When Can We Solve the Weighted Low Rank Approximation Problem in Truly Subquadratic Time?

Chenyang Li[1] **Yingyu Liang**[2,3] **Zhenmei Shi**[3] **Zhao Song**[4]

[1]Fuzhou University. [2]The University of Hong Kong. [3]University of Wisconsin-Madison.
[4]The Simons Institute for the Theory of Computing at the University of California, Berkeley.

Abstract

The weighted low-rank approximation problem is a fundamental numerical linear algebra problem and has many applications in machine learning. Given a $n \times n$ weight matrix W and a $n \times n$ matrix A, the goal is to find two low-rank matrices $U, V \in \mathbb{R}^{n \times k}$ such that the cost of $\|W \circ (UV^\top - A)\|_F^2$ is minimized. Previous work has to pay $\Omega(n^2)$ time when matrices A and W are dense, e.g., having $\Omega(n^2)$ non-zero entries. In this work, we show that there is a certain regime, even if A and W are dense, we can still hope to solve the weighted low-rank approximation problem in almost linear $n^{1+o(1)}$ time.

1 INTRODUCTION

Weighted low-rank approximation problem (Srebro and Jaakkola, 2003) is a fundamental numerical linear algebra problem related to matrix completion (Li et al., 2016b), faster SVD decomposition (Bhojanapalli et al., 2014, 2016), and principal component analysis (Xu et al., 2021). It is a prevalent focus over recent years and has been widely applied in machine learning in broad practical applications, such as vision detection (Chen et al., 2022), representation learning (Wang et al., 2019), image classification (Pu et al., 2023), language model (Hsu et al., 2021), weather prediction (Wilson et al., 2018), and many more. We can formulate the weighted low-rank approximation problem as below.

Definition 1.1. *Given two $n \times n$ matrices A and W, the goal is to find two $n \times k$ matrices U, V such that $\|W \circ (UV^\top - A)\|_F^2 \leq (1+\epsilon) \min_{\{A' | \mathrm{rank}(A') = k\}} \|W \circ$*

$(A' - A)\|_F^2$. *Here, $W \circ B$ denotes a matrix with entries given by $W_{i,j} B_{i,j}$, where $\circ$ is the Hadamard product operation.*

Here, we provide an example to show that attention computation can be formulated as a weighted low-rank approximation problem. Let $Q, K, V \in \mathbb{R}^{n \times d}$ be the query, key, and value matrix. Let $A := \mathsf{Activation}(QK^\top) \in \mathbb{R}^{n \times n}$ be the attention matrix, e.g, using $\mathsf{Softmax}$ as activation. Let $W \in \mathbb{R}^{n \times n}$ be the attention mask. Then, we have the attention output as $(A \circ W)V$. We can clearly see that $A \circ W$ is the target of the weighted low-rank approximation problem.

There is some well-known hardness in the weighted low-rank approximation problem. Gillis and Glineur (2011); Razenshteyn et al. (2016) showed that, when we want to find an approximate solution with $\epsilon = 1/\mathrm{poly}(n)$, the weighted low-rank approximation problem is "equivalent " to NP-hard problem. To practically fix it, under certain assumptions (e.g., the weighted matrix W has r distinct columns/rows for small r), we may have a polynomial time complexity algorithm to solve the weighted low-rank approximation problem. Thus, there are many theoretical and empirical works studying how we are able to solve the weighted low-rank approximation problem efficiently (e.g., Eriksson and Van Den Hengel (2012); Zheng et al. (2019); Song et al. (2023b); Woodruff and Yasuda (2024) and many more).

However, the best previous works still need to solve weighted low-rank approximation in $\Omega(n^2)$ if A has $\Omega(n^2)$ non-zero entries and W has $\Omega(n^2)$ non-zero entries. Therefore, in this work, we raise a natural fundamental mathematical question:

> *Is there some dense regime setting for A and W so that we can solve the weighted low-rank approximation problem in almost linear n time?*

The answer is positive. When we have mild practical assumptions (see detail in Assumption 3.2), we can solve the weighted low-rank approximation problem

in almost linear complexity, i.e., $n^{1+o(1)}$. We state our main results in the following, whose proof is in Section 7.

Theorem 1.2 (Main result). *Let A and W denote two $n \times n$ matrices. Assume each entry of A and W needs n^γ bits[1] to represent, where $\gamma = o(1)$. Let r be the number of distinct columns and rows of W (Definition 3.1). Assume $A \circ W$ has at most $r \cdot p$ distinct columns and at most $r \cdot p$ distinct rows, where $p = n^{o(1)}$. Assume that $k^2 r = O(\log n / \log \log n)$. For every constant $\epsilon > 0$, there is an algorithm running in time, $n^{1+o(1)}$ time which outputs a factorization (into an $n \times k$ and a $k \times n$ matrix) of a rank-k matrix $\widehat{A}$ for which*

$$\|W \circ (A - \widehat{A})\|_F^2 \leq (1 + \epsilon) \, \mathrm{OPT},$$

with probability at least 0.99, *where*

$$\mathrm{OPT} := \min_{\{A_k | \mathrm{rank}(A_k) = k\}} \|W \circ (A - A_k)\|_F^2.$$

Theorem 1.2 shows that, with high probability, we can solve the weighted low-rank approximation problem in almost linear complexity when there is a small number of distinct rows and columns in the target matrix. Furthermore, for any $\delta \in (0, 0.1)$, if we conduct $O(\log(1/\delta))$ repetitions and take the median of results as the final output, our success probability can be boosted to $1 - \delta$. Note that it is a standard way to boost the success probability from constant 0.99 to $1 - \delta$ for any $\delta \in (0, 0.1)$. The high-level idea is a majority vote, which is equivalent to taking the median.

Comparison with Previous Works. Our results are beyond Razenshteyn et al. (2016) in the following sense. Under the condition in Theorem 1.2, their algorithm requires $(\mathrm{nnz}(A) + \mathrm{nnz}(W)) \cdot n^{o(1)} + n^{1+o(1)}$ running time complexity, where $\mathrm{nnz}(A)$ means number of non-zero entries in A. When A or W are dense matrix, their complexity is $n^{2+o(1)}$, while our complexity is $n^{1+o(1)}$. We use a similar high-level proof framework as Razenshteyn et al. (2016). However, as our problem setup changes, all analyses need to be updated accordingly. In particular, Theorem 6.1 in Section 6 is brand new. The high-level intuition is that we need to carefully handle the distinct patterns in the analysis to avoid the quadratic time complexity. Then, during the matrix product, we can rewrite the summation order. Then, by fine-grained analysis, we can group many operations together due to the few distinct patterns and bypass the quadratic complexity.

---

[1]Each entry in a matrix can be represented by n^γ bits. In a real dataset, if we use the float32 format, then as long as $n^\gamma \geq 32$, our assumption holds.

Roadmap. In Section 2, we provide related work of weighted low-rank. In Section 3, we provide the preliminary of our problem setup and some tools from previous works. Then, in Section 4, we provide a warm up of our analysis to introduce our technique. In Section 5, we provide the lower bound of cost used in binary search. In Section 6, we introduce how to estimate the OPT value. In Section 7, we can successfully recover our solutions. Finally, we state our conclusion in Section 8.

2 RELATED WORK

2.1 Weighted Low Rank Applications

Weighted low-rank techniques have been widely applied in broad machine learning applications such as grayscale-thermal detection Li et al. (2016a), object detection Tang et al. (2016); Chen et al. (2022), fault detection Du et al. (2017), defect detection Mo et al. (2020); Ji et al. (2020), background estimation Dutta and Li (2017a); Dutta et al. (2018), multi-task learning Fan et al. (2018), robust visual representation learning Kulesza et al. (2015); Wen et al. (2018); Wang et al. (2019), adversarial learning Langenberg et al. (2019), image restoration Peng et al. (2014); Chang et al. (2020), image clustering and classification Shang et al. (2011); Wang and Liu (2017); Wu et al. (2018); Fu et al. (2021, 2022a,b); Khan et al. (2023); Pu et al. (2023), robust principal component analysis Xu et al. (2021), language models training Hua et al. (2022), language model compression Hsu et al. (2021), weather prediction Wilson et al. (2018), tensor training Chen et al. (2021); Zhang et al. (2022), domain generalization Shi et al. (2024a) and many more. Weighted low-rank techniques have also been widely used in signal processing for filter design and noise removal Lu et al. (1997); Li et al. (2010); Jiang et al. (2015).

2.2 Weighted Low Rank in Attention Mechanism

Particularly, a line of works shows that the attention matrix may have some low-rank property, even under softmax activation function by polynomial approximation methods, e.g., Alman and Song (2023); Ke et al. (2025); Liang et al. (2025); Chen et al. (2024); Li et al. (2024b); Liang et al. (2024a,b,c,d); Shi et al. (2024c); Xu et al. (2024); Alman and Song (2024a,b,c); Hu et al. (2024a,b,c,d). Thus, under such conditions, we may use weighted low-rank approximations for transformers' attention acceleration. For a few distinct columns or rows, empirically, we see that the attention matrix has some good patterns Jiang et al. (2024); Chen et al. (2025b); Li et al. (2025); Chen et al. (2025a); Shi et al.

Chenyang Li[1], Yingyu Liang[2,3], Zhenmei Shi[3], Zhao Song[4]

(2024b). In this work, we focus on the theoretical analysis and leave its empirical justification as future work.

2.3 Weighted Low Rank Approximation and Acceleration

Many previous works try to solve in an efficient way empirically and theoretically Manton et al. (2003); Schuermans et al. (2004); Werner and Jansson (2006); Markovsky and Van Huffel (2007); Markovsky (2008); Eriksson and Van Den Hengel (2012); Meng et al. (2013); Markovsky and Usevich (2014); Razenshteyn et al. (2016); Li et al. (2016b); Dutta (2016); Dutta and Li (2017a,b); Ban et al. (2019); Huang et al. (2019); Zheng et al. (2019); Sun et al. (2020); Tuzhilina and Hastie (2021); Yalcin et al. (2022); Song et al. (2023b); Zhang et al. (2024). Particularly, recently, Woodruff and Yasuda (2024) proposes an algorithm to output a slightly higher rank output as a proxy to solve the weighted low-rank approximation problem efficiently. Li et al. (2016b) develops an efficient framework for alternating minimization to get the weighted low-rank approximation. Similarly, Song et al. (2023b) proposes a more robust framework to get the solution.

2.4 Sketching for Numerical Linear Algebra

In this work, we use the sketching technique to accelerate a submodular optimization problem. We provide a brief overview of prior sketching work across various domains. Sketching has been utilized to enhance numerous continuous optimization challenges, including linear programming Cohen et al. (2019b); Song (2019); Brand (2020); Jiang et al. (2021); Song and Yu (2021); Gu and Song (2022), empirical risk minimization Lee et al. (2019); Qin et al. (2023a), the cutting plane method Jiang et al. (2020), calculating the John Ellipsoid Cohen et al. (2019a); Song et al. (2022b), and many more. Beyond continuous optimization, sketching has also been applied to several discrete optimization issues Deng et al. (2022); Song et al. (2022a); Zhang (2022); Jiang et al. (2023). Additionally, sketching concepts have been implemented in addressing theoretical problems in large language models, such as exponential and softmax regression Li et al. (2023); Gao et al. (2023); Deng et al. (2023b); Li et al. (2024a), and reducing the feature dimension of attention matrices Deng et al. (2023c). Sketching techniques prove valuable in various machine learning tasks, including matrix completion Gu et al. (2023), adversarial training Gao et al. (2022), training over-parameterized neural tangent kernel regression Brand et al. (2021); Song et al. (2021); Zhang (2022); Alman et al. (2023); Hu et al. (2022), matrix sensing Deng et al. (2023a); Qin et al. (2023b), kernel density estimation Qin et al. (2022b), and federated learning Song

et al. (2023a). Moreover, the application of sketching extends to the theory of relational databases Qin et al. (2022a).

3 PRELIMINARY

We define $[n] := \{1, 2, \ldots, n\}$. Let $\mathbb{R}$ denote the real numbers, and $\mathbb{R}_{\geq 0}$ denote the nonnegative real numbers. Let $\|A\|$ denote the spectral norm of matrix A. Let $\|A\|_F^2 = \sum_{i,j} A_{i,j}^2$ denote the Frobenius norm of A. Let $W \circ A$ denote the entry-wise product of matrices W and A. Let $\|A\|_W^2 = \sum_{i,j} W_{i,j}^2 A_{i,j}^2$ denote the weighted Frobenius norm of A. Let $\mathrm{nnz}(A)$ denote the number of nonzero entries of A. Let $\det(A)$ denote the determinant of a square matrix A. Let $A^\top$ denote the transpose of A. Let $A^\dagger$ denote the Moore-Penrose pseudo-inverse of A. Let A^{-1} denote the inverse of a full rank square matrix A.

For the weight matrix W, we always use $W_{*,j}$ to denote the j-th column of W, and $W_{i,*}$ to denote the i-th row of W. Let $\mathrm{diag}(W_{*,j})$ denote the diagonal matrix with entries from the vector $W_{*,j}$. Let $\mathrm{diag}(W_{i,*})$ denote the diagonal matrix with entries from the vector $W_{i,*}$.

Definition 3.1 (Distinct rows and columns). *Given a matrix $R \in m \times n$, we have $R_{*,1}, R_{*,2}, \ldots, R_{*,n} \in \mathbb{R}^m$ as its column vectors, and $R_{1,*}^\top, R_{2,*}^\top, \ldots, R_{m,*}^\top \in \mathbb{R}^n$ as its row vectors. We define $r := |\{R_{*,1}, R_{*,2}, \ldots, R_{*,n}\}|$ and $p := |\{R_{1,*}^\top, R_{2,*}^\top, \ldots, R_{m,*}^\top\}|$, where $|\cdot|$ means the cardinality of a set, i.e., the number of elements in a set. Then, we say R has r distinct columns and p distinct rows.*

Assumption 3.2. *Let $r, p \in \mathbb{N}_+$. We assume $n \times n$ matrix W has r distinct columns and rows. We also assume that $W \circ A$ has rp distinct columns and rows.*

In a real application, when the data has strong periodicity, e.g., signal processing, or uniformly draws from a small set, e.g., classification, or the data has a certain structure, e.g., block attention mask in Transformers (Vaswani et al., 2017; Dong et al., 2024), the matrix may have a small number of distinct columns or rows.

Let $\mathbb{Z}[x_1, x_2, \cdots, x_v]$ denote the set of multi-variable polynomial with v variables and the coefficients are from $\mathbb{Z}$. Furthermore, if all the coefficients of a polynomial are from $\{0, 1, \cdots, 2^H\}$, then we say the bitsize of coefficients of this polynomial is H.

Lemma 3.3 (Cramer's rule). *Let R be an $n \times n$ invertible matrix. Then, for each $i \in [n]$, $j \in [n]$,*

$$(R^{-1})_{i,j} = \det(R_{-j}^{-i}) / \det(R),$$

where R_{-j}^{-i} is the matrix R with the i-th row and j-th column removed.

3.1 Polynomial System Decision Solver

Here, we formally state the theorem of the decision problem. For a full discussion of algorithms in real algebraic geometry, we refer the reader to Basu et al. (2005) and Basu (2014).

Theorem 3.4 (Decision Problem (Renegar, 1992a,b; Basu et al., 1996)). *Given a real polynomial system $P(x_1, x_2, \cdots, x_v)$ having v variables and m polynomial constraints $f_i(x_1, x_2, \cdots, x_v) \Delta_i 0, \forall i \in [m]$, where Δ_i is any of the "standard relations": $\{>, \geq, =, \neq, \leq, <\}$, let d denote the maximum degree of all the polynomial constraints and let H denote the maximum bitsize of the coefficients of all the polynomial constraints. Then in*

$$(md)^{O(v)} \operatorname{poly}(H)$$

time, one can determine whether a real solution exists for the polynomial system P.

3.2 Lower Bound on Cost

The key result we used for proving the lower bound is the following bound on the minimum value attained by an integer polynomial restricted to a compact connected component of a basic closed semi-algebraic subset of $\mathbb{R}^v$ (see details in Definition 3.9) defined by polynomials with integer coefficients in terms of the degrees and the bitsizes of the coefficients of the polynomials involved.

Theorem 3.5 (Jeronimo et al. (2013)). *Let $T := \{x \in \mathbb{R}^v \mid f_1(x) \geq 0, \cdots, f_\ell(x) \geq 0, f_{\ell+1}(x) = 0, \cdots, f_m(x) = 0\}$ be defined by polynomials $f_1, \cdots, f_m \in \mathbb{Z}[x_1, \cdots, x_v]$ with $v \geq 2$, degrees bounded by an even integer d and coefficients of absolute value at most H bitsize, and let C be a compact connected component of T. Let $g \in \mathbb{Z}[x_1, \cdots, x_v]$ (here g can be viewed as an objective function) be a polynomial of degree at most d and coefficients of absolute value bounded by H. Let $\widetilde{H} := \max\{H, 2v + 2m\}$. Then, the minimum value that g takes over C satisfies that if it is not zero, then its absolute value is $\geq$*

$$2^{-2^{3v \log(d)} \log(\widetilde{H})}.$$

3.3 Upper Bound on Cost

Now, we provide the upper bound on the OPT, which is defined below.

Definition 3.6. *Given $A, W \in \mathbb{R}^{n \times n}$ and $k \in [n]$, we define* OPT *as*

$$\mathrm{OPT} := \min_{U \in \mathbb{R}^{n \times k}, V \in \mathbb{R}^{n \times k}} \|(UV^\top - A) \circ W\|_F^2.$$

Lemma 3.7 (folklore). *Let A, W, OPT be defined in Definition 3.6. Let each entry of A, W can be represented by n^γ bits, with $\gamma \in (0, 1)$. Then, we have*

$$\mathrm{OPT} \leq \operatorname{poly}(n) \cdot 2^{n^\gamma}.$$

Proof. Set U and V to be the zero matrices. □

3.4 Sketching Tool

We use a sketching tool from previous work.

Lemma 3.8 (Theorem 3.1 in Razenshteyn et al. (2016)). *Let $A^1, \ldots, A^m \in \mathbb{R}^{n \times k}$ be m matrices of size $n \times k$. Let $b^1, \ldots, b^m \in \mathbb{R}^{n \times 1}$ be m column vectors of dimension n. For $1 \leq i \leq m$ denote: $x^i = \arg\min_{x \in \mathbb{R}^k} \|A^i x - b^i\|_2^2$ the solution of the i-th regression problem. Let $S \in \mathbb{R}^{t \times n}$ be a random matrix with i.i.d. Gaussian entries with zero mean and standard deviation $1/\sqrt{t}$. For $1 \leq i \leq m$ denote $y^i = \arg\min_{y \in \mathbb{R}^k} \|SA^i y - Sb^i\|_2^2$ the solution of the i-th regression problem in the sketch space. We claim that for every $0 < \epsilon < 1/2$, with high probability, one can set $t = O(k/\epsilon)$ such that:*

$$\sum_{i=1}^m \|A^i y^i - b^i\|_2^2 \leq (1 + \epsilon) \cdot \sum_{i=1}^m \|A^i x^i - b^i\|_2^2.$$

3.5 Backgrounds on Semi-Algebraic Sets

The following real algebraic geometry definitions are needed when proving a lower bound for the minimum nonzero cost of our problem. For a full discussion, we refer the reader to Bochnak et al. (1987). Here, we use the brief summary by Basu et al. (2005).

Definition 3.9 (Basu et al. (2005)). *Let R be a real closed field. Given $x = (x_1, \cdots, x_v) \in R^v, r \in R, r > 0$, we denote*

$$B_v(x, r) := \{y \in R^v \mid \|y - x\|^2 < r^2\} \quad \text{(open ball)},$$
$$\bar{B}_v(x, r) := \{y \in R^v \mid \|y - x\|^2 \leq r^2\} \quad \text{(closed ball)}.$$

A set $S \subset R^v$ is open if it is the union of open balls, i.e., if every point of U is contained in an open ball contained in U.

A set $S \subset R^v$ is closed if its complement is open. Clearly, the arbitrary union of open sets is open, and the arbitrary intersection of closed sets is closed.

Semi-algebraic sets are defined by a finite number of polynomial inequalities and equalities.

A semi-algebraic set has a finite number of connected components, each of which is semi-algebraic. Here, we use the topological definition of a connected component, which is a maximal connected subset (ordered by

inclusion), where connected means it cannot be divided into two disjoint nonempty closed sets.

A closed and bounded semi-algebraic set is compact.

A semi-algebraic set $S \subset R^v$ is semi-algebraically connected if S is not the disjoint union of two non-empty semialgebraic sets that are both closed in S. Or, equivalently, S does not contain a non-empty semi-algebraic strict subset which is both open and closed in S.

A semi-algebraically connected component of a semi-algebraic set S is a maximal semi-algebraically connected subset of S.

4 WARMUP

In this section, to demonstrate the new technique, we prove the following theorem.

Lemma 4.1. *Given two $n \times n$ size matrices A and W, $1 \leq k \leq n$ such that: Let each entry of A, W can be represented by n^γ bits, with $\gamma \in (0, 1)$; Let OPT be defined as Definition 3.6. Then, we can show*

- **Part 1.** *There is an algorithm that runs in $2^{O(nk \log n)}$ time, and outputs a number Λ such that $\mathrm{OPT} \leq \Lambda \leq 2 \, \mathrm{OPT}$.*

- **Part 2.** *There is an algorithm that runs in $2^{O(nk \log n)}$ time and returns $U \in \mathbb{R}^{n \times k}$ and $V \in \mathbb{R}^{n \times k}$ such that $\|(UV^\top - A) \circ W\|_F^2 \leq 2 \, \mathrm{OPT}$.*

Proof. **Proof of Part 1.**

We can create $2nk$ variables to explicitly represent each entry of U and V. Let $g(x) = \|W \circ (UV^\top - A)\|_F^2$. Let $L = 2^{n^\gamma}$. Then, we can write down a polynomial system (the decision problem defined in Theorem 3.4)

$$\min g(x)$$
$$\text{s.t.} \quad U_{i,j} \in [-L, L], \forall i, j$$
$$V_{i,j} \in [-L, L], \forall i, j$$

Using Theorem 3.5, we know the above system has

$$\mathsf{m} = 2nk, \mathsf{v} = 2nk, \mathsf{d} = 4, \mathsf{H} = n^\gamma, \widetilde{\mathsf{H}} = n^\gamma + O(nk).$$

The lower bound on $g(x)$ (if $g(x)$ is not zero) is going to be

$$c_{\text{lower}} = 2^{-2^{3\mathsf{v} \log(\mathsf{d})} \log(\widetilde{H})}$$
$$\geq 2^{-2^{O(nk)} \log(n^\gamma + nk)}$$
$$\geq 2^{-2^{O(nk \log n)}}.$$

By Lemma 3.7, we know the upper bound is $C_{\text{upper}} = \mathrm{poly}(n) \cdot 2^{n^\gamma}$. After knowing the lower bound and up-

per bound on cost, the number of binary search iterations is upper-bounded by

$$\log(\frac{C_{\text{upper}}}{c_{\text{lower}}}) = \log(\frac{\mathrm{poly}(n) 2^{n^\gamma}}{2^{-2^{O(nk \log n)}}}) \leq 2^{O(nk \log n)}.$$

In each of the above iterations, we need to run Theorem 3.4 with the system

$$\text{s.t.} \quad g(x) \in [\Gamma_t, 2\Gamma_t], \ U_{i,j} \in [-L, L], \ V_{i,j} \in [-L, L]$$

with parameters,

$$\mathsf{m} = 2nk + 1, \mathsf{v} = 2nk, \mathsf{d} = 4, \mathsf{H} = n^\gamma.$$

Then, running time complexity is

$$(\mathsf{md})^{O(\mathsf{v})} \cdot \mathrm{poly}(\mathsf{H}) = (10nk)^{O(nk)} \cdot \mathrm{poly}(n^\gamma)$$
$$= 2^{O(nk \log n)}.$$

Thus, combining the number of iterations and time for each iteration, we can find the number $\Gamma \in [\mathrm{OPT}, 2\,\mathrm{OPT}]$.

Proof of Part 2.

Next, similar to **Part 1**, we need to repeat the binary search for $2nk$ times for each variable in U and V, and each time, the number of total binary search steps is n^γ. Thus, we can output the U, V in the same running time as finding Γ. $\qquad \square$

In the next few sections, we will explain how to reduce the number of variables and how to reduce the number of constraints.

5 LOWER BOUND ON OPT

We assume that W has r distinct rows and r distinct columns. Then, we get rid of the dependence on n in the degree.

Theorem 5.1 (Implicitly in Razenshteyn et al. (2016)). *Assuming that W has r distinct rows and r distinct columns, each entry of A and W needs n^γ bits to represent. Assume $\mathrm{OPT} > 0$. Then we know that, with high probability,*

$$\mathrm{OPT} \geq 2^{-n^\gamma} 2^{\widetilde{O}(rk^2/\epsilon)}.$$

Proof. We use $A_{i,*} \in \mathbb{R}^n$ denote the i-th row of A. We use $W_{i,*} \in \mathbb{R}^n$ to denote the i-th row of W. Let $(U_1)_{i,*}$ denote the i-th row of U_1. For any $n \times k$ matrix U_1 and for any $k \times n$ matrix Z_1, we have

$$\|(U_1 Z_1 - A) \circ W\|_F^2$$

$$= \sum_{i=1}^{n} \|(U_1)_{i,*} Z_1 \operatorname{diag}(W_{i,*}) - A_{i,*} \operatorname{diag}(W_{i,*})\|_2^2.$$

Based on the observation that W has r distinct rows, we use group $g_{1,1}, g_{1,2}, \cdots, g_{1,r}$ to denote r disjoint sets such that

$$\cup_{i=1}^{r} g_{1,i} = [n]$$

For any $i \in [r]$, for any $j_1, j_2 \in g_{1,i}$, we have $W_{j_1,*} = W_{j_2,*}$.

Thus, we can have

$$\sum_{i=1}^{n} \|(U_1)_{i,*} Z_1 \operatorname{diag}(W_{i,*}) - A_{i,*} \operatorname{diag}(W_{i,*})\|_2^2$$

$$= \sum_{i=1}^{r} \sum_{\ell \in g_{1,i}} \|(U_1)_{\ell,*} Z_1 \operatorname{diag}(W_{\ell,*}) - A_{\ell,*} \operatorname{diag}(W_{\ell,*})\|_2^2.$$

We can sketch the objective function by choosing Gaussian matrices $S_1 \in \mathbb{R}^{n \times s_1}$ with $s_1 = O(k/\epsilon)$.

$$\sum_{i=1}^{n} \|(U_1)_{i,*} Z_1 \operatorname{diag}(W_{i,*}) S_1 - A_{i,*} \operatorname{diag}(W_{i,*}) S_1\|_2^2.$$

Let $\widehat{U}_1$ denote the optimal solution of the sketch problem,

$$\widehat{U}_1 = \arg \min_{U_1 \in \mathbb{R}^{n \times k}}$$

$$\sum_{i=1}^{n} \|(U_1)_{i,*} Z_1 \operatorname{diag}(W_{i,*}) S_1 - A_{i,*} \operatorname{diag}(W_{i,*}) S_1\|_2^2.$$

By properties of S_1, plugging $\widehat{U}_1$ into the original problem, we obtain

$$\sum_{i=1}^{r} \sum_{\ell \in g_{1,i}} \|(\widehat{U}_1)_{\ell,*} Z_1 \operatorname{diag}(W_{\ell,*}) - A_{\ell,*} \operatorname{diag}(W_{\ell,*})\|_2^2$$

$$\leq (1 + \epsilon) \cdot \mathrm{OPT}.$$

Let R denote the set of all $\mathsf{S}(g_{1,i})$ (for all $i \in [r]$ and $|R| = r$)

Note that $\widehat{U}_1$ also has the following form, for each $\ell \in L \subset [n]$ (Note that $|L| = rp$.)

$$(\widehat{U}_1)_{\ell,*} = A_{\ell,*} \operatorname{diag}(W_{\ell,*}) S_1 \cdot (Z_1 \operatorname{diag}(W_{\ell,*}) S_1)^{\dagger}$$

$$= A_{\ell,*} \operatorname{diag}(W_{\ell,*}) S_1 \cdot (Z_1 \operatorname{diag}(W_{\ell,*}) S_1)^{\top}$$

$$\cdot ((Z_1 \operatorname{diag}(W_{\ell,*}) S_1)(Z_1 \operatorname{diag}(W_{\ell,*}) S_1)^{\top})^{-1}.$$

Recall the number of different $\operatorname{diag}(W_{\ell,*})$ is at most r.

For each $k \times s_1$ matrix $Z_1 \operatorname{diag}(W_{\ell,*}) S_1$, we create $k \times s_1$ variables to represent it. Thus, we create r matrices,

$$\{Z_1 D_{W_{i,*}} S_1\}_{i \in R}.$$

For simplicity, let $P_{1,i} \in \mathbb{R}^{k \times s_1}$ denote $Z_1 \operatorname{diag}(W_{i,*}) S_1$. Then we can rewrite $\widehat{U}^i$ as follows

$$\widehat{U}_1^i = A_{i,*} \operatorname{diag}(W_{i,*}) S_1 \cdot P_{1,i}^{\top} (P_{1,i} P_{1,i}^{\top})^{-1}.$$

If $P_{1,i} P_{1,i}^{\top} \in \mathbb{R}^{k \times k}$ has rank-k, then we can use Cramer's rule to write down the inverse of $P_{1,i} P_{1,i}^{\top}$. For the situation, it is not full rank. We can guess the rank. Let $t_i \leq k$ denote the rank of $P_{1,i}$. Then, we need to figure out a maximal linearly independent subset of columns of $P_{1,i}$. We can also guess all the possibilities, which is at most $2^{O(k)}$. Because we have r different $P_{1,i}$, the total number of guesses we have is at most $2^{O(rk)}$. Thus, we can write down $(P_{1,i} P_{1,i}^{\top})^{-1}$ according to Cramer's rule. Note that $(P_{1,i} P_{1,i}^{\top})^{-1}$ can be view as P_a/P_b where P_a is a polynomial and P_b is another polynomial which is essentially $\det(P_{1,i} P_{1,i}^{\top})$.

After $\widehat{U}_1$ is obtained, we fix $\widehat{U}_1$. We consider

$$\widehat{U}_2 = \arg \min_{U_2 \in \mathbb{R}^{n \times k}} \|(\widehat{U}_1 U_2^{\top} - A) \circ W\|_F^2.$$

In a similar way, we can get and write $\widehat{U}_2$.

Overall, by creating $l = O(rk^2/\epsilon)$ variables, we have rational polynomials $\widehat{U}_1(x)$ and $\widehat{U}_2(x)$. Note that $\widehat{U}_1(x)$ only has rp different rows, and same for $\widehat{U}_2(x)$.

Indeed, now we have only $2r$ distinct denominators (w.l.o.g., assume the first r columns are distinct and the first r rows are distinct),

$$h_{1,i}(x) = \det(P_{1,i} P_{1,i}^{\top}), \forall i \in [r]$$

$$h_{2,i}(x) = \det(P_{2,i} P_{2,i}^{\top}), \forall i \in [r].$$

Then, we can write down the following optimization problem,

$$\min_{x \in \mathbb{R}^l} p(x)/q(x)$$

$$\text{s.t. } h_{1,i}^2(x) \neq 0, h_{2,i}^2(x) \neq 0, \forall i \in [r],$$

$$q(x) = \prod_{i=1}^{r} h_{1,i}^2(x) h_{2,i}^2(x),$$

where $q(x)$ has degree $O(rk)$, the maximum coefficient in absolute value is

$$(2^{n^{\gamma}})^{O(rk)}$$

and the number of variables $O(rk^2/\epsilon)$. However, that formulation $p(x)/q(x)$ is not a polynomial, to further make it a polynomial, we introduce variable y:

$$\min_{x \in \mathbb{R}^l} p(x) y$$

Chenyang Li[1], Yingyu Liang[2,3], Zhenmei Shi[3], Zhao Song[4]

s.t. $h_{1,i}^2(x) \neq 0, h_{2,i}^2(x) \neq 0, \forall i \in [r]$,

$$q(x) = \prod_{i=1}^r h_{1,i}^2(x) h_{2,i}^2(x)$$

$$q(x)y - 1 = 0.$$

Applying Theorem 3.5, with parameters

$$\mathsf{m} = O(r), \mathsf{v} = O(rk^2/\epsilon), \mathsf{d} = O(r),$$
$$\mathsf{H} = 2^{O(n^\gamma rk)}, \widetilde{H} = O(\mathsf{H}).$$

we can achieve the following minimum nonzero cost:

$$\geq 2^{-2^{3\mathsf{v}\log(\mathsf{d})}\log(\widetilde{H})} \geq 2^{-n^\gamma 2^{\widetilde{O}(rk^2/\epsilon)}}.$$

Thus, we complete the proofs. $\qquad \square$

6 FEW DISTINCT COLUMNS

In this section, we try to estimate the OPT value.

Theorem 6.1. *Given a matrix A and W, each entry can be written using $O(n^\gamma)$ bits for $\gamma > 0$. Given $A \in \mathbb{R}^{n \times n}, W \in \mathbb{R}^{n \times n}, 1 \leq k \leq n$. Assume that W has r distinct columns and rows.*

Then, with high probability, one can output a number Λ in time $n^{1+\gamma} \cdot p \cdot 2^{O(k^2 r/\epsilon)}$ such that

$$\text{OPT} \leq \Lambda \leq (1 + \epsilon)\,\text{OPT}\,.$$

Proof. Let $U_1^*, U_2^* \in \mathbb{R}^{n \times k}$ denote the matrices satisfying

$$\|W \circ (U_1^*(U_2^*)^\top - A)\|_F^2 = \text{OPT}\,.$$

We use $A_{i,*} \in \mathbb{R}^n$ denote the i-th row of A. We use $W_{i,*} \in \mathbb{R}^n$ to denote the i-th row of W. Let $(U_1)_{i,*}$ denote the i-th row of U_1. Let $Z_1 = (U_2^*)^\top$. For any $n \times k$ matrix U_1,

$$\|(U_1 Z_1 - A) \circ W\|_F^2$$
$$= \sum_{i=1}^n \|(U_1)_{i,*} Z_1 \operatorname{diag}(W_{i,*}) - A_{i,*} \operatorname{diag}(W_{i,*})\|_2^2.$$

Based on the observation that W has r distinct rows, we use group $g_{1,1}, g_{1,2}, \cdots, g_{1,r}$ to denote r disjoint sets such that

$$\cup_{i=1}^r g_{1,i} = [n]$$

For any $i \in [r]$, for any $j_1, j_2 \in g_{1,i}$, we have $W_{j_1,*} = W_{j_2,*}$.

Next, based on assumptions on W and A, we use $g_{1,i,1}$, $g_{1,i,2}$, $g_{1,i,p}$ to denote p groups such that

$$\cup_{j=1}^p g_{1,i,j} = g_{1,i}.$$

For any $i \in [r]$, for any $j \in [p]$, for any $\ell_1, \ell_2 \in g_{1,i,j}$, we have $(W_{\ell_1,*} \circ A_{\ell_1,*}) = (W_{\ell_2,*} \circ A_{\ell_2,*})$.

Let $\mathsf{S}(g_{1,i,j})$ denote the smallest index from set $g_{1,i,j}$

Thus, we can have

$$\sum_{i=1}^n \|(U_1)_{i,*} Z_1 \operatorname{diag}(W_{i,*}) - A_{i,*} \operatorname{diag}(W_{i,*})\|_2^2$$
$$= \sum_{i=1}^r \sum_{j=1}^p \sum_{\ell \in g_{1,i,j}} \|(U_1)_{\ell,*} Z_1 \operatorname{diag}(W_{\ell,*})$$
$$- A_{\ell,*} \operatorname{diag}(W_{\ell,*})\|_2^2$$
$$= \sum_{i=1}^r \sum_{j=1}^p |g_{1,i,j}| \cdot \|(U_1)_{\mathsf{S}(g_{1,i,j}),*} Z_1 \operatorname{diag}(W_{\mathsf{S}(g_{1,i,j}),*})$$
$$- A_{\mathsf{S}(g_{1,i,j}),*} \operatorname{diag}(W_{\mathsf{S}(g_{1,i,j}),*})\|_2^2.$$

We can sketch the objective function by choosing Gaussian matrices $S_1 \in \mathbb{R}^{n \times s_1}$ with $s_1 = O(k/\epsilon)$.

$$\sum_{i=1}^r \sum_{j=1}^p |g_{1,i,j}| \cdot \|(U_1)_{\mathsf{S}(g_{1,i,j}),*} Z_1 \operatorname{diag}(W_{\mathsf{S}(g_{1,i,j}),*})S_1$$
$$- A_{\mathsf{S}(g_{1,i,j}),*} \operatorname{diag}(W_{\mathsf{S}(g_{1,i,j}),*})S_1\|_2^2.$$

Let $\widehat{U}_1$ denote the optimal solution of the sketch problem,

$$\widehat{U}_1 = \arg\min_{U_1} \sum_{i=1}^r \sum_{j=1}^p |g_{1,i,j}|$$
$$\cdot \|(U_1)_{\mathsf{S}(g_{1,i,j}),*} Z_1 \operatorname{diag}(W_{\mathsf{S}(g_{1,i,j}),*})S_1$$
$$- A_{\mathsf{S}(g_{1,i,j}),*} \operatorname{diag}(W_{\mathsf{S}(g_{1,i,j}),*})S_1\|_2^2.$$

By properties of S_1, plugging $\widehat{U}_1$ into the original problem, we obtain

$$\sum_{i=1}^r \sum_{j=1}^p |g_{1,i,j}| \cdot \|(U_1)_{\mathsf{S}(g_{1,i,j}),*} Z_1 \operatorname{diag}(W_{\mathsf{S}(g_{1,i,j}),*})$$
$$- A_{\mathsf{S}(g_{1,i,j}),*} \operatorname{diag}(W_{\mathsf{S}(g_{1,i,j}),*})\|_2^2 \leq (1 + \epsilon) \cdot \text{OPT}\,.$$

Let R denote the set of all $\mathsf{S}(g_{1,i})$ (for all $i \in [r]$ and $|R| = r$).

Let L denote the set of all $\mathsf{S}(g_{1,i,j})$ (for all $i \in [r]$, $j \in [p]$ and $|L| = rp$).

Note that $\widehat{U}_1$ also has the following form, for each $\ell \in L \subset [n]$ (Note that $|L| = rp$.)

$$(U_1)_{\ell,*} = A_{\ell,*} \operatorname{diag}(W_{\ell,*})S_1 \cdot (Z_1 \operatorname{diag}(W_{\ell,*})S_1)^\dagger$$
$$= A_{\ell,*} \operatorname{diag}(W_{\ell,*})S_1 \cdot (Z_1 \operatorname{diag}(W_{\ell,*})S_1)^\top$$
$$\cdot ((Z_1 \operatorname{diag}(W_{\ell,*})S_1)(Z_1 \operatorname{diag}(W_{\ell,*})S_1)^\top)^{-1}.$$

Recall the number of different $A_{\ell,*} \operatorname{diag}(W_{\ell,*})$ is at most rp, and the number of different $\operatorname{diag}(W_{\ell,*})$ is at

most r. For each $k \times s_1$ matrix $Z_1 \operatorname{diag}(W_{\ell,*}) S_1$, we create $k \times s_1$ variables to represent it. Thus, we create r matrices,

$$\{Z_1 \operatorname{diag}(W_{i,*}) S_1\}_{i \in R}.$$

For simplicity, let $P_{1,i} \in \mathbb{R}^{k \times s_1}$ denote $Z_1 \operatorname{diag}(W_{i,*}) S_1$. Then we can rewrite $(\widehat{U}_1)_{i,*}$ as follows

$$(\widehat{U}_1)_{i,*} = A_{i,*} \operatorname{diag}(W_{i,*}) S_1 \cdot P_{1,i}^\top (P_{1,i} P_{1,i}^\top)^{-1}.$$

If $P_{1,i} P_{1,i}^\top \in \mathbb{R}^{k \times k}$ has rank-k, then we can use Cramer's rule to write down the inverse of $P_{1,i} P_{1,i}^\top$. For the situation, it is not full rank. We can guess the rank. Let $t_i \leq k$ denote the rank of $P_{1,i}$. Then, we need to figure out a maximal linearly independent subset of columns of $P_{1,i}$. We can also guess all the possibilities, which is at most $2^{O(k)}$. Because we have r different $P_{1,i}$, the total number of guesses we have is at most $2^{O(rk)}$. Thus, we can write down $(P_{1,i} P_{1,i}^\top)^{-1}$ according to Cramer's rule. Note that $(P_{1,i} P_{1,i}^\top)^{-1}$ can be view as P_a / P_b where P_a is a polynomial and P_b is another polynomial which is essentially $\det(P_{1,i} P_{1,i})$.

After $\widehat{U}_1$ is obtained, we will fix $\widehat{U}_1$ in the next round.

For any $n \times k$ matrix U_2, we can rewrite

$$\|(\widehat{U}_1 U_2^\top - A) \circ \|_F^2$$
$$= \sum_{i=1}^n \| \operatorname{diag}(W_{*,i}) \widehat{U}_1 (U_2^\top)_{*,i} - \operatorname{diag}(W_{*,i}) A_{*,i}\|_2^2.$$

Based on the observation that W has r columns rows, we use group $g_{2,1}, g_{2,2}, \cdots, g_{2,r}$ to denote r disjoint sets such that

$$\cup_{i=1}^r g_{2,i} = [n].$$

For any $i \in [r]$, for any $j_1, j_2 \in g_{2,i}$, we have $W_{*,j_1} = W_{*,j_2}$.

Next, based on assumptions on W and A, we use $g_{2,i,1}$, $g_{2,i,2}$, $g_{2,i,p}$ to denote p groups such that

$$\cup_{j=1}^p g_{2,i,j} = g_{2,i}.$$

For any $i \in [r]$, for any $j \in [p]$, for any $\ell_1, \ell_2 \in g_{2,i,j}$, we have $(W_{*,\ell_1} \circ A_{*,\ell_1}) = (W_{*,\ell_2} \circ A_{*,\ell_2})$.

Let $\mathsf{S}(g_{2,i,j})$ denote the smallest index from set $g_{2,i,j}$

Thus, we can have

$$\sum_{i=1}^n \| \operatorname{diag}(W_{*,i}) \widehat{U}_1 (U_2^\top)_{*,i} - \operatorname{diag}(W_{*,i}) A_{*,i}\|_2^2$$
$$= \sum_{i=1}^r \sum_{j=1}^p \sum_{\ell \in g_{1,i,j}} \| \operatorname{diag}(W_{*,\ell}) \widehat{U}_1 (U_2^\top)_{*,\ell}$$

$$- \operatorname{diag}(W_{*,\ell}) A_{*,\ell} \|_2^2$$
$$= \sum_{i=1}^r \sum_{j=1}^p |g_{2,i,j}| \cdot \| \operatorname{diag}(W_{*,\mathsf{S}(g_{2,i,j})}) \widehat{U}_1 (U_2^\top)_{*,\mathsf{S}(g_{2,i,j})}$$
$$- \operatorname{diag}(W_{*,\mathsf{S}(g_{1,i,j})}) A_{*,\mathsf{S}(g_{1,i,j})} \|_2^2.$$

We can sketch the objective function by choosing Gaussian matrices $S_2 \in \mathbb{R}^{n \times s_1}$ with $s_2 = O(k/\epsilon)$.

$$\sum_{i=1}^r \sum_{j=1}^p |g_{2,i,j}| \cdot \| S_2 \operatorname{diag}(W_{\mathsf{S}(g_{2,i,j}),*}) \widehat{U}_1 (U_2^\top)_{*,\mathsf{S}(g_{2,i,j})}$$
$$- S_2 \operatorname{diag}(W_{*,\mathsf{S}(g_{2,i,j})}) A_{*,\mathsf{S}(g_{2,i,j})} \|_2^2.$$

Let $\widehat{U}_1$ denote the optimal solution of the sketch problem,

$$\widehat{U}_1 = \arg\min_{U_1} \sum_{i=1}^r \sum_{j=1}^p |g_{2,i,j}|$$
$$\cdot \| S_2 \operatorname{diag}(W_{\mathsf{S}(g_{2,i,j}),*}) \widehat{U}_1 (U_2^\top)_{*,\mathsf{S}(g_{2,i,j})}$$
$$- S_2 \operatorname{diag}(W_{*,\mathsf{S}(g_{2,i,j})}) A_{*,\mathsf{S}(g_{2,i,j})} \|_2^2.$$

By properties of S_1, plugging $\widehat{U}_2$ into the original problem, we obtain

$$\sum_{i=1}^r \sum_{j=1}^p |g_{2,i,j}| \cdot \| \operatorname{diag}(W_{\mathsf{S}(g_{2,i,j}),*}) \widehat{U}_1 (\widehat{U}_2^\top)_{*,\mathsf{S}(g_{2,i,j})}$$
$$- \operatorname{diag}(W_{*,\mathsf{S}(g_{2,i,j})}) A_{*,\mathsf{S}(g_{2,i,j})} \|_2^2 \leq (1 + \epsilon) \cdot \text{OPT}.$$

Let R denote the set of all $\mathsf{S}(g_{1,i})$ (for all $i \in [r]$ and $|R| = r$).

Let L denote the set of all $\mathsf{S}(g_{1,i,j})$ (for all $i \in [r]$, $j \in [p]$ and $|L| = rp$).

Note that $\widehat{U}_1$ also has the following form, for each $\ell \in L \subset [n]$ (Note that $|L| = rp$.)

$$(U_1)_{\ell,*} = (S_2 \operatorname{diag}(W_{*,\ell}) \widehat{U}_1)^\dagger \cdot S_2 \operatorname{diag}(W_{*,\ell}) A_{*,\ell}$$
$$= ((S_2 \operatorname{diag}(W_{*,\ell}) \widehat{U}_1)(S_2 \operatorname{diag}(W_{*,\ell}) \widehat{U}_1)^\top)^{-1}$$
$$\cdot S_2 \operatorname{diag}(W_{*,\ell}) \widehat{U}_1 \cdot S_2 \operatorname{diag}(W_{*,\ell}) A_{*,\ell}.$$

Recall the number of different $A_{*,\ell} \operatorname{diag}(W_{*,\ell})$ is at most rp, and the number of different $\operatorname{diag}(W_{*,\ell})$ is at most r.

For each $s_2 \times k$ matrix $S_2 \operatorname{diag}(W_{*,i}) \widehat{U}_1$, we create $s_2 \times k$ variables to represent it. Thus, we create r matrices,

$$\{S_2 \operatorname{diag}(W_{*,i}) \widehat{U}_1\}_{i \in R}.$$

For simplicity, let $P_{2,i} \in \mathbb{R}^{k \times s_2}$ denote $S_2 \operatorname{diag}(W_{*,i}) \widehat{U}_1$. Then we can rewrite $\widehat{U}_2$ as follows

$$(\widehat{U}_2^\top)_{*,i} = (P_{2,i} P_{2,i}^\top)^{-1} P_{2,i} S_2 \operatorname{diag}(W_{*,i}) A_{*,i}.$$

Chenyang Li[1], Yingyu Liang[2,3], Zhenmei Shi[3], Zhao Song[4]

In a similar way, we can write $\widehat{U}_2$. Overall, by creating $l = O(rk^2/\epsilon)$ variables, we have rational polynomials $\widehat{U}_1(x)$ and $\widehat{U}_2(x)$. Note that $\widehat{U}_1(x)$ only has rp different rows, and same for $\widehat{U}_2(x)$.

Putting it all together, we can write the objective function,

$$\min_{x \in \mathbb{R}^l} \|(\widehat{U}_1(x)\widehat{U}_2(x)^\top - A) \circ W\|_F^2$$
$$\text{s.t.} \quad h_{1,i}(x) \neq 0, \forall i \in [r]$$
$$h_{2,i}(x) \neq 0, \forall i \in [r].$$

Note that $\widehat{U}_1(x)\widehat{U}_2(x) \circ W$ only has rp distinct rows. Also, $A \circ W$ only has rp distinct rows. Writing down the objective function $\|(\widehat{U}_1(x)\widehat{U}_2(x)^\top - A) \circ W\|_F^2$ only requires $n(rp) \cdot \text{poly}(kr/\epsilon)$ time.

Combining the binary search explained in Lemma 4.1 with the lower bound on cost (Theorem 5.1) we obtained, we can find the solution for the original problem in time,

$$(np \cdot \text{poly}(kr/\epsilon) + n2^{\widetilde{O}(rk^2/\epsilon)}) \cdot n^\gamma = n^{1+\gamma}p \cdot 2^{\widetilde{O}(rk^2/\epsilon)}.$$

□

7 RECOVER A SOLUTION

We state our results and proof of recovering a solution.

Theorem 7.1. *Given a matrix A and W, each entry can be written using $O(n^\gamma)$ bits for $\gamma > 0$. Given $A \in \mathbb{R}^{n \times n}, W \in \mathbb{R}^{n \times n}, 1 \leq k \leq n$ and $\epsilon \in (0, 0.1)$. Assume W has r distinct columns and rows. Assume $A \circ W$ has at most $r \cdot p$ distinct columns and at most $r \cdot p$ distinct rows. Let OPT be defined as Definition 3.6. Then, with high probability, one can output two matrices $U, V \in \mathbb{R}^{n \times k}$ in time $n^{1+\gamma} \cdot 2^{O(k^2r/\epsilon)}$ such that*

$$\|(UV^\top - A) \circ W\|_F^2 \leq (1 + \epsilon)\text{OPT}.$$

Further, if we choose (1) $k^2r = O(\log n/\log\log n)$, (2) $\epsilon \in (0, 0.1)$ to be a small constant, (3) $p = n^{o(1)}$ and (4) $\gamma = o(1)$, then the running time becomes $n^{1+o(1)}$.

Proof. Here, we show how to recover an approximate solution, not only the value of OPT.

The idea is to recover the entries of U and V one by one and use the algorithm from the previous section for the corresponding decision problem. We initialize the semialgebraic set to be

$$S = \{x \in \mathbb{R}^l \mid q(x) \neq 0, p(x) \leq \Lambda q(x)\}.$$

We start by recovering the first entry of U. We perform the binary search to localize the entry, which

takes $\log(2^{n^\gamma})$ invocations of the decision algorithm. For each step of binary search, we use Theorem 3.5 to determine whether the following semi-algebraic set S is empty or not,

$$S \cap \{U_{1,1}(x) \geq \widehat{U}_{1,1}^-, U_{1,1}(x) \leq \widehat{U}_{1,1}^+\}.$$

After that, we declare the first entry of U to be any point in this interval. Then, we add an equality constraint that fixes the entry of $\widehat{U}$ to this value and add a new constraint into S permanently, e.g., $S \leftarrow S \cap \{U_{1,1}(x) = \widehat{U}_{1,1}\}$. Next, we repeat the same with the second entry of U and so on.

This allows us to recover a solution of cost at most $(1 + \epsilon)\text{OPT}$ in time

$$n^{1+\gamma} \cdot p \cdot 2^{O(k^2r/\epsilon)}.$$

If we choose $\gamma = o(1)$, $\epsilon = \Theta(1)$, $p = n^{o(1)}$ and $k^2r = O(\log n/\log\log n)$, then the running time becomes $n^{1+o(1)}$. Thus, we complete the proof. □

8 CONCLUSION

We showed that the weighted low-rank approximation problem could be solved in almost linear $n^{1+o(1)}$ time when the weighted matrix W has few distinct columns and rows and $W \circ A$ has few distinct columns and rows. This demonstrates that truly subquadratic time is achievable for a dense regime not previously known to be tractable. Future work could generalize the assumptions and explore applications of the algorithm.

9 LIMITATIONS

Although our algorithm can achieve a good approximation ratio in theory, it may not achieve the ideal result in practical implementation due to various factors such as computational resource limitation, data quality problems, or incomplete model specification.

10 SOCIETAL IMPACT

In this paper, we introduce an algorithm that can solve weighted low-rank approximation problems in near-linear time under certain conditions. Our paper is purely theoretical and empirical in nature (a mathematics problem), and thus, we foresee no immediate negative ethical impact. Our algorithms can improve the efficiency of data processing and model training, allowing complex algorithms to be run in resource-limited environments and accelerating scientific research and technological innovation. By reducing computing requirements, energy consumption can be reduced, and the environment can benefit.

References

Josh Alman and Zhao Song. Fast attention requires bounded entries. *arXiv preprint arXiv:2302.13214*, 2023.

Josh Alman and Zhao Song. How to capture higher-order correlations? generalizing matrix softmax attention to kronecker computation. In *The Twelfth International Conference on Learning Representations*, 2024a. URL https://openreview.net/forum?id=v0zNCwwkaV.

Josh Alman and Zhao Song. Fast rope attention: Combining the polynomial method and fast fourier transform. *manuscript*, 2024b.

Josh Alman and Zhao Song. The fine-grained complexity of gradient computation for training large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024c. URL https://openreview.net/forum?id=up4tWnwRol.

Josh Alman, Jiehao Liang, Zhao Song, Ruizhe Zhang, and Danyang Zhuo. Bypass exponential time preprocessing: Fast neural network training via weight-data correlation preprocessing. In *NeurIPS*, 2023.

Frank Ban, David Woodruff, and Richard Zhang. Regularized weighted low rank approximation. *Advances in neural information processing systems*, 32, 2019.

Saugata Basu. Algorithms in real algebraic geometry: a survey. *arXiv preprint arXiv:1409.1534*, 2014.

Saugata Basu, Richard Pollack, and Marie-Françoise Roy. On the combinatorial and algebraic complexity of quantifier elimination. *Journal of the ACM (JACM)*, 43(6):1002–1045, 1996.

Saugata Basu, Richard Pollack, and Marie-Francoise Roy. Algorithms in real algebraic geometry, vol. 20033, 2005.

Srinadh Bhojanapalli, Prateek Jain, and Sujay Sanghavi. Tighter low-rank approximation via sampling the leveraged element. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 902–920. SIAM, 2014.

Srinadh Bhojanapalli, Anastasios Kyrillidis, and Sujay Sanghavi. Dropping convexity for faster semi-definite optimization. In *Conference on Learning Theory*, pages 530–582. PMLR, 2016.

Jacek Bochnak, Michel Coste, and Marie-Françoise Roy. *Géométrie algébrique réelle*, volume 12. Springer Science & Business Media, 1987.

Jan van den Brand. A deterministic linear program solver in current matrix multiplication time. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 259–278. SIAM, 2020.

Jan van den Brand, Binghui Peng, Zhao Song, and Omri Weinstein. Training (overparametrized) neural networks in near-linear time. In *ITCS*, 2021.

Yi Chang, Luxin Yan, Xi-Le Zhao, Houzhang Fang, Zhijun Zhang, and Sheng Zhong. Weighted low-rank tensor recovery for hyperspectral image restoration. *IEEE transactions on cybernetics*, 50(11): 4558–4572, 2020.

Bo Chen, Xiaoyu Li, Yingyu Liang, Zhenmei Shi, and Zhao Song. Bypassing the exponential dependency: Looped transformers efficiently learn in-context by multi-step gradient descent. In *International Conference on Artificial Intelligence and Statistics*, 2025a.

Bo Chen, Yingyu Liang, Zhizhou Sha, Zhenmei Shi, and Zhao Song. Hsr-enhanced sparse attention acceleration. In *Conference on Parsimony and Learning*. PMLR, 2025b.

Chuan Chen, Zhe-Bin Wu, Zi-Tai Chen, Zi-Bin Zheng, and Xiong-Jun Zhang. Auto-weighted robust low-rank tensor completion via tensor-train. *Information Sciences*, 567:100–115, 2021.

Xiaolong Chen, Wei Xu, Shuping Tao, Tan Gao, Qinping Feng, and Yongjie Piao. Total variation weighted low-rank constraint for infrared dim small target detection. *Remote Sensing*, 14(18):4615, 2022.

Yifang Chen, Jiayan Huo, Xiaoyu Li, Yingyu Liang, Zhenmei Shi, and Zhao Song. Fast gradient computation for rope attention in almost linear time. *arXiv preprint arXiv:2412.17316*, 2024.

Michael B Cohen, Ben Cousins, Yin Tat Lee, and Xin Yang. A near-optimal algorithm for approximating the john ellipsoid. In *Conference on Learning Theory*, pages 849–873. PMLR, 2019a.

Michael B Cohen, Yin Tat Lee, and Zhao Song. Solving linear programs in the current matrix multiplication time. In *STOC*, 2019b.

Yichuan Deng, Zhao Song, and Omri Weinstein. Discrepancy minimization in input-sparsity time. *arXiv preprint arXiv:2210.12468*, 2022.

Yichuan Deng, Zhihang Li, and Zhao Song. An improved sample complexity for rank-1 matrix sensing. *arXiv preprint arXiv:2303.06895*, 2023a.

Yichuan Deng, Zhihang Li, and Zhao Song. Attention scheme inspired softmax regression. *arXiv preprint arXiv:2304.10411*, 2023b.

Yichuan Deng, Sridhar Mahadevan, and Zhao Song. Randomized and deterministic attention sparsifica-

tion algorithms for over-parameterized feature dimension. *arxiv preprint: arxiv 2304.03426*, 2023c.

Juechu Dong, Boyuan Feng, Driss Guessous, Yanbo Liang, and Horace He. Flex attention: A programming model for generating optimized attention kernels. *arXiv preprint arXiv:2412.05496*, 2024.

Zhaohui Du, Xuefeng Chen, Han Zhang, Boyuan Yang, Zhi Zhai, and Ruqiang Yan. Weighted low-rank sparse model via nuclear norm minimization for bearing fault detection. *Journal of Sound and Vibration*, 400:270–287, 2017.

Aritra Dutta. Weighted low-rank approximation of matrices: some analytical and numerical aspects. 2016.

Aritra Dutta and Xin Li. Weighted low rank approximation for background estimation problems. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1853–1861, 2017a.

Aritra Dutta and Xin Li. A fast algorithm for a weighted low rank approximation. In *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, pages 93–96. IEEE, 2017b.

Aritra Dutta, Xin Li, and Peter Richtárik. Weighted low-rank approximation of matrices and background modeling. *arXiv preprint arXiv:1804.06252*, 2018.

Anders Eriksson and Anton Van Den Hengel. Efficient computation of robust weighted low-rank matrix approximations using the l_1 norm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34 (9):1681–1690, 2012.

Baojie Fan, Xiaomao Li, Yang Cong, and Yandong Tang. Structured and weighted multi-task low rank tracker. *Pattern Recognition*, 81:528–544, 2018.

Zhiqiang Fu, Yao Zhao, Dongxia Chang, and Yiming Wang. A hierarchical weighted low-rank representation for image clustering and classification. *Pattern Recognition*, 112:107736, 2021.

Zhiqiang Fu, Yao Zhao, Dongxia Chang, Yiming Wang, and Jie Wen. Latent low-rank representation with weighted distance penalty for clustering. *IEEE Transactions on Cybernetics*, 2022a.

Zhiqiang Fu, Yao Zhao, Dongxia Chang, Xingxing Zhang, and Yiming Wang. Auto-weighted low-rank representation for clustering. *Knowledge-Based Systems*, 251:109063, 2022b.

Yeqi Gao, Lianke Qin, Zhao Song, and Yitan Wang. A sublinear adversarial training algorithm. *arXiv preprint arXiv:2208.05395*, 2022.

Yeqi Gao, Zhao Song, and Junze Yin. An iterative algorithm for rescaled hyperbolic functions regression. *arXiv preprint arXiv:2305.00660*, 2023.

Nicolas Gillis and François Glineur. Low-rank matrix approximation with weights or missing data is np-hard. *SIAM Journal on Matrix Analysis and Applications*, 32(4):1149–1165, 2011.

Yuzhou Gu and Zhao Song. A faster small treewidth sdp solver. *arXiv preprint arXiv:2211.06033*, 2022.

Yuzhou Gu, Zhao Song, Junze Yin, and Lichen Zhang. Low rank matrix completion via robust alternating minimization in nearly linear time. *arXiv preprint arXiv:2302.11068*, 2023.

Yen-Chang Hsu, Ting Hua, Sungen Chang, Qian Lou, Yilin Shen, and Hongxia Jin. Language model compression with weighted low-rank factorization. In *International Conference on Learning Representations*, 2021.

Hang Hu, Zhao Song, Omri Weinstein, and Danyang Zhuo. Training overparametrized neural networks in sublinear time. *arXiv preprint arXiv:2208.04508*, 2022.

Jerry Yao-Chieh Hu, Maojiang Su, En-Jui Kuo, Zhao Song, and Han Liu. Computational limits of low-rank adaptation (lora) fine-tuning for transformer models. In *The Thirteenth International Conference on Learning Representations*, 2024a.

Jerry Yao-Chieh Hu, Wei-Po Wang, Ammar Gilani, Chenyang Li, Zhao Song, and Han Liu. Fundamental limits of prompt tuning transformers: Universality, capacity and efficiency. *arXiv preprint arXiv:2411.16525*, 2024b.

Jerry Yao-Chieh Hu, Weimin Wu, Yi-Chen Lee, Yu-Chao Huang, Minshuo Chen, and Han Liu. On statistical rates of conditional diffusion transformers: Approximation, estimation and minimax optimality. *arXiv preprint arXiv:2411.17522*, 2024c.

Jerry Yao-Chieh Hu, Weimin Wu, Zhuoru Li, Sophia Pi, , Zhao Song, and Han Liu. On statistical rates and provably efficient criteria of latent diffusion transformers (dits). *Advances in Neural Information Processing Systems*, 38, 2024d.

Ting Hua, Yen-Chang Hsu, Felicity Wang, Qian Lou, Yilin Shen, and Hongxia Jin. Numerical optimizations for weighted low-rank estimation on language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1404–1416, 2022.

Yan Huang, Guisheng Liao, Yijian Xiang, Lei Zhang, Jie Li, and Arye Nehorai. Low-rank approximation via generalized reweighted iterative nuclear and frobenius norms. *IEEE Transactions on Image Processing*, 29:2244–2257, 2019.

Gabriela Jeronimo, Daniel Perrucci, and Elias Tsigaridas. On the minimum of a polynomial func-

tion on a basic closed semialgebraic set and applications. *SIAM Journal on Optimization*, 23(1):241–255, 2013.

Xuan Ji, Jiuzhen Liang, Lan Di, Yunfei Xia, Zhenjie Hou, Zhan Huan, and Yuxi Huan. Fabric defect fetection via weighted low-rank decomposition and laplacian regularization. *Journal of Engineered Fibers and Fabrics*, 15:1558925020957654, 2020.

Haotian Jiang, Yin Tat Lee, Zhao Song, and Sam Chiu-wai Wong. An improved cutting plane method for convex optimization, convex-concave games and its applications. In *STOC*, 2020.

Haotian Jiang, Yin Tat Lee, Zhao Song, and Lichen Zhang. Convex minimization with integer minima in $\widetilde{O}(n^4)$ time. *arXiv preprint arXiv:2304.03426*, 2023.

Huiqiang Jiang, Yucheng Li, Chengruidong Zhang, Qianhui Wu, Xufang Luo, Surin Ahn, Zhenhua Han, Amir H Abdi, Dongsheng Li, Chin-Yew Lin, et al. Minference 1.0: Accelerating pre-filling for long-context llms via dynamic sparse attention. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Jielin Jiang, Jian Yang, Yan Cui, and Lei Luo. Mixed noise removal by weighted low rank model. *Neurocomputing*, 151:817–826, 2015.

Shunhua Jiang, Zhao Song, Omri Weinstein, and Hengjie Zhang. Faster dynamic matrix inverse for faster lps. In *STOC*, 2021.

Yekun Ke, Xiaoyu Li, Yingyu Liang, Zhizhou Sha, Zhenmei Shi, and Zhao Song. On computational limits and provably efficient criteria of visual autoregressive models: A fine-grained complexity analysis. *arXiv preprint arXiv:2501.04377*, 2025.

Mohammad Ahmar Khan, Ghufran Ahmad Khan, Jalaluddin Khan, Taushif Anwar, Zubair Ashraf, Ibrahim Atoum, Naved Ahmad, Mohammad Shahid, Mohammad Ishrat, and Abdulrahman Abdullah Alghamdi. Adaptive weighted low-rank sparse representation for multi-view clustering. *IEEE Access*, 2023.

Alex Kulesza, Nan Jiang, and Satinder Singh. Low-rank spectral learning with weighted loss functions. In *Artificial Intelligence and Statistics*, pages 517–525. PMLR, 2015.

Peter Langenberg, Emilio Rafael Balda, Arash Behboodi, and Rudolf Mathar. On the effect of low-rank weights on adversarial robustness of neural networks. *arXiv preprint arXiv:1901.10371*, 2019.

Yin Tat Lee, Zhao Song, and Qiuyi Zhang. Solving empirical risk minimization in the current matrix multiplication time. In *Conference on Learning Theory (COLT)*, pages 2140–2157. PMLR, 2019.

Chenglong Li, Xiao Wang, Lei Zhang, Jin Tang, Hejun Wu, and Liang Lin. Weighted low-rank decomposition for robust grayscale-thermal foreground detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(4):725–738, 2016a.

Chenyang Li, Yingyu Liang, Zhenmei Shi, and Zhao Song. Exploring the frontiers of softmax: Provable optimization, applications in diffusion model, and beyond. *arXiv preprint arXiv:2405.03251*, 2024a.

Chenyang Li, Yingyu Liang, Zhenmei Shi, Zhao Song, and Tianyi Zhou. Fourier circuits in neural networks and transformers: A case study of modular arithmetic with multiple inputs. In *International Conference on Artificial Intelligence and Statistics*, 2025.

Xiaoyu Li, Yingyu Liang, Zhenmei Shi, and Zhao Song. A tighter complexity analysis of sparseGPT. In *Workshop on Machine Learning and Compression, NeurIPS 2024*, 2024b.

Yanen Li, Jia Hu, ChengXiang Zhai, and Ye Chen. Improving one-class collaborative filtering by incorporating rich user information. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 959–968, 2010.

Yuanzhi Li, Yingyu Liang, and Andrej Risteski. Recovery guarantee of weighted low-rank approximation via alternating minimization. In *International Conference on Machine Learning*, pages 2358–2367. PMLR, 2016b.

Zhihang Li, Zhao Song, and Tianyi Zhou. Solving regularized exp, cosh and sinh regression problems. *arXiv preprint, 2303.15725*, 2023.

Yingyu Liang, Heshan Liu, Zhenmei Shi, Zhao Song, Zhuoyan Xu, and Junze Yin. Conv-basis: A new paradigm for efficient attention inference and gradient computation in transformers. *arXiv preprint arXiv:2405.05219*, 2024a.

Yingyu Liang, Zhizhou Sha, Zhenmei Shi, Zhao Song, and Yufa Zhou. Multi-layer transformers gradient can be approximated in almost linear time. *arXiv preprint arXiv:2408.13233*, 2024b.

Yingyu Liang, Zhenmei Shi, Zhao Song, and Chiwun Yang. Toward infinite-long prefix in transformer. *arXiv preprint arXiv:2406.14036*, 2024c.

Yingyu Liang, Zhenmei Shi, Zhao Song, and Yufa Zhou. Tensor attention training: Provably efficient learning of higher-order transformers. *arXiv preprint arXiv:2405.16411*, 2024d.

Yingyu Liang, Jiangxuan Long, Zhenmei Shi, Zhao Song, and Yufa Zhou. Beyond linear approximations: A novel pruning approach for attention matrix. In *International Conference on Learning Representations*, 2025.

Chenyang Li[1], Yingyu Liang[2,3], Zhenmei Shi[3], Zhao Song[4]

W-S Lu, S-C Pei, and P-H Wang. Weighted low-rank approximation of general complex matrices and its application in the design of 2-d digital filters. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 44(7):650–655, 1997.

Jonathan H Manton, Robert Mahony, and Yingbo Hua. The geometry of weighted low-rank approximations. *IEEE Transactions on Signal Processing*, 51(2):500–514, 2003.

Ivan Markovsky. Structured low-rank approximation and its applications. *Automatica*, 44(4):891–909, 2008.

Ivan Markovsky and Konstantin Usevich. Software for weighted structured low-rank approximation. *Journal of Computational and Applied Mathematics*, 256:278–292, 2014.

Ivan Markovsky and Sabine Van Huffel. Left vs right representations for solving weighted low-rank approximation problems. *Linear algebra and its applications*, 422(2-3):540–552, 2007.

Deyu Meng, Zongben Xu, Lei Zhang, and Ji Zhao. A cyclic weighted median method for l1 low-rank matrix factorization with missing entries. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 704–710, 2013.

Dongmei Mo, Wai Keung Wong, Zhihui Lai, and Jie Zhou. Weighted double-low-rank decomposition with application to fabric defect detection. *IEEE Transactions on Automation Science and Engineering*, 18(3):1170–1190, 2020.

Yigang Peng, Jinli Suo, Qionghai Dai, and Wenli Xu. Reweighted low-rank matrix recovery and its application in image restoration. *IEEE transactions on cybernetics*, 44(12):2418–2430, 2014.

Xinyu Pu, Hangjun Che, Baicheng Pan, Man-Fai Leung, and Shiping Wen. Robust weighted low-rank tensor approximation for multiview clustering with mixed noise. *IEEE Transactions on Computational Social Systems*, 2023.

Lianke Qin, Rajesh Jayaram, Elaine Shi, Zhao Song, Danyang Zhuo, and Shumo Chu. Adore: Differentially oblivious relational database operators. In *VLDB*, 2022a.

Lianke Qin, Aravind Reddy, Zhao Song, Zhaozhuo Xu, and Danyang Zhuo. Adaptive and dynamic multiresolution hashing for pairwise summations. In *BigData*, 2022b.

Lianke Qin, Zhao Song, Lichen Zhang, and Danyang Zhuo. An online and unified algorithm for projection matrix vector multiplication with application to empirical risk minimization. In *AISTATS*, 2023a.

Lianke Qin, Zhao Song, and Ruizhe Zhang. A general algorithm for solving rank-one matrix sensing. *arXiv preprint arXiv:2303.12298*, 2023b.

Ilya Razenshteyn, Zhao Song, and David P Woodruff. 'weighted low rank approximations with provable guarantees'. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 250–263, 2016.

James Renegar. On the computational complexity and geometry of the first-order theory of the reals. part i: Introduction. preliminaries. the geometry of semialgebraic sets. the decision problem for the existential theory of the reals. *Journal of symbolic computation*, 13(3):255–299, 1992a.

James Renegar. On the computational complexity and geometry of the first-order theory of the reals. part ii: The general decision problem. preliminaries for quantifier elimination. *Journal of Symbolic Computation*, 13(3):301–327, 1992b.

Mieke Schuermans, Philippe Lemmerling, and Sabine Van Huffel. Structured weighted low rank approximation. *Numerical linear algebra with applications*, 11(5-6):609–618, 2004.

Fanhua Shang, LC Jiao, Jiarong Shi, Maoguo Gong, and RH Shang. Fast density-weighted low-rank approximation spectral clustering. *Data Mining and Knowledge Discovery*, 23:345–378, 2011.

Zhenmei Shi, Yifei Ming, Ying Fan, Frederic Sala, and Yingyu Liang. Domain generalization via nuclear norm regularization. In *Conference on Parsimony and Learning*, pages 179–201. PMLR, 2024a.

Zhenmei Shi, Yifei Ming, Xuan-Phi Nguyen, Yingyu Liang, and Shafiq Joty. Discovering the gems in early layers: Accelerating long-context llms with 1000x input token reduction. *arXiv preprint arXiv:2409.17422*, 2024b.

Zhenmei Shi, Junyi Wei, Zhuoyan Xu, and Yingyu Liang. Why larger language models do incontext learning differently? *arXiv preprint arXiv:2405.19592*, 2024c.

Zhao Song. *Matrix theory: optimization, concentration, and algorithms*. The University of Texas at Austin, 2019.

Zhao Song and Zheng Yu. Oblivious sketching-based central path method for linear programming. In *International Conference on Machine Learning*, pages 9835–9847. PMLR, 2021.

Zhao Song, Lichen Zhang, and Ruizhe Zhang. Training multi-layer over-parametrized neural network in subquadratic time. *arXiv preprint arXiv:2112.07628*, 2021.

Zhao Song, Zhaozhuo Xu, and Lichen Zhang. Speeding up sparsification using inner product search data structures. *arXiv preprint arXiv:2204.03209*, 2022a.

Zhao Song, Xin Yang, Yuanyuan Yang, and Tianyi Zhou. Faster algorithm for structured john ellipsoid computation. *arXiv preprint arXiv:2211.14407*, 2022b.

Zhao Song, Yitan Wang, Zheng Yu, and Lichen Zhang. Sketching for first order method: Efficient algorithm for low-bandwidth channel and vulnerability. In *ICML*, 2023a.

Zhao Song, Mingquan Ye, Junze Yin, and Lichen Zhang. Efficient alternating minimization with applications to weighted low rank approximation. *arXiv preprint arXiv:2306.04169*, 2023b.

Nathan Srebro and Tommi Jaakkola. Weighted low-rank approximations. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 720–727, 2003.

Le Sun, Feiyang Wu, Tianming Zhan, Wei Liu, Jin Wang, and Byeungwoo Jeon. Weighted nonlocal low-rank tensor decomposition method for sparse unmixing of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:1174–1188, 2020.

Chang Tang, Pichao Wang, Changqing Zhang, and Wanqing Li. Salient object detection via weighted low rank matrix recovery. *IEEE Signal Processing Letters*, 24(4):490–494, 2016.

Elena Tuzhilina and Trevor Hastie. Weighted low rank matrix approximation and acceleration. *arXiv preprint arXiv:2109.11057*, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Lei Wang, Bangjun Wang, Zhao Zhang, Qiaolin Ye, Liyong Fu, Guangcan Liu, and Meng Wang. Robust auto-weighted projective low-rank and sparse recovery for visual representation. *Neural Networks*, 117: 201–215, 2019.

Xiaotao Wang and Fang Liu. Weighted low-rank representation-based dimension reduction for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 14(11):1938–1942, 2017.

Jie Wen, Bob Zhang, Yong Xu, Jian Yang, and Na Han. Adaptive weighted nonnegative low-rank representation. *Pattern Recognition*, 81:326–340, 2018.

Karl Werner and Magnus Jansson. Reduced rank linear regression and weighted low rank approxima-

tions. *IEEE transactions on signal processing*, 54 (6):2063–2075, 2006.

Tyler Wilson, Pang-Ning Tan, and Lifeng Luo. A low rank weighted graph convolutional approach to weather prediction. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 627–636. IEEE, 2018.

David Woodruff and Taisuke Yasuda. Reweighted solutions for weighted low rank approximation. In *International Conference on Machine Learning*. PMLR, 2024.

Qidi Wu, Yibing Li, Yun Lin, and Ruolin Zhou. Weighted sparse image classification based on low rank representation. *Computers, Materials & Continua*, 56(1), 2018.

Zhengqin Xu, Huasong Xing, Shun Fang, Shiqian Wu, and Shoulie Xie. Double-weighted low-rank matrix recovery based on rank estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 172–180, 2021.

Zhuoyan Xu, Zhenmei Shi, and Yingyu Liang. Do large language models have compositional ability? an investigation into limitations and scalability. In *Conference on Language Modeling*, 2024.

Baturalp Yalcin, Haixiang Zhang, Javad Lavaei, and Somayeh Sojoudi. Factorization approach for low-complexity matrix completion problems: Exponential number of spurious solutions and failure of gradient methods. In *International Conference on Artificial Intelligence and Statistics*, pages 319–341. PMLR, 2022.

Haixiang Zhang, Baturalp Yalcin, Javad Lavaei, and Somayeh Sojoudi. A new complexity metric for nonconvex rank-one generalized matrix completion. *Mathematical Programming*, 207(1):227–268, 2024.

Lichen Zhang. Speeding up optimizations via data structures: Faster search, sample and maintenance. Master's thesis, Carnegie Mellon University, 2022.

Yang Zhang, Yao Wang, Zhi Han, Yandong Tang, et al. Effective tensor completion via element-wise weighted low-rank tensor train with overlapping ket augmentation. *IEEE Transactions on circuits and systems for video technology*, 32(11):7286–7300, 2022.

Jianwei Zheng, Mengjie Qin, Xiaolong Zhou, Jiafa Mao, and Hongchuan Yu. Efficient implementation of truncated reweighting low-rank matrix approximation. *IEEE Transactions on Industrial Informatics*, 16(1):488–500, 2019.

Chenyang Li[1], Yingyu Liang[2,3], Zhenmei Shi[3], Zhao Song[4]

Checklist

1. For all models and algorithms presented, check if you include:

   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]

   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]

   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Not Applicable]

2. For any theoretical claim, check if you include:

   (a) Statements of the full set of assumptions of all theoretical results. [Yes]

   (b) Complete proofs of all theoretical results. [Yes]

   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:

   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Not Applicable]

   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Not Applicable]

   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Not Applicable]

   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Not Applicable]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

   (a) Citations of the creator if your work uses existing assets. [Not Applicable]

   (b) The license information of the assets, if applicable. [Not Applicable]

   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]

   (d) Information about consent from data providers/curators. [Not Applicable]

   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

   (a) The full text of instructions given to participants and screenshots. [Not Applicable]

   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]