R(\tilde{\Theta}(\hat{t}))| \leq \max_{t \in T} |\hat{V}(t) - R(\tilde{\Theta}(t))|$, which can be bounded by [Proposition 3.7](#). As a result, the union bound gives the following

$$(4.9) \quad |\hat{V}(\hat{t}) - R(\hat{\Theta}(t^*))| < (2\sqrt{2} + 1) \cdot w_n,$$

with probability at least $1 - 5c_1 n^{-1} \cdot (\#T)$. The result follows by using the union bound to combine (4.4), (4.5), (4.6), and (4.9). \square

The lower bound on the success probability depends on the number of tuning parameters (via the cardinality of T), which is consistent with empirical results where using many tuning parameters leads to overfitting [45]. Another note is we must control the decomposition complexity (by ensuring that ρ_q is sufficiently small relative to n) to guarantee the above oracle inequality is achieved. Lastly, this result implies that the cross-validation procedure is *efficient* (in the sense of [2]) when $\#T$ and ρ_q grow sufficiently slowly in relation to n .

5. Sparsity in Hierarchical Decompositions. Sparsity in the tensor ψ_x can be used to improve the performance of our methods. Here, sparsity means parameters $\theta_{\mathcal{X}_k}$ that are equal to 1, because this corresponds to a parameter not influencing the tensor value ψ_x . In particular, we define a best sparse hierarchical decomposition as

$$(5.1) \quad \hat{\Theta} = \arg \min \{ \hat{R}(\Theta) \mid \Theta \in \Omega, \sum_{k=1}^m \sum_{\mathcal{X}_k \in \mathcal{R}_k} |\log \theta_{\mathcal{X}_k}| \leq \lambda \}.$$

The convex reparametrization is

$$(5.2) \quad \hat{U} = \arg \min \{ \hat{R}(U) \mid U \in \Phi, \sum_{k=1}^m \sum_{\mathcal{X}_k \in \mathcal{R}_k} |u_{\mathcal{X}_k}| \leq \lambda \}.$$

Our convex reformulation matches the normal notion of coefficient sparsity because sparsity means the $u_{\mathcal{X}_k} = \log \theta_{\mathcal{X}_k}$ are equal to 0. The $\sum_{k=1}^m \sum_{\mathcal{X}_k \in \mathcal{R}_k} |u_{\mathcal{X}_k}| \leq \lambda$ constraint in the convex reformulation is just an ℓ_1 -norm inequality, and so it can be represented using a linear in ρ number of linear inequalities using an LP lift [59]. Moreover, we can still solve this convex formulation in polynomial time; the proof is nearly identical to that of Proposition 3.4, and so it is not included here.

The key result regarding the above sparsity-exploiting formulations is an extension of Proposition 3.7, from which we can then prove results analogous to those above for decomposition (Theorem 3.9), approximation (Theorem 3.10), and completion (Theorems 4.2, 4.3, and 4.4) of positive tensors. We will not belabor this point by explicitly including these corresponding results or their proofs. We instead prove only this key result on concentration of the empirical loss when Θ satisfies the following additional constraint: $\sum_{k=1}^m \sum_{\mathcal{X}_k \in \mathcal{R}_k} |\log \theta_{\mathcal{X}_k}| \leq \lambda$.

PROPOSITION 5.1. *Under **A1–A3** and for any Γ , we have*

$$(5.3) \quad \mathbb{P}\left(\sup_{\Theta \in \Omega'} |\hat{R}(\Theta) - R(\Theta)| < t\right) \geq 1 - \exp\left(-C_3 n \left(t - C_4 \lambda \sqrt{\frac{\log \rho}{n}}\right)^2\right),$$

where $\Omega' = \{\Theta \in \Omega : \|\log \Theta\|_1 \leq \lambda\}$, and constants $C_3, C_4 > 0$ depend on μ, M .

Proof. The proof follows that of Proposition 3.7, and so we only highlight the differences. As before, we refer to the $\mathbb{1}_{\mathcal{X}_k = \mathcal{X}_k^{(i)}}$ as pseudo-predictors, and the $u_{\mathcal{X}_k}$ are still the parameters. If we define $\chi \in \{0, 1\}^\rho$ to be the vector of pseudo-predictors, then $\|\chi\|_1 = m$, $\|\chi\|_2 = \sqrt{m}$, and $\|\chi\|_\infty = 1$. The primary difference in this case is that the parameters belong to the modified set $\Phi' = \{U \in \Phi : \|U\|_1 \leq \lambda\}$. And so, results from [29] immediately give that $R(\mathbb{F}_W) \leq \lambda \sqrt{2 \log 2\rho/n}$. The result follows by combining this with (3.18) and (3.19). \square

This result shows that using soft-thresholding (via the $\|\log \Theta\|_1 \leq \lambda$ constraint) achieves performance that leverages the sparsity of the entries. In particular, the statistical convergence rate (i.e., the upper bounds in Theorems 3.9, 3.10, 4.2, 4.3, and 4.4) implied by the proposition depends on effective dimension ρ (rather than the total number of tensor entries $\prod r_i$) and on λ (rather than the ℓ_1 -norm of parameters that are upper-bounded by M). Expanding further, we would have the following convergence rates (i.e., the upper bounds in Theorems 3.9, 3.10, 4.2, 4.3, and 4.4) for positive tensor decomposition, approximation, and completion depending on the structure we leverage:

Structure	Convergence Rate
None	$O_p\left(\sqrt{\frac{r^p \log n}{n}}\right)$
Low Rank	$O_p\left(\sqrt{\frac{m\rho \log n}{n}}\right)$
Sparse	$O_p\left(\lambda \sqrt{\frac{\log r^p \log n}{n}}\right)$
Sparse + Low Rank	$O_p\left(\lambda \sqrt{\frac{\log \rho \log n}{n}}\right)$

There is an additional statistical implication of sparsity in our framework. Simultaneously regularizing for multiple sparse structures using convex approaches only regularizes with respect to the single most useful structure [48, 42]. In tensor completion, sparsity in the entries and the low-rank structure signify two distinct sparse structures. Our framework overcomes the limitation of using only convex approaches to regularize with respect to these two sparse structures by combining hard- and soft-

thresholding (similar to [15]). In particular, we estimate a partition $\hat{\Gamma}$ using hard-thresholding applied to empirical risk gaps $\hat{\mathcal{G}}_{jq}$; and so we can use soft-thresholding to exploit sparsity in the coefficients of the partition decomposition.

This combination of hard- and soft-thresholding allows our framework to handle other types of sparsity models. One class of models [24] is a low-rank tensor corrupted by a sparse additive perturbation. Our approach can estimate a similar model: In particular, consider the model of a low-rank tensor corrupted by a sparse multiplicative perturbation. Then, we can solve

$$(5.4) \quad (\hat{\Theta}, \hat{E}) = \arg \min \left\{ \frac{1}{n} \sum_{i=1}^n (-y\langle i \rangle \cdot (\log e_{\mathcal{X}\langle i \rangle} + \sum_{k=1}^m \log \theta_{\mathcal{X}_k\langle i \rangle}) + e_{\mathcal{X}\langle i \rangle} \cdot \prod_{k=1}^m \theta_{\mathcal{X}_k\langle i \rangle}) \mid \Theta \in \Omega, \sum_{\mathcal{X} \in \mathcal{R}} |\log e_{\mathcal{X}}| \leq \lambda \right\}.$$

The convex reparametrization is

$$(5.5) \quad (\hat{U}, \hat{W}) = \arg \min \left\{ \frac{1}{n} \sum_{i=1}^n (-y\langle i \rangle \cdot (w_{\mathcal{X}\langle i \rangle} + \sum_{k=1}^m u_{\mathcal{X}_k\langle i \rangle}) + \exp(w_{\mathcal{X}\langle i \rangle} + \sum_{k=1}^m u_{\mathcal{X}_k\langle i \rangle}) \mid U \in \Phi, \sum_{\mathcal{X} \in \mathcal{R}} |w_{\mathcal{X}}| \leq \lambda \right\},$$

where we have the equivalence relation $w_{\mathcal{X}} = \log e_{\mathcal{X}}$. We do not develop the theory for this (or other similar) models in this paper.

6. Numerical Example. We compare our proposed estimators to three recent estimators for tensor completion. More specifically, we compare five approaches:

1. The first estimator (*Square Nuclear Norm* method) [42] is given by $\hat{\psi} = \arg \min_{\phi} \left\{ \frac{1}{n} \sum_{i=1}^n (y\langle i \rangle - \phi_{x\langle i \rangle})^2 \mid \|\text{reshape}(\phi_{(1)}, \prod_{j=1}^s r_j, \prod_{j=s+1}^p r_j)\|_* \leq \lambda \right\}$, where $\phi_{(1)}$ is the unfolding of ϕ (into a matrix) along the first dimension [16, 42], the value s minimizes $|\prod_{j=1}^s r_j - \prod_{j=s+1}^p r_j|$, $\text{reshape}(T, n_1, n_2)$ is a function that reshapes a matrix T to have n_1 rows and n_2 columns, and $\lambda > 0$ is a constant.

2. The second estimator (*Maximum Nuclear Norm* method) [62] is given by $\hat{\psi} = \min_{\phi} \left\{ \frac{1}{n} \sum_{i=1}^n (y\langle i \rangle - \phi_{x\langle i \rangle})^2 \mid \max_j \{\|\phi_{(j)}\|_*\} \leq \lambda \right\}$, where $\phi_{(j)}$ is the unfolding of ϕ along the j -th dimension, and $\lambda > 0$ is a constant.

3. The third estimator (*Alternating Least Squares* method) [31] identifies a best CP decomposition [31, 27] by solving $\hat{\psi} = \min_{\phi} \left\{ \frac{1}{n} \sum_{i=1}^n (y\langle i \rangle - \phi_{x\langle i \rangle})^2 \mid \phi = \sum_{j=1}^q v_1^j \otimes \cdots \otimes v_p^j \right\}$ using alternating least squares (ALS). This is an ordinary least squares (OLS) problem in the variables v_w^j , for all $j \in [q]$, when the v_k^j , for all $k \in [p] \setminus w \wedge j \in [q]$, are fixed. ALS minimizes this objective by iterating the index w between $1, \dots, p$ and solving the resulting OLS problems. Though ALS is the most common approach for computing tensor decompositions [31], it typically converges to a local minimum [31].

4. The fourth estimator (*Partition Log-Linear* method) is our tensor completion algorithm with cross-validation from §4.5 and with the estimator (3.2).

5. The fifth estimator (*Sparse Partition Log-Linear* method) is our tensor completion algorithm with cross-validation from §4.5 and with the estimator (5.1).

The versions of the nuclear norm estimators we use are different from those in [42, 62] because here we measure tensor entries with noise; the versions presented in [42, 62] deal with the noiseless case. Because we have noise, we instead minimize the deviation between measurements and estimates subject to a constraint that the nuclear norms are not large. This is a common formulation for the noisy case of sparse estimation problems (see for instance [47, 26, 11]). Also, we use a variant of the Maximum Nuclear Norm with simpler computation than [62], which converts the maximum into a smooth formulation that is amenable to specialized algorithm design.

Gamma Distribution $k = 1, \theta = 1$ (with Variance 1)						
	n					
	10	50	100	500	1000	5000
Square Nuclear Norm	9.28	7.96	6.44	2.10	1.12	0.24
Maximum Nuclear Norm	9.29	8.51	7.44	2.58	1.61	0.68
Alternating Least Squares	54.10	10.43	4.00	1.39	1.16	0.11
Partition Log-Linear	16.46	4.17	2.43	0.35	0.16	0.03
Sparse Partition Log-Linear	4.67	3.13	2.60	0.31	0.16	0.03

Gamma Distribution $k = \frac{1}{5}, \theta = 5$ (with Variance 5)						
	n					
	10	50	100	500	1000	5000
Square Nuclear Norm	9.31	9.27	9.23	6.18	4.40	1.03
Maximum Nuclear Norm	9.31	9.29	9.25	6.71	4.25	1.46
Alternating Least Squares	278.78	78.10	49.47	3.65	1.92	1.12
Partition Log-Linear	15.52	8.72	7.03	2.18	1.31	0.16
Sparse Partition Log-Linear	5.10	4.08	3.90	2.67	1.27	0.16

TABLE 1
Median of Average Estimation Error Over 100 Trials

Numerical implementations of the five estimators have been made available¹. We implemented (i) our (Sparse) Partition Log-Linear method using the MATLAB toolbox for MOSEK [41], (ii) the Square and Maximum Nuclear Norm methods using the CVX package [25] for MATLAB, and (iii) the Alternating Least Squares method using MATLAB. Our implementation code is not optimized for speed, and we have not studied the choice of algorithms for solving the convex reformulation of our estimators. However, our estimators compute quickly because there are no constraints on matrix positive semidefiniteness (unlike the methods using nuclear norm). We observed that our estimators computed faster than the nuclear norm estimators, but we do not include benchmarks because optimized code was not used.

The first numerical example consists of synthetic data generated from the tensor

$$(6.1) \quad \psi = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

at two different noise levels, and we examine the estimation error as the amount of data increases for a fixed model. The random variable $(1+z)$ has gamma distribution with shape $k > 0$ and scale $\theta > 0$. A gamma distribution is used for the noise because it has support over $[0, \infty)$, and can be specified to ensure $\mathbb{E}(z) = 0$ as required by **A2**. Though the unbounded support technically violates the assumption in **A2** on the boundedness of the noise, this boundedness is not a crucial assumption and can be relaxed (as we discussed earlier). The numerical results support this conclusion.

Entries of the tensor are measured uniformly, and we used leave- k -out cross-validation with $k = n/2$ to select the tuning parameters of the different approaches. Results for 100 repeated simulations are shown in Table 1. The table reports average prediction error under a square loss $\mathcal{E} = (\prod r_i)^{-1} \sum_{x \in \mathcal{R}} (\psi_x - \hat{\psi}_x)^2$. The results indicate that our estimation procedure is competitive with existing approaches to tensor completion. For each scenario, either the Partition Log-Linear or Sparse Partition Log-Linear approach has the lowest estimation error.

¹<http://ieor.berkeley.edu/~aaswani/plrt/>

7. Regression with Categorical Variables. We refer to a model with purely categorical predictors and a numeric response as a *combinatorial regression model*. In particular, suppose there are p categorical predictors. For the j -th predictor with r_j different categories, we can assign each category to a unique integer in $[r_j]$. With this notation, a combinatorial regression model can be written as $\psi_x := \mathbb{E}[y|x] = \theta_{x_1, x_2, \dots, x_p}$, where $x_j \in [r_j]$ is the j -th predictor, and $\theta : [r_1] \times \dots \times [r_p] \rightarrow \mathbb{R}$ is a function from a discrete to continuous space.

Even without knowledge about θ , we can still identify the model because of the finiteness of the predictor space. Specifically, if we have measurements $(x\langle i \rangle, y\langle i \rangle)$ for $i = 1, \dots, n$, then we can identify the model using $\hat{\psi}_x = \frac{1}{|\mathcal{I}(x)|} \sum_{i \in \mathcal{I}(x)} y\langle i \rangle$, where $\mathcal{I}(x) = \{i : x\langle i \rangle = x\}$. Under typical assumptions on noise, this is a consistent estimator. Unfortunately, its convergence rate $O_p(r^p/n)$, where $r = \max r_j$, is exponentially slow in p ; this is not surprising because there is a combinatorial explosion that leads to a curse of dimensionality if we try to estimate each value of ψ_x separately.

The standard approach to reducing dimensionality is to (i) define *coding variables* (e.g., dummy predictors) to convert categorical variables into numerical values, and (ii) perform regression using the coding variables [14]. Though this converges at $O_p(rp/n)$, this can be restrictive because the impact of different predictors x_j, x_k for $j \neq k$ is completely decoupled, which is not reflective of a combinatorial model. (In principal, variables can be coupled by defining pairwise (or higher) coding variables, but this is typically done using domain knowledge.)

Our notation for a combinatorial regression model is suggestive of another interpretation of low-rank structure: We propose the novel interpretation that a combinatorial regression model can be represented by a low-rank tensor. The tensor is indexed by the x_j , which are integers. Thus, the problem of estimating a combinatorial regression problem is equivalent to a noisy low-rank tensor completion problem. Given the discussion of tensor completion in this paper, we can consider an example of identifying a combinatorial regression model with real data using tensor completion.

7.1. Violacein Pathway. Bioengineered metabolic pathways hold promise for the production of pharmaceuticals and transportation fuels, and they are constructed in a combinatorial fashion by varying different discrete design elements. This combinatorial nature makes it challenging to engineer the pathway to maximize production of the bioproduct, and so one proposed idea is to (i) construct a model relating design parameters to the amount of bioproduct produced, and then (ii) use this model to determine which combination of design elements maximizes the bioproduct [35].

Bioengineered pathways can be represented by a combinatorial regression model, and so it is instructive to apply tensor completion methods. In the pathway studied in [35], there are five predictors $p = 5$, and each predictor has five levels $r_j = 5$ for $j = 1, \dots, 5$. The data is categorized into either a training data set or a validation data set, and each respective data set consists of different experiments with explicitly different predictor values (i.e., design elements) used for each; the validation data set was constructed to be a true validation data set for the original model in [35].

A comparison of predicted and measured values for models computed using different approaches is shown in Figure 2. Sparse Partition Log-Linear is not separately shown because cross-validation chose λ to make the model identical to Partition Log-Linear. All models were constructed using data designated as the training set in [35], and the predictions and measured values in Figure 2 correspond to data designated as the validation set in [35]. The equipment could not measure values smaller than 0.5, and so measured values and model predictions smaller than this were set to 0.5.

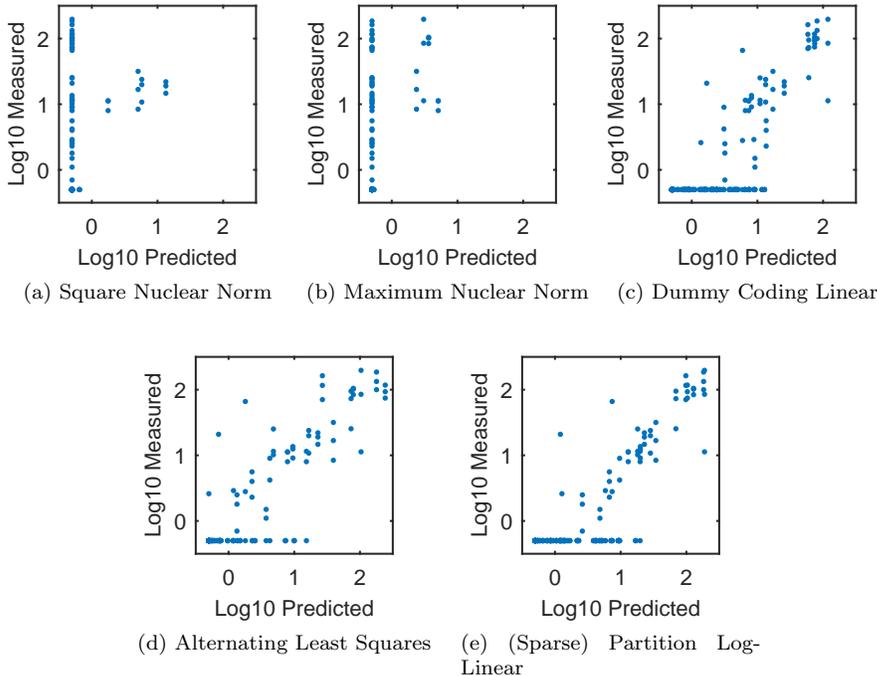


FIG. 2. Comparison between predicted and measured violacein production levels.

The predictions of (Sparse) Partition Log-Linear most closely match the measured values. Dummy Coding Linear (i.e., model in [35] using [3]) and Alternating Least Squares perform less well, and Square Nuclear Norm and Maximum Nuclear Norm do not work well for this data. Quantitatively, Spearman’s rank correlation coefficient is interesting because we are interested in models that can predict the relative amount of bioproduct for a particular combinatorial design. The Spearman correlation coefficient (for measurements above the minimum detectable level of 0.5) is 0.84 for (Sparse) Partition Log-Linear, 0.80 for Alternating Least Squares, 0.75 for Dummy Coding Linear, 0.17 for Maximum Nuclear Norm, and -0.26 for Square Nuclear Norm. We also conducted a bias-corrected bootstrap hypothesis test [19] to determine whether the model fit improvement of (Sparse) Partition Log-Linear (as compared to a model computed by the other methods) was statistically significant: This hypothesis test returned ($p=0.009$) for Alternating Least Squares, ($p<0.001$) for Dummy Coding Linear, ($p<0.001$) for Maximum Nuclear Norm, and ($p<0.001$) for Square Nuclear Norm. These results indicate the model fit improvement of (Sparse) Partition Log-Linear is statistically significant.

8. Conclusion. We defined a new decomposition for positive tensors, showed it can be computed in polynomial time using a randomized algorithm, and justified the design of this decomposition and its loss function by identifying important cases where they coincide with the usual tensor decomposition and squared loss function. We extended this framework to tensor completion, and showed our approach has improved statistical performance in comparison to existing approaches. We provided a novel interpretation of regression problems with categorical variables as tensor completion

problems, and numerical examples with synthetic data and data from a bioengineered metabolic network displayed the improved performance of our approach. Our current work includes application of our approach to regression with categorical variables on larger data sets, and we have found our approach computationally scales well in terms of the number of data points n , tensor order p , and dimension r . However, scalability issues arise as the effective dimension $\rho(\Gamma) = \sum_{k=1}^m \prod_{j \in F_k} r_j$ (which has exponential size in terms of the cardinality of F_k) grows. It may be interesting to develop specialized optimization algorithms to solve our formulation in such settings.

Acknowledgements. The author thanks John E. Dueber and Michael E. Lee for providing the violacein data set [35].

REFERENCES

- [1] A. AGARWAL, S. NEGAHBAN, AND M. WAINWRIGHT, *Noisy matrix decomposition via convex relaxation: Optimal rates in high dimensions*, Ann. Stat., 40 (2012), pp. 1171–1197.
- [2] S. ARLOT AND A. CELISSE, *A survey of cross-validation procedures for model selection*, Statistics Surveys, 4 (2010), pp. 40–79.
- [3] A. ASWANI, P. BICKEL, AND C. TOMLIN, *Regression on manifolds: Estimation of the exterior derivative*, Ann. Stat., 39 (2011), pp. 48–81.
- [4] A. BANERJEE, X. GUO, AND H. WANG, *On the optimality of conditional expectation as a Bregman predictor*, IEEE Trans. Inf. Theor., 51 (2005), pp. 2664–2669.
- [5] P. BARTLETT AND S. MENDELSON, *Rademacher and gaussian complexities: Risk bounds and structural results*, J. Mach. Learn. Res., (2002).
- [6] P. BICKEL AND E. LEVINA, *Covariance regularization by thresholding*, Ann. Stat., 36 (2008), pp. 2577–2604.
- [7] S. BOUCHERON, G. LUGOSI, AND P. MASSART, *Concentration Inequalities: A Nonasymptotic Theory of Independence*, OUP Oxford, 2013.
- [8] S. BOYD AND L. VANDENBERGHE, *Convex Optimization*, Cambridge University Press, 2004.
- [9] J.-F. CAI, E. CANDÈS, AND Z. SHEN, *A singular value thresholding algorithm for matrix completion*, SIAM J. Optim., 20 (2010), pp. 1956–1982.
- [10] E. CANDÈS, X. LI, Y. MA, AND J. WRIGHT, *Robust principal component analysis?*, J. ACM, 58 (2011), pp. 11:1–11:37.
- [11] E. CANDÈS AND Y. PLAN, *Matrix completion with noise*, Proc. IEEE, 98 (2010), pp. 925–936.
- [12] S. CHATTERJEE, *Matrix estimation by universal singular value thresholding*, Ann. Stat., 43 (2015), pp. 177–214.
- [13] E. CHI AND T. KOLDA, *On tensors, sparsity, and nonnegative factorizations*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1272–1299.
- [14] J. COHEN, P. COHEN, S. WEST, AND L. AIKEN, *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*, Taylor & Francis, 2013.
- [15] A. D’ASPREMONT, L. EL GHAOUI, M. JORDAN, AND G. LANCKRIET, *A direct formulation of sparse PCA using semidefinite programming*, SIAM Review, 49 (2007).
- [16] V. DE SILVA AND L.-H. LIM, *Tensor rank and the ill-posedness of the best low-rank approximation problem*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1084–1127.
- [17] P. DIACONIS AND B. STURMFELS, *Algebraic algorithms for sampling from conditional distributions*, Ann. Stat., 26 (1998), pp. 363–397.
- [18] M. DRTON, B. STURMFELS, AND S. SULLIVANT, *Lectures on Algebraic Statistics*, Birkhäuser, 2009.
- [19] B. EFRON, *Better bootstrap confidence intervals*, Journal of the American statistical Association, 82 (1987), pp. 171–185.
- [20] M. FAZEL, H. HINDI, AND S. BOYD, *A rank minimization heuristic with application to minimum order system approximation*, in Proceedings of the American Control Conference, vol. 6, IEEE, 2001, pp. 4734–4739.
- [21] S. FRIEDLAND AND L.-H. LIM, *Computational complexity of tensor nuclear norm*, arXiv:1410.6072, (2014).
- [22] S. GANDY, B. RECHT, AND I. YAMADA, *Tensor completion and low-n-rank tensor recovery via convex optimization*, Inverse Problems, 27 (2011), p. 025010.
- [23] M. GAVISH AND D. DONOHO, *The optimal hard threshold for singular values is $(4/\sqrt{3})$* , IEEE Trans. Inf. Theor., 60 (2014), pp. 5040–5053.

- [24] D. GOLDFARB AND Z. QIN, *Robust low-rank tensor recovery: Models and algorithms*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 225–253.
- [25] M. GRANT AND S. BOYD, *Graph implementations for nonsmooth convex programs*, in Recent Advances in Learning and Control, Springer-Verlag Limited, 2008, pp. 95–110.
- [26] E. GREENSHTEIN AND Y. RITOV, *Persistence in high-dimensional linear predictor selection and the virtue of overparametrization*, Bernoulli, 10 (2004), pp. 971–988.
- [27] C. HILLAR AND L.-H. LIM, *Most tensor problems are np-hard*, J. ACM, 60 (2013), pp. 45:1–45:39.
- [28] P. JAIN AND S. OH, *Provable tensor factorization with missing data*, in Advances in Neural Information Processing Systems, 2014, pp. 1431–1439.
- [29] S. KAKADE, K. SRIDHARAN, AND A. TEWARI, *On the complexity of linear prediction: Risk bounds, margin bounds, and regularization*, in Advances in Neural Information Processing Systems, 2009, pp. 793–800.
- [30] R. KESHAVAN, A. MONTANARI, AND S. OH, *Matrix completion from a few entries*, IEEE Trans. Inf. Theor., 56 (2010), pp. 2980–2998.
- [31] T. KOLDA AND B. BADER, *Tensor decompositions and applications*, SIAM Review, 51 (2009), pp. 455–500.
- [32] J. LAFFERTY, H. LIU, AND L. WASSERMAN, *Statistical Machine Learning*, 2010. In preparation.
- [33] J. LANDSBERG, *Tensors: Geometry and Applications*, Graduate studies in mathematics, American Mathematical Society, 2012.
- [34] M. LEDOUX AND M. TALAGRAND, *Probability in Banach Spaces: Isoperimetry and Processes*, Springer, 1991.
- [35] M. LEE, A. ASWANI, A. HAN, C. TOMLIN, AND J. DUEBER, *Expression-level optimization of a multi-enzyme pathway in the absence of a high-throughput assay*, Nucleic Acids Res., (2013).
- [36] L.-H. LIM AND P. COMON, *Nonnegative approximations of nonnegative tensors*, J. Chemometrics, 23 (2009), pp. 432–441.
- [37] J. LIU, P. MUSIALSKI, P. WONKA, AND J. YE, *Tensor completion for estimating missing values in visual data*, IEEE Trans. Pattern Anal. Mach. Intell., 35 (2013), pp. 208–220.
- [38] S. MA, D. GOLDFARB, AND L. CHEN, *Fixed point and bregman iterative methods for matrix rank minimization*, Math. Prog., 128 (2011), pp. 321–353.
- [39] B. MANTHEY AND H. RÖGLIN, *Worst-case and smoothed analysis of k-meansclustering with Bregman divergences*, in Algorithms and Computation, Springer, 2009, pp. 1024–1033.
- [40] A. MONTANARI AND E. RICHARD, *A statistical model for tensor pca*, arXiv preprint arXiv:1411.1076, (2014).
- [41] MOSEK APS, *The MOSEK optimization toolbox for MATLAB manual. Version 7.1*, 2015.
- [42] C. MU, B. HUANG, J. WRIGHT, AND D. GOLDFARB, *Square deal: Lower bounds and improved relaxations for tensor recovery*, in Proceedings of the International Conference on Machine Learning, 2014, pp. 73–81.
- [43] A. NEMIROVSKI, *Interior point polynomial time methods in convex programming*, tech. report, Georgia Institute of Technology, 2004.
- [44] Y. NESTEROV AND A. NEMIROVSKII, *Interior-Point Polynomial Algorithms in Convex Programming*, Society for Industrial and Applied Mathematics, 1994.
- [45] A. NG, *Preventing “overfitting” of cross-validation data*, in Proceedings of the International Conference on Machine Learning, 1997, pp. 245–253.
- [46] J. NIE AND L. WANG, *Semidefinite relaxations for best rank-1 tensor approximations*, SIAM Journal on Matrix Analysis and Applications, 35 (2014), pp. 1155–1179.
- [47] M. OSBORNE, B. PRESNELL, AND B. TURLACH, *On the lasso and its dual*, J. Comp. Graph. Stat., 9 (2000), pp. 319–337.
- [48] S. OYMAK, A. JALALI, M. FAZEL, Y. ELДАР, AND B. HASSIBI, *Simultaneously structured models with application to sparse and low-rank matrices*, IEEE Trans. Inf. Theor., 61 (2015), pp. 2886–2908.
- [49] Y. QI, P. COMON, AND L.-H. LIM, *Uniqueness of nonnegative tensor approximations*, arXiv preprint arXiv:1410.8129, (2014).
- [50] H. RAUHUT AND Ž. STOJANAC, *Tensor theta norms and low rank recovery*, arXiv:1505.05175, (2015).
- [51] B. RECHT, M. FAZEL, AND P. PARRILO, *Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization*, SIAM Review, 52 (2010), pp. 471–501.
- [52] J. RENEGAR, *A Mathematical View of Interior-Point Methods in Convex Optimization*, SIAM J. Matrix Anal. Appl., 2001.
- [53] R. ROCKAFELLAR AND R. WETS, *Variational Analysis*, Springer, 2009.
- [54] J. SAUNDERSON, V. CHANDRASEKARAN, P. PARRILO, AND A. WILLSKY, *Diagonal and low-rank*

- matrix decompositions, correlation matrices, and ellipsoid fitting*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1395–1416.
- [55] M. SIGNORETTO, L. DE LATHAUWER, AND J. SUYKENS, *Nuclear norms for tensors and their use for convex multilinear estimation*, Tech. Report 10-186, ESAT, K.U.Leuven, 2010.
- [56] R. TOMIOKA, K. HAYASHI, AND H. KASHIMA, *Estimation of low-rank tensors via convex optimization*, arXiv preprint arXiv:1010.0789, (2010).
- [57] A. USCHMAJEV, *A new convergence proof for the higher-order power method and generalizations*, Pacific Journal Of Optimization, 11 (2015), pp. 309–321.
- [58] M. WELLING AND M. WEBER, *Positive tensor factorization*, Pattern Recognition Letters, 22 (2001), pp. 1255–1261.
- [59] M. YANNAKAKIS, *Expressing combinatorial optimization problems by linear programs*, J. Comput. Syst. Sci., 43 (1991), pp. 441–466.
- [60] M. YUAN AND C.-H. ZHANG, *On tensor completion via nuclear norm minimization*, Foundations of Computational Mathematics, (2015), pp. 1–38.
- [61] ———, *Incoherent tensor norms and their applications in higher order tensor completion*, arXiv:1606.03504, (2016).
- [62] X. ZHANG, Z. ZHOU, D. WANG, AND Y. MA, *Hybrid singular value thresholding for tensor completion*, in AAAI Conference on Artificial Intelligence, 2014.
- [63] H. ZOU, T. HASTIE, AND R. TIBSHIRANI, *Sparse principal component analysis*, J. Comp. Graph. Stat., 15 (2006), pp. 265–286.