Nonnegative Matrix Factorization

Example of NMF

Example of NMF

Given : $A \in \mathbb{R}^{n \times n}$, $n = 4$, $k = 2$

Example of NMF

Given : $A \in \mathbb{R}^{n \times n}$, $n = 4$, $k = 2$

$$A = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 2 & 4 & 2 & 2 \\ 3 & 5 & 2 & 2 \end{bmatrix}$$

Example of NMF

Given : $A \in \mathbb{R}^{n \times n}$, $n = 4$, $k = 2$

Question : Are there two matrices $U, V^\top \in \mathbb{R}^{n \times k}$, such that
$UV = A, U \geqslant 0, V \geqslant 0$

$$A = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 2 & 4 & 2 & 2 \\ 3 & 5 & 2 & 2 \end{bmatrix}$$

Example of NMF

Given : $A \in \mathbb{R}^{n \times n}$, $n = 4$, $k = 2$

Question : Are there two matrices $U, V^\top \in \mathbb{R}^{n \times k}$, such that
$UV = A, U \geqslant 0, V \geqslant 0$

$$
\begin{aligned}
A &= \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 2 & 4 & 2 & 2 \\ 3 & 5 & 2 & 2 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 2 & 2 \\ 3 & 2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}
\end{aligned}
$$

Example of NMF

Given : $A \in \mathbb{R}^{n \times n}$, $n = 4$, $k = 2$

Question : Are there two matrices $U, V^\top \in \mathbb{R}^{n \times k}$, such that
$UV = A, U \geqslant 0, V \geqslant 0$

$$
\begin{aligned}
A &= \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 \\ 2 & 4 & 2 & 2 \\ 3 & 5 & 2 & 2 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ 2 & 2 \\ 3 & 2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \\
&= UV
\end{aligned}
$$

Main Question, Nonnegative Matrix Factorization

Question

Given matrix $A \in \mathbb{R}^{n \times n}$ and $k \geqslant 1$, is there an algorithm that can determine if there exist two matrices $U, V^\top \in \mathbb{R}^{n \times k}$,

$$UV = A, U \geqslant 0, V \geqslant 0.$$

Or, are there any hardness results?

Main Question, Nonnegative Matrix Factorization

Question

Given matrix $A \in \mathbb{R}^{n\times n}$ and $k \geqslant 1$, is there an algorithm that can determine if there exist two matrices $U, V^\top \in \mathbb{R}^{n\times k}$,

$$UV = A, U \geqslant 0, V \geqslant 0.$$

Or, are there any hardness results?

- Equivalent to computing the nonnegative rank of A, $\mathrm{rank}_+(A)$

Main Question, Nonnegative Matrix Factorization

Question

Given matrix $A \in \mathbb{R}^{n \times n}$ and $k \geqslant 1$, is there an algorithm that can determine if there exist two matrices $U, V^\top \in \mathbb{R}^{n \times k}$,

$$UV = A, U \geqslant 0, V \geqslant 0.$$

Or, are there any hardness results?

- Equivalent to computing the nonnegative rank of A, $\mathrm{rank}_+(A)$
- Fundamental question in machine learning

Main Question, Nonnegative Matrix Factorization

Question

Given matrix $A \in \mathbb{R}^{n \times n}$ and $k \geqslant 1$, is there an algorithm that can determine if there exist two matrices $U, V^\top \in \mathbb{R}^{n \times k}$,

$$UV = A, U \geqslant 0, V \geqslant 0.$$

Or, are there any hardness results?

- Equivalent to computing the nonnegative rank of A, $\mathrm{rank}_+(A)$
- Fundamental question in machine learning
- Applications

Main Question, Nonnegative Matrix Factorization

Question

Given matrix $A \in \mathbb{R}^{n \times n}$ and $k \geqslant 1$, is there an algorithm that can determine if there exist two matrices $U, V^\top \in \mathbb{R}^{n \times k}$,

$$UV = A, U \geqslant 0, V \geqslant 0.$$

Or, are there any hardness results?

- Equivalent to computing the nonnegative rank of A, $\mathrm{rank}_+(A)$
- Fundamental question in machine learning
- Applications
  - Text mining, Spectral data analysis, Scalable Internet distance prediction, Non-stationary speech denoising, Bioinformatics, Nuclear imaging, etc.

Rank vs. Nonnegative Rank

- $\mathrm{rank}(A)$ is the rank and $\mathrm{rank}_+(A)$ is the nonnegative rank

Rank vs. Nonnegative Rank

- $\mathrm{rank}(A)$ is the rank and $\mathrm{rank}_+(A)$ is the nonnegative rank
  - $\mathrm{rank}(A) \leqslant \mathrm{rank}_+(A)$

Rank vs. Nonnegative Rank

- $\text{rank}(A)$ is the rank and $\text{rank}_+(A)$ is the nonnegative rank
  - $\text{rank}(A) \leqslant \text{rank}_+(A)$
  - Vavasis'09, determining whether $\text{rank}_+(A) = \text{rank}(A)$ is NP-hard.

Rank vs. Nonnegative Rank

- $\mathrm{rank}(A)$ is the rank and $\mathrm{rank}_+(A)$ is the nonnegative rank
  - $\mathrm{rank}(A) \leqslant \mathrm{rank}_+(A)$
  - Vavasis'09, determining whether $\mathrm{rank}_+(A) = \mathrm{rank}(A)$ is NP-hard.

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Rank vs. Nonnegative Rank

- $\mathrm{rank}(A)$ is the rank and $\mathrm{rank}_+(A)$ is the nonnegative rank
  - $\mathrm{rank}(A) \leqslant \mathrm{rank}_+(A)$
  - Vavasis'09, determining whether $\mathrm{rank}_+(A) = \mathrm{rank}(A)$ is NP-hard.

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathrm{rank}(A) = 3,$$

Rank vs. Nonnegative Rank

- $\mathrm{rank}(A)$ is the rank and $\mathrm{rank}_+(A)$ is the nonnegative rank
  - $\mathrm{rank}(A) \leqslant \mathrm{rank}_+(A)$
  - Vavasis'09, determining whether $\mathrm{rank}_+(A) = \mathrm{rank}(A)$ is NP-hard.

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\mathrm{rank}(A) = 3, \quad \text{but } \mathrm{rank}_+(A) = 4$$

Polynomial System Verifier

Polynomial System Verifier

[Renegar'92, Basu-Pollack-Roy'96]

Polynomial System Verifier
[Renegar'92, Basu-Pollack-Roy'96]

Given : a polynomial system $P(x)$ over the real numbers

Polynomial System Verifier
[Renegar'92, Basu-Pollack-Roy'96]

Given : a polynomial system $P(x)$ over the real numbers
v : #variables, $x = (x_1, x_2, \cdots, x_v)$

Polynomial System Verifier
[Renegar'92, Basu-Pollack-Roy'96]

Given : a polynomial system $P(x)$ over the real numbers

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

m : #polynomial constraints $f_i(x) \geqslant 0, \forall i \in [m]$

Polynomial System Verifier
[Renegar'92, Basu-Pollack-Roy'96]

Given : a polynomial system $P(x)$ over the real numbers

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

m : #polynomial constraints $f_i(x) \geqslant 0, \forall i \in [m]$

d : maximum degree of all polynomial constraints

Polynomial System Verifier

[Renegar'92, Basu-Pollack-Roy'96]

Given : a polynomial system $P(x)$ over the real numbers

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

m : #polynomial constraints $f_i(x) \geqslant 0, \forall i \in [m]$

d : maximum degree of all polynomial constraints

H : the bitsizes of the coefficients of the polynomials

Polynomial System Verifier

[Renegar'92, Basu-Pollack-Roy'96]

Given : a polynomial system $P(x)$ over the real numbers

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

m : #polynomial constraints $f_i(x) \geqslant 0, \forall i \in [m]$

d : maximum degree of all polynomial constraints

H : the bitsizes of the coefficients of the polynomials

In $(md)^{O(v)} \operatorname{poly}(H)$ time, can
decide if there exists a solution to polynomial system P

Main Idea

1. Write $\min_{U,V^\top \in \mathbb{R}^{n\times k}, U,V \geqslant 0} \|UV - A\|_F^2$ as a polynomial system that has $\mathrm{poly}(k)$ variables and $\mathrm{poly}(n)$ constraints and degree

2. Use polynomial system verifier to solve it

Algorithm

Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Question :

Algorithm

Given : $A \in \mathbb{R}^{n\times n}$, $k \in \mathbb{N}$

Question : Are there matrices $U, V^\top \in \mathbb{R}^{n\times k}$ such that

Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Question : Are there matrices U, $V^\top \in \mathbb{R}^{n \times k}$ such that $UV = A$

Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Question : Are there matrices $U, V^\top \in \mathbb{R}^{n \times k}$ such that
$UV = A$
$U \geqslant 0$, $V \geqslant 0$

Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Question : Are there matrices $U, V^\top \in \mathbb{R}^{n \times k}$ such that
$UV = A$
$U \geqslant 0$, $V \geqslant 0$

Output :

Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Question : Are there matrices U, $V^\top \in \mathbb{R}^{n \times k}$ such that
$UV = A$
$U \geqslant 0$, $V \geqslant 0$

Output : Yes or No

Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Question : Are there matrices $U, V^\top \in \mathbb{R}^{n \times k}$ such that
$UV = A$
$U \geqslant 0$, $V \geqslant 0$

Output : Yes or No
in $n^{2^{O(k)}}$ time Arora-Ge-Kannan-Moitra'12

Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Question : Are there matrices $U, V^\top \in \mathbb{R}^{n \times k}$ such that
$UV = A$
$U \geqslant 0$, $V \geqslant 0$

Output : Yes or No
in $n^{2^{O(k)}}$ time Arora-Ge-Kannan-Moitra'12
in $2^{O(k^3)} n^{O(k^2)}$ time Moitra'13

k-SUM

k-SUM

k-SUM

k-SUM

k-SUM

Given: a set of n values $\{s_1, s_2, \cdots, s_n\}$ each in the range $[0, 1]$

k-SUM

k-SUM

Given: a set of n values $\{s_1, s_2, \cdots, s_n\}$ each in the range $[0, 1]$

Determine:

k-SUM

k-SUM

Given: a set of n values $\{s_1, s_2, \cdots, s_n\}$ each in the range $[0, 1]$

Determine: if there is a set of k numbers that sum to exactly $k/2$

k-SUM

k-SUM

Given: a set of n values $\{s_1, s_2, \cdots, s_n\}$ each in the range $[0, 1]$

Determine: if there is a set of k numbers that sum to exactly $k/2$

k-SUM hardness [Patrascu-Williams'10]

k-SUM

k-SUM

Given: a set of n values $\{s_1, s_2, \cdots, s_n\}$ each in the range $[0, 1]$

Determine: if there is a set of k numbers that sum to exactly $k/2$

k-SUM hardness [Patrascu-Williams'10]

Assume: 3-SAT on n variables cannot be solved in $2^{o(n)}$ time

k-SUM

k-SUM

Given: a set of n values $\{s_1, s_2, \cdots, s_n\}$ each in the range $[0, 1]$

Determine: if there is a set of k numbers that sum to exactly $k/2$

k-SUM hardness [Patrascu-Williams'10]

Assume: 3-SAT on n variables cannot be solved in $2^{o(n)}$ time

then k-SUM cannot be solved in $n^{o(k)}$ time

Hardness - Under Exponential Time Hypothesis

Hardness - Under Exponential Time Hypothesis

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Hardness - Under Exponential Time Hypothesis

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Output : Are there two matrices $U, V^\top \in \mathbb{R}^{n \times k}$, such that
$UV = A$
$U \geqslant 0$, $V \geqslant 0$

Hardness - Under Exponential Time Hypothesis

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Output : Are there two matrices $U, V^\top \in \mathbb{R}^{n \times k}$, such that
$UV = A$
$U \geqslant 0, V \geqslant 0$

Assume : Exponential Time Hypothesis
[Impagliazzo-Paturi-Zane'98]

Hardness - Under Exponential Time Hypothesis

Given : $A \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$

Output : Are there two matrices $U, V^\top \in \mathbb{R}^{n \times k}$, such that
$UV = A$
$U \geqslant 0$, $V \geqslant 0$

Assume : Exponential Time Hypothesis
[Impagliazzo-Paturi-Zane'98]

Requires : $n^{\Omega(k)}$ time

Open Problems

- The upper bound is $n^{O(k^2)}$ while the lower bound is $n^{\Omega(k)}$ - what is the right answer?

Weighted Low Rank Approximation

Weighted Low Rank Approximation

Weighted Low Rank Approximation

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $k \in \mathbb{N}$, $\epsilon > 0$

Weighted Low Rank Approximation

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output :

Weighted Low Rank Approximation

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.

$$\left\| W \circ (\widehat{A} - A) \right\|_F^2$$

Weighted Low Rank Approximation

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.

$$\| W \circ (\widehat{A} - A) \|_F^2 \leqslant (1 + \epsilon) \min_{\text{rank}-k\ A'} \| W \circ (A' - A) \|_F^2$$

$$\left\| W \circ (\widehat{A} - A) \right\|_F^2$$

Weighted Low Rank Approximation

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.

$$\| W \circ (\widehat{A} - A) \|_F^2 \leqslant (1 + \epsilon) \min_{\text{rank}-k \ A'} \underbrace{\| W \circ (A' - A) \|_F^2}_{\sum_{i,j} W_{i,j}^2 (A'_{i,j} - A_{i,j})^2}$$

$$\left\| W \circ (\widehat{A} - A) \right\|_F^2$$

Weighted Low Rank Approximation

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.

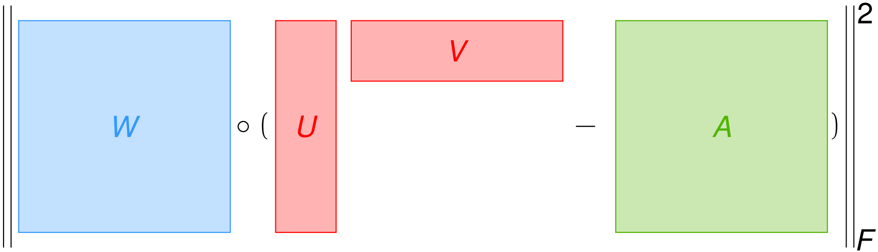
$$\| W \circ (\widehat{A} - A) \|_F^2 \leqslant (1 + \epsilon) \overbrace{\min_{\text{rank}-k\ A'} \underbrace{\| W \circ (A' - A) \|_F^2}_{\sum_{i,j} W_{i,j}^2 (A'_{i,j} - A_{i,j})^2}}^{\text{OPT}}$$

$$\left\| \boxed{W} \circ \left(\boxed{\widehat{A}} - \boxed{A} \right) \right\|_F^2$$

Weighted Low Rank Approximation

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t.
$\| W \circ (UV - A) \|_F^2 \leqslant (1 + \epsilon)\, \mathrm{OPT}$



Motivation

Motivation


DIVIDED WE FALL
CIVIL WAR
MAY 6
JIM CARREY
MASK
FROM ZERO TO HERO
HANGOVER
LINCOLN
ZOOTOPIA
LORD OF THE RINGS

Motivation


DIVIDED WE FALL
CIVIL WAR
JIM CARREY
MASK
FROM ZERO TO HERO
HANGOVER
LINCOLN
ZOOTOPIA
Harry Potter
LORD OF THE RINGS

Motivation

Action Comedy Historical Cartoon Magical

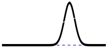
Motivation

Action Comedy Historical Cartoon Magical

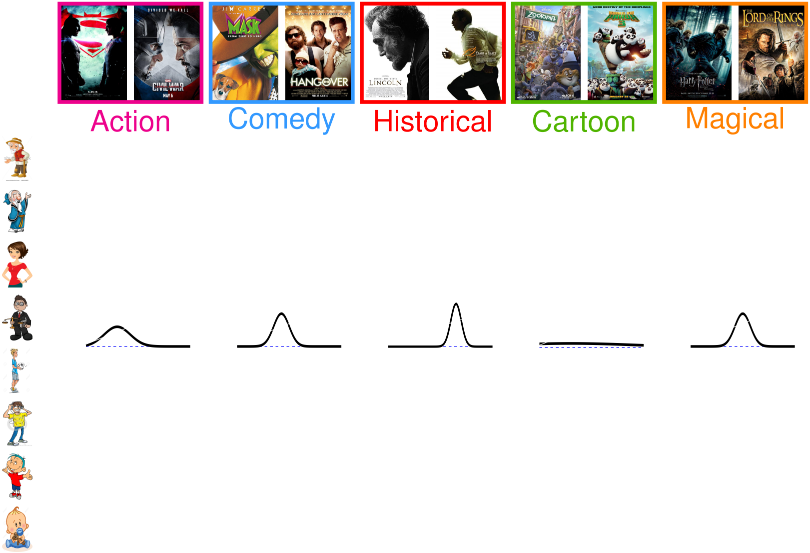
9 8

Motivation

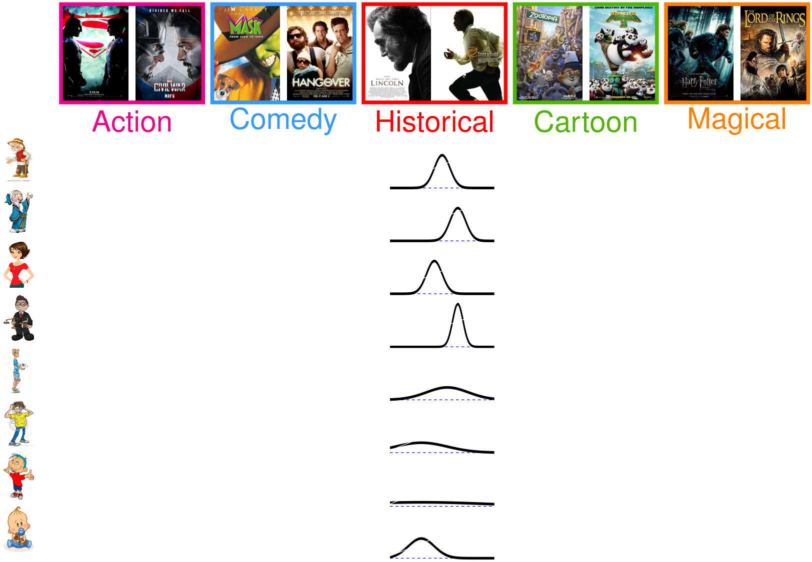
Action Comedy Historical Cartoon Magical



Motivation



Motivation



Motivation

Action Comedy Historical Cartoon Magical

$\frac{1}{5}$	$\frac{1}{5}$
$\frac{1}{1}$	$\frac{1}{1}$
$\frac{1}{1}$	$\frac{1}{1}$
$\frac{1}{5}$	$\frac{1}{5}$
$\frac{1}{1}$	$\frac{1}{1}$
$\frac{1}{4}$	$\frac{1}{4}$
$\frac{1}{6}$	$\frac{1}{6}$
$\frac{1}{1}$	$\frac{1}{1}$

Motivation

Action		Comedy		Historical		Cartoon		Magical	
$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{1}$	$\frac{1}{1}$						
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$						
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$						
$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{6}$						
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{3}$	$\frac{1}{3}$						
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{2}$						
$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{2}$						
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$						

Motivation

	Action		Comedy		Historical		Cartoon	Magical
	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{3}$	$\frac{1}{3}$		
	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{6}$	$\frac{1}{6}$		
	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$		
	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{2}$		
	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{7}$	$\frac{1}{7}$		
	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{5}$	$\frac{1}{5}$		
	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$		
	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{1}$	$\frac{1}{1}$		

Motivation

Action		Comedy		Historical		Cartoon		Magical
$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{2}$	$\frac{1}{2}$	
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{1}$	$\frac{1}{1}$	
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{2}$	
$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{1}$	$\frac{1}{1}$	
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{7}$	$\frac{1}{7}$	$\frac{1}{5}$	$\frac{1}{5}$	
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{1}$	$\frac{1}{1}$	
$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{3}$	
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$	

Motivation

Action		Comedy		Historical		Cartoon		Magical	
$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{2}$	$\frac{1}{2}$
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{6}$
$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{7}$	$\frac{1}{7}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{2}$	$\frac{1}{2}$
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{3}$	$\frac{1}{3}$
$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$	$\frac{1}{6}$
$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{1}$	$\frac{1}{1}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{2}$

Motivation

Action Comedy Historical Cartoon Magical

Motivation

Action Comedy Historical Cartoon Magical

8 8

8 8

Motivation

Action Comedy Historical Cartoon Magical

8	8
8	8
4	4
4	4

Motivation

Action Comedy Historical Cartoon Magical

8	8
8	8
4	4
4	4
1	1
1	1

Motivation

Action		Comedy		Historical		Cartoon		Magical	
				8	8				
				8	8				
				4	4				
				4	4				
				1	1				
				1	1				
				0	0				
				0	0				

Motivation

Action		Comedy		Historical		Cartoon		Magical	
				8	8				
				8	8				
				4	4				
				4	4				
				1	1				
				1	1				
				0	0	8	8		
				0	0	8	8		

Motivation

Action		Comedy		Historical		Cartoon		Magical	
				8	8				
				8	8				
				4	4				
				4	4				
				1	1	4	4		
				1	1	4	4		
				0	0	8	8		
				0	0	8	8		

Motivation

Action Comedy Historical Cartoon Magical

				8	8				
				8	8				
				4	4	2	2		
				4	4	2	2		
				1	1	4	4		
				1	1	4	4		
				0	0	8	8		
				0	0	8	8		

Motivation

Action

Comedy

Historical

Cartoon

Magical

8	8	1	1
8	8	1	1
4	4	2	2
4	4	2	2
1	1	4	4
1	1	4	4
0	0	8	8
0	0	8	8

Matrix Completion

Matrix Completion

Given : $A \in \{\mathbb{R}, ?\}^{n \times n}$, $\Omega \subset [n] \times [n]$ and $k \in \mathbb{R}$

Matrix Completion

Given : $A \in \{\mathbb{R}, ?\}^{n \times n}$, $\Omega \subset [n] \times [n]$ and $k \in \mathbb{R}$

$$A = \begin{bmatrix} 1 & ? & ? & 0 \\ ? & 1 & 0 & ? \\ ? & 0 & 1 & ? \\ 0 & ? & ? & 1 \end{bmatrix}, k = 2$$

Matrix Completion

Given : $A \in \{\mathbb{R}, ?\}^{n\times n}$, $\Omega \subset [n] \times [n]$ and $k \in \mathbb{R}$

$$A = \begin{bmatrix} 1 & ? & ? & 0 \\ ? & 1 & 0 & ? \\ ? & 0 & 1 & ? \\ 0 & ? & ? & 1 \end{bmatrix}, k = 2$$

Output : $A_{\bar{\Omega}}$ s.t. $\mathrm{rank}(A) = k$

Matrix Completion

Given : $A \in \{\mathbb{R}, ?\}^{n \times n}$, $\Omega \subset [n] \times [n]$ and $k \in \mathbb{R}$

$$A = \begin{bmatrix} 1 & ? & ? & 0 \\ ? & 1 & 0 & ? \\ ? & 0 & 1 & ? \\ 0 & ? & ? & 1 \end{bmatrix}, k = 2$$

Output : $A_{\bar{\Omega}}$ s.t. $\mathrm{rank}(A) = k$

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \; A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

Matrix Completion

Given : $A \in \{\mathbb{R}, ?\}^{n \times n}$, $\Omega \subset [n] \times [n]$ and $k \in \mathbb{R}$

$$A = \begin{bmatrix} 1 & ? & ? & 0 \\ ? & 1 & 0 & ? \\ ? & 0 & 1 & ? \\ 0 & ? & ? & 1 \end{bmatrix}, k = 2$$

Output : $A_{\bar{\Omega}}$ s.t. $\mathrm{rank}(A) = k$

Algorithms : Candes-Recht'09, Candes-Recht'10, Keshavan'12
Hardt-Wootters'14, Jain-Netrapalli-Sanghavi'13
Hardt'15, Sun-Luo'15

Matrix Completion

Given : $A \in \{\mathbb{R}, ?\}^{n \times n}$, $\Omega \subset [n] \times [n]$ and $k \in \mathbb{R}$

$$A = \begin{bmatrix} 1 & ? & ? & 0 \\ ? & 1 & 0 & ? \\ ? & 0 & 1 & ? \\ 0 & ? & ? & 1 \end{bmatrix}, k = 2$$

Output : $A_{\bar{\Omega}}$ s.t. $\mathrm{rank}(A) = k$

Algorithms : Candes-Recht'09, Candes-Recht'10, Keshavan'12
Hardt-Wootters'14, Jain-Netrapalli-Sanghavi'13
Hardt'15, Sun-Luo'15

Hardness : Peeters'96, Gillis-Glineur'11
Hardt-Meka-Raghavendra-Weitz'14

More Real-life Datasets

More Real-life Datasets

RateBeer

Tasters

Beers


Heaven Or Helles
Boston Beer Company (Samuel Adams)
Munich Helles Lager / 5.00% ABV

3.71/5 rDev +0.5% | Avg: 3.69
look: 3.25 | smell: 3.5 | taste: 4 | feel: 3.75 | overall: 3.5

Pours a slightly hazy golden straw color with a pillowy 1-1 1/2 inch head that slowly dissipates to slightly less than 1 finger of head with good lacing. The nose had aromas of bread malt and a spiciness I belie cv is pepper. The mouthfeel was lightly carbonated with a richness/sweetness. A slight bitterness on the back end rounded out the flavor palate and struck a nice balance. I found it very satisfying and I'd go as far as saying it would make a nice session beer. The only thing I don't like is that it is a limited release.

★ 533 characters

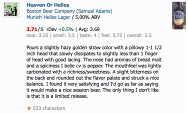
Ratings

More Real-life Datasets

RateBeer

Tasters

Beers

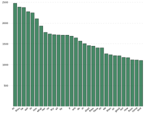


Ratings

Documents

Words

Books



Frequency

More Real-life Datasets

RateBeer

Tasters

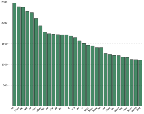
Beers

Ratings

Documents

Words

Books



Frequency

Amazon

Users

Products

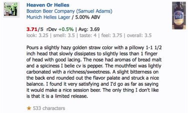
Ratings

More Real-life Datasets

RateBeer

Tasters

Beers

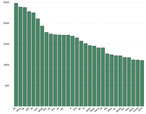


Ratings

Documents

Words

Books



Frequency

Amazon

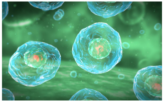
Users

Products

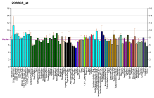
Ratings

Biology

Genes



Cells



Levels

Several Results in [Razenshteyn, Song, W]

- Algorithm for Weighted low rank approximation(WLRA) problem

Several Results in [Razenshteyn, Song, W]

- Algorithm for Weighted low rank approximation(WLRA) problem
  - W has r distinct rows and columns

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Several Results in [Razenshteyn, Song, W]

- Algorithm for Weighted low rank approximation(WLRA) problem
  - W has r distinct rows and columns
  - W has r distinct columns

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Several Results in [Razenshteyn, Song, W]

- Algorithm for Weighted low rank approximation(WLRA) problem
  - W has r distinct rows and columns
  - W has r distinct columns
  - W has rank at most r

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Several Results in [Razenshteyn, Song, W]

- Algorithm for Weighted low rank approximation(WLRA) problem
  - W has r distinct rows and columns
  - W has r distinct columns
  - W has rank at most r
- Hardness for Weighted low rank approximation(WLRA) problem

r Distinct Rows and Columns

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Output :

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.

$$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon) \min_{\text{rank}-k\ A'} \|W \circ (A - A')\|_F^2$$

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.

$$\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon) \min_{\text{rank}-k\ A'} \sum_{i,j} W_{i,j}^2 (A_{i,j} - A'_{i,j})^2$$

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.

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

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.
$\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1+\epsilon)\mathrm{OPT}$
with prob. $9/10$

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.
$\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon)\mathrm{OPT}$
with prob. $9/10$
in $O((\mathrm{nnz}(A) + \mathrm{nnz}(W)) \cdot n^{\gamma}) + n \cdot 2^{\widetilde{O}(k^2 r/\epsilon)}$ time

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, \; r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.
$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1 + \epsilon)\mathrm{OPT}$
with prob. $9/10$
in $O((\mathrm{nnz}(A) + \mathrm{nnz}(W)) \cdot n^{\gamma}) + n \cdot 2^{\widetilde{O}(k^2 r/\epsilon)}$ time
for an arbitrarily small constant $\gamma > 0$

r Distinct Rows and Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 3 & 3 & 4 \\ 1 & 1 & 1 & 3 & 3 & 4 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 2 & 2 & 2 & 5 & 5 & 0 \\ 7 & 7 & 7 & 4 & 4 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.
$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\mathrm{OPT}$
with prob. $9/10$
in $O((\mathrm{nnz}(A) + \mathrm{nnz}(W)) \cdot n^{\gamma}) + n \cdot 2^{\widetilde{O}(k^2 r/\epsilon)}$ time
for an arbitrarily small constant $\gamma > 0$
fixed parameter tractable

r Distinct Columns

r Distinct Columns

Given : $\quad A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

r Distinct Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

r Distinct Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output :

r Distinct Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.

r Distinct Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.

$$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon) \min_{\mathrm{rank}-k\ A'} \|W \circ (A - A')\|_F^2$$

r Distinct Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.

$$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon) \min_{\text{rank}-k\ A'} \sum_{i,j} W_{i,j}^2 (A_{i,j} - A'_{i,j})^2$$

r Distinct Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.
$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\mathrm{OPT}$

r Distinct Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.
$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\mathrm{OPT}$
with prob. $9/10$

r Distinct Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.
$\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon)\mathrm{OPT}$
with prob. $9/10$
in $O((\mathrm{nnz}(A) + \mathrm{nnz}(W)) \cdot n^{\gamma}) + n \cdot 2^{\widetilde{O}(k^2 r^2 / \epsilon)}$ time

r Distinct Columns

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.
$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\mathrm{OPT}$
with prob. $9/10$
in $O((\mathrm{nnz}(A) + \mathrm{nnz}(W)) \cdot n^{\gamma}) + n \cdot 2^{\widetilde{O}(k^2 r^2/\epsilon)}$ time
for an arbitrarily small constant $\gamma > 0$

r Distinct Columns

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 3 \\ 3 & 3 & 3 & 1 & 1 & 5 \\ 4 & 4 & 4 & 1 & 1 & 1 \\ 5 & 5 & 5 & 1 & 1 & 3 \\ 6 & 6 & 6 & 1 & 1 & 5 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.
$\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon) \mathrm{OPT}$
with prob. $9/10$
in $O((\mathrm{nnz}(A) + \mathrm{nnz}(W)) \cdot n^\gamma) + n \cdot 2^{\widetilde{O}(k^2 r^2 / \epsilon)}$ time
for an arbitrarily small constant $\gamma > 0$
fixed parameter tractable

Rank r

Rank r

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Rank r

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Rank r

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Output :

Rank r

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.

Rank r

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.

$$\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon) \min_{\mathrm{rank}-k\ A'} \| W \circ (A - A') \|_F^2$$

Rank r

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.

$$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon) \min_{\text{rank}-k\ A'} \sum_{i,j} W_{i,j}^2 (A_{i,j} - A'_{i,j})^2$$

Rank r

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, \ r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.

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

Rank r

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.
$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\mathrm{OPT}$
with prob. $9/10$

Rank r

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n\times n}$ s.t.
$\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\mathrm{OPT}$
with prob. $9/10$
in $n^{O(k^2 r/\epsilon)}$ time

Rank r

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}, r = 3$$

Output : rank-k $\widehat{A} \in \mathbb{R}^{n \times n}$ s.t.
$\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon) \mathrm{OPT}$
with prob. $9/10$
in $n^{O(k^2 r/\epsilon)}$ time
previously only r=1 was known to be in polynomial time

Random-4SAT Hypothesis

Random-4SAT Hypothesis

[Feige'02, Goerdt-Lanka'04]

Random-4SAT Hypothesis

[Feige'02, Goerdt-Lanka'04]

Given : random 4-SAT formula $\mathcal{S}$ on n variables

Random-4SAT Hypothesis

[Feige'02, Goerdt-Lanka'04]

Given : random 4-SAT formula $\mathcal{S}$ on n variables
each clause has 4 literals

Random-4SAT Hypothesis

[Feige'02, Goerdt-Lanka'04]

Given : random 4-SAT formula $\mathcal{S}$ on n variables
each clause has 4 literals
each of $\Theta(n^4)$ clauses is picked ind. with prob. $\Theta(1/n^3)$
$m = \Theta(n)$ is the number of clauses

S_1 $(x_4 \vee x_{32} \vee x_9 \vee x_{20})$

Random-4SAT Hypothesis

[Feige'02, Goerdt-Lanka'04]

Given : random 4-SAT formula $\mathcal{S}$ on n variables
each clause has 4 literals
each of $\Theta(n^4)$ clauses is picked ind. with prob. $\Theta(1/n^3)$
$m = \Theta(n)$ is the number of clauses

Any algorithm that :

$$
\begin{array}{ll}
S_1 & (x_4 \vee x_{32} \vee x_9 \vee x_{20}) \\
S_2 & (x_2 \vee x_{13} \vee x_5 \vee x_{25}) \\
 & \cdots\cdots \\
S_m & (x_{11} \vee x_{19} \vee x_{24} \vee x_6)
\end{array}
\qquad \mathcal{S} = S_1 \wedge S_2 \wedge \cdots \wedge S_m
$$

Random-4SAT Hypothesis

[Feige'02, Goerdt-Lanka'04]

Given : random 4-SAT formula $\mathcal{S}$ on n variables
each clause has 4 literals
each of $\Theta(n^4)$ clauses is picked ind. with prob. $\Theta(1/n^3)$
$m = \Theta(n)$ is the number of clauses

Any algorithm that :
outputs 1 with prob. 1 when $\mathcal{S}$ is satisfiable
outputs 0 with prob. $\geqslant 1/2$
probability is over the input instances

$$
\begin{array}{ll}
S_1 & (x_4 \vee x_{32} \vee x_9 \vee x_{20}) \\
S_2 & (x_2 \vee x_{13} \vee x_5 \vee x_{25}) \\
 & \cdots\cdots \\
S_m & (x_{11} \vee x_{19} \vee x_{24} \vee x_6)
\end{array}
\qquad \mathcal{S} = S_1 \wedge S_2 \wedge \cdots \wedge S_m
$$

Random-4SAT Hypothesis
[Feige'02, Goerdt-Lanka'04]

Given : random 4-SAT formula $\mathcal{S}$ on n variables
each clause has 4 literals
each of $\Theta(n^4)$ clauses is picked ind. with prob. $\Theta(1/n^3)$
$m = \Theta(n)$ is the number of clauses

Any algorithm that :
outputs 1 with prob. 1 when $\mathcal{S}$ is satisfiable
outputs 0 with prob. $\geqslant 1/2$
probability is over the input instances

Requires : $2^{\Omega(n)}$ time

$$\begin{array}{ll} S_1 & (x_4 \vee x_{32} \vee x_9 \vee x_{20}) \\ S_2 & (x_2 \vee x_{13} \vee x_5 \vee x_{25}) \\ & \cdots\cdots \\ S_m & (x_{11} \vee x_{19} \vee x_{24} \vee x_6) \end{array}$$

$$\mathcal{S} = S_1 \wedge S_2 \wedge \cdots \wedge S_m$$

Maximum Edge Biclique(MEB)

Maximum Edge Biclique(MEB)

Given : A bipartite graph $G = (U, V, E)$ with $|U| = |V| = n$

Maximum Edge Biclique(MEB)

Given : A bipartite graph $G = (U, V, E)$ with $|U| = |V| = n$

Output : A k_1-by-k_2 complete bipartite subgraph of G

Maximum Edge Biclique(MEB)

Given : A bipartite graph $G = (U, V, E)$ with $|U| = |V| = n$

Output : A k_1-by-k_2 complete bipartite subgraph of G
s.t. $k_1 \cdot k_2$ is maximized

Maximum Edge Biclique(MEB)

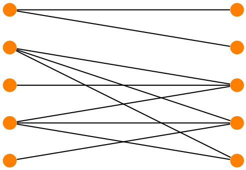
Given : A bipartite graph $G = (U, V, E)$ with $|U| = |V| = n$

Output : A k_1-by-k_2 complete bipartite subgraph of G
s.t. $k_1 \cdot k_2$ is maximized

Maximum Edge Biclique(MEB)

Given : A bipartite graph $G = (U, V, E)$ with $|U| = |V| = n$

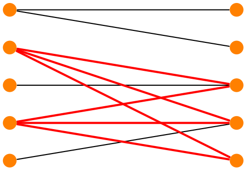
Output : A k_1-by-k_2 complete bipartite subgraph of G
s.t. $k_1 \cdot k_2$ is maximized



Maximum Edge Biclique(MEB)

Given : A bipartite graph $G = (U, V, E)$ with $|U| = |V| = n$

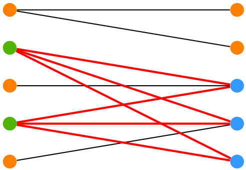
Output : A k_1-by-k_2 complete bipartite subgraph of G
s.t. $k_1 \cdot k_2$ is maximized



Maximum Edge Biclique(MEB)

Given : A bipartite graph $G = (U, V, E)$ with $|U| = |V| = n$

Output : A k_1-by-k_2 complete bipartite subgraph of G
s.t. $k_1 \cdot k_2$ is maximized



Maximum Edge Biclique(MEB)

[Goerdt-Lanka'04]

Maximum Edge Biclique(MEB)

[Goerdt-Lanka'04]

Assume : Random 4-SAT hardness hypothesis

Maximum Edge Biclique(MEB)

[Goerdt-Lanka'04]

Assume : Random 4-SAT hardness hypothesis

there exist two constants $\epsilon_1 > \epsilon_2 > 0$ such that

Maximum Edge Biclique(MEB)

[Goerdt-Lanka'04]

Assume : Random 4-SAT hardness hypothesis

there exist two constants $\epsilon_1 > \epsilon_2 > 0$ such that

Any algorithm that : distinguishes between

bipartite graphs $G = (U, V, E)$ with $|U| = |V| = n$ in two cases

Maximum Edge Biclique(MEB)

[Goerdt-Lanka'04]

Assume : Random 4-SAT hardness hypothesis

there exist two constants $\epsilon_1 > \epsilon_2 > 0$ such that

Any algorithm that : distinguishes between

bipartite graphs $G = (U, V, E)$ with $|U| = |V| = n$ in two cases

1. there is a clique of size $\geqslant (n/16)^2(1 + \epsilon_1)$

Maximum Edge Biclique(MEB)

[Goerdt-Lanka'04]

Assume : Random 4-SAT hardness hypothesis

there exist two constants $\epsilon_1 > \epsilon_2 > 0$ such that

Any algorithm that : distinguishes between

bipartite graphs $G = (U, V, E)$ with $|U| = |V| = n$ in two cases

1. there is a clique of size $\geqslant (n/16)^2(1 + \epsilon_1)$
2. all bipartite cliques are of size $\leqslant (n/16)^2(1 + \epsilon_2)$

Maximum Edge Biclique(MEB)
[Goerdt-Lanka'04]

Assume : Random 4-SAT hardness hypothesis

there exist two constants $\epsilon_1 > \epsilon_2 > 0$ such that

Any algorithm that : distinguishes between

bipartite graphs $G = (U, V, E)$ with $|U| = |V| = n$ in two cases

1. there is a clique of size $\geqslant (n/16)^2(1 + \epsilon_1)$
2. all bipartite cliques are of size $\leqslant (n/16)^2(1 + \epsilon_2)$

Requires :

Maximum Edge Biclique(MEB)

[Goerdt-Lanka'04]

Assume : Random 4-SAT hardness hypothesis

there exist two constants $\epsilon_1 > \epsilon_2 > 0$ such that

Any algorithm that : distinguishes between

bipartite graphs $G = (U, V, E)$ with $|U| = |V| = n$ in two cases

1. there is a clique of size $\geqslant (n/16)^2(1 + \epsilon_1)$
2. all bipartite cliques are of size $\leqslant (n/16)^2(1 + \epsilon_2)$

Requires : $2^{\Omega(n)}$ time

Reduction

Reduction

MEB

Reduction

MEB Given : U, V with $|U| = |V| = n$

Reduction

MEB	Given :	U, V with $\lvert U\rvert = \lvert V\rvert = n$
	Output :	a biclique s.t. $k_1 k_2$ is maximized

Reduction

MEB Given : U, V with $|U| = |V| = n$

Output : a biclique s.t. $k_1 k_2$ is maximized

(complement) $|E| - k_1 k_2$ is minimized

Reduction

MEB	Given :	U, V with $\lvert U\rvert = \lvert V\rvert = n$
	Output :	a biclique s.t. $k_1 k_2$ is maximized
		(complement) $\lvert E\rvert - k_1 k_2$ is minimized

WLRA

Reduction

MEB	Given :	U, V with $\lvert U\rvert = \lvert V\rvert = n$
	Output :	a biclique s.t. $k_1 k_2$ is maximized
		(complement) $\lvert E\rvert - k_1 k_2$ is minimized

WLRA	Given :	$A \in \{0,1\}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$

Reduction

MEB Given : U, V with $|U| = |V| = n$

Output : a biclique s.t. $k_1 k_2$ is maximized

(complement) $|E| - k_1 k_2$ is minimized

WLRA Given : $A \in \{0,1\}^{n \times n}, W \in \mathbb{R}^{n \times n}$

Output : $u, v \in \mathbb{R}^n$ s.t. $\|W \circ (uv^\top - A)\|_F^2$ is minimized

Reduction

MEB Given : U, V with $|U| = |V| = n$

Output : a biclique s.t. $k_1 k_2$ is maximized

(complement) $|E| - k_1 k_2$ is minimized

WLRA Given : $A \in \{0,1\}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$

Output : $u, v \in \mathbb{R}^n$ s.t. $\| W \circ (uv^\top - A) \|_F^2$ is minimized

Reduction

MEB Given : U, V with $|U| = |V| = n$

Output : a biclique s.t. $k_1 k_2$ is maximized

(complement) $|E| - k_1 k_2$ is minimized

$$A_{i,j} = \begin{cases} 1 & \text{if edge } (U_i, V_j) \text{ exists} \\ 0 & \text{otherwise} \end{cases}$$

WLRA Given : $A \in \{0,1\}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$

Output : $u, v \in \mathbb{R}^n$ s.t. $\|W \circ (uv^\top - A)\|_F^2$ is minimized

Reduction

MEB Given : U, V with $|U| = |V| = n$

Output : a biclique s.t. $k_1 k_2$ is maximized

(complement) $|E| - k_1 k_2$ is minimized

$$A_{i,j} = \begin{cases} 1 \\ 0 \end{cases} \quad W_{i,j} = \begin{cases} 1 & \text{if edge } (U_i, V_j) \text{ exists} \\ n^6 & \text{otherwise} \end{cases}$$

WLRA Given : $A \in \{0,1\}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$

Output : $u, v \in \mathbb{R}^n$ s.t. $\| W \circ (uv^\top - A) \|_F^2$ is minimized

Main Results - Hardness

Main Results - Hardness

Weighted Low Rank Approximation Hardness

Main Results - Hardness

Weighted Low Rank Approximation Hardness

Given : $A \in \mathbb{R}^{n \times n}$, distinct r columns $W \in \mathbb{R}^{n \times n}$

Main Results - Hardness

Weighted Low Rank Approximation Hardness

Given : $A \in \mathbb{R}^{n\times n}$, distinct r columns $W \in \mathbb{R}^{n\times n}$
$k \in \mathbb{N}$, $\epsilon > 0$, $W_{ij} \in \{0, 1, 2, \cdots, \text{poly}(n)\}$

Main Results - Hardness

Weighted Low Rank Approximation Hardness

Given : $A \in \mathbb{R}^{n \times n}$, distinct r columns $W \in \mathbb{R}^{n \times n}$
$k \in \mathbb{N}$, $\epsilon > 0$, $W_{ij} \in \{0, 1, 2, \cdots, \mathrm{poly}(n)\}$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$
with prob. $9/10$

Main Results - Hardness

Weighted Low Rank Approximation Hardness

Given : $A \in \mathbb{R}^{n\times n}$, distinct r columns $W \in \mathbb{R}^{n\times n}$
$k \in \mathbb{N}$, $\epsilon > 0$, $W_{ij} \in \{0, 1, 2, \cdots, \mathrm{poly}(n)\}$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$
with prob. $9/10$

Assume : Random-4SAT Hypothesis
[Feige'02, Goerdt-Lanka'04]

Main Results - Hardness

Weighted Low Rank Approximation Hardness

Given : $A \in \mathbb{R}^{n \times n}$, distinct r columns $W \in \mathbb{R}^{n \times n}$
$k \in \mathbb{N}$, $\epsilon > 0$, $W_{ij} \in \{0, 1, 2, \cdots, \mathrm{poly}(n)\}$

Output : rank-k $\widehat{A}$ s.t. $\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$
with prob. $9/10$

Assume : Random-4SAT Hypothesis
[Feige'02, Goerdt-Lanka'04]

$\exists \epsilon_0 > 0$, for any algorithm with $\epsilon < \epsilon_0$ and $k \geqslant 1$

Main Results - Hardness

Weighted Low Rank Approximation Hardness

Given : $A \in \mathbb{R}^{n \times n}$, distinct r columns $W \in \mathbb{R}^{n \times n}$
$k \in \mathbb{N}$, $\epsilon > 0$, $W_{ij} \in \{0, 1, 2, \cdots, \mathrm{poly}(n)\}$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$
with prob. $9/10$

Assume : Random-4SAT Hypothesis
[Feige'02, Goerdt-Lanka'04]

$\exists \epsilon_0 > 0$, for any algorithm with $\epsilon < \epsilon_0$ and $k \geqslant 1$

Requires : $2^{\Omega(r)}$ time

Algorithmic Techniques

- Tools

Algorithmic Techniques

- Tools
  - Polynomial system verifier [Renegar'92, Basu-Pollack-Roy'96]

Algorithmic Techniques

- Tools
  - Polynomial system verifier [Renegar'92, Basu-Pollack-Roy'96]
  - Lower bound on the cost [Jeronimo-Perrucci-Tsigaridas'13]

Algorithmic Techniques

- Tools
  - Polynomial system verifier [Renegar'92, Basu-Pollack-Roy'96]
  - Lower bound on the cost [Jeronimo-Perrucci-Tsigaridas'13]
  - Multiple regression sketch

Algorithmic Techniques

- Tools
  - Polynomial system verifier [Renegar'92, Basu-Pollack-Roy'96]
  - Lower bound on the cost [Jeronimo-Perrucci-Tsigaridas'13]
  - Multiple regression sketch
- Algorithm, assuming a rank r weight matrix W

Algorithmic Techniques

- Tools
  - Polynomial system verifier [Renegar'92, Basu-Pollack-Roy'96]
  - Lower bound on the cost [Jeronimo-Perrucci-Tsigaridas'13]
  - Multiple regression sketch
- Algorithm, assuming a rank r weight matrix W
  - Warmup, inefficient WLRA algorithm

Algorithmic Techniques

- Tools
  - Polynomial system verifier [Renegar'92, Basu-Pollack-Roy'96]
  - Lower bound on the cost [Jeronimo-Perrucci-Tsigaridas'13]
  - Multiple regression sketch
- Algorithm, assuming a rank r weight matrix W
  - Warmup, inefficient WLRA algorithm
  - "Guessing a sketch"

Polynomial System Verifier (Recall)

Polynomial System Verifier (Recall)

[Renegar'92, Basu-Pollack-Roy'96]

Polynomial System Verifier (Recall)
[Renegar'92, Basu-Pollack-Roy'96]

Given : a real polynomial system $P(x)$

Polynomial System Verifier (Recall)

[Renegar'92, Basu-Pollack-Roy'96]

Given : a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

Polynomial System Verifier (Recall)

[Renegar'92, Basu-Pollack-Roy'96]

Given : a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

m : #polynomial constraints $f_i(x) \geqslant 0, \forall i \in [m]$

Polynomial System Verifier (Recall)
[Renegar'92, Basu-Pollack-Roy'96]

Given : a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

m : #polynomial constraints $f_i(x) \geqslant 0, \forall i \in [m]$

d : maximum degree of all polynomial constraints

Polynomial System Verifier (Recall)

[Renegar'92, Basu-Pollack-Roy'96]

Given : a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

m : #polynomial constraints $f_i(x) \geqslant 0, \forall i \in [m]$

d : maximum degree of all polynomial constraints

H : the bitsizes of the coefficients of the polynomials

Polynomial System Verifier (Recall)
[Renegar'92, Basu-Pollack-Roy'96]

Given : a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

m : #polynomial constraints $f_i(x) \geqslant 0, \forall i \in [m]$

d : maximum degree of all polynomial constraints

H : the bitsizes of the coefficients of the polynomials

It takes $(md)^{O(v)} \operatorname{poly}(H)$ time to
decide if there exists a solution to polynomial system P

Lower Bound on the Cost

Lower Bound on the Cost

[Jeronimo-Perrucci-Tsigaridas'13]

Lower Bound on the Cost

[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

Lower Bound on the Cost

[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$

Lower Bound on the Cost
[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$

Lower Bound on the Cost

[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$
d : maximum degree of all polynomials

Lower Bound on the Cost
[Jeronimo-Perrucci-Tsigaridas'13]

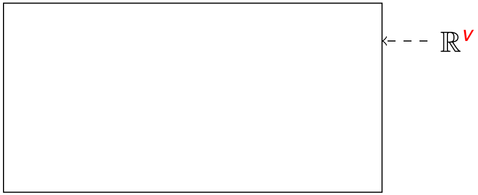
Given: a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$
d : maximum degree of all polynomials
H : the bitsizes of the coefficients of the polynomials

Lower Bound on the Cost
[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

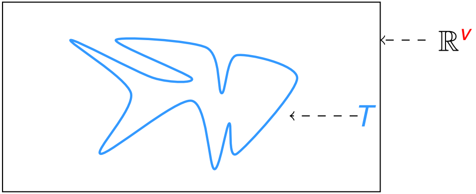
v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$
d : maximum degree of all polynomials
H : the bitsizes of the coefficients of the polynomials



Lower Bound on the Cost
[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$
d : maximum degree of all polynomials
H : the bitsizes of the coefficients of the polynomials
T : $\{x \in \mathbb{R}^v | f_1(x) \geqslant 0, \cdots, f_m(x) \geqslant 0\}$

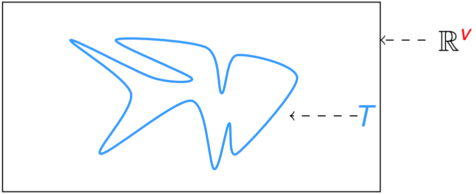


Lower Bound on the Cost

[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

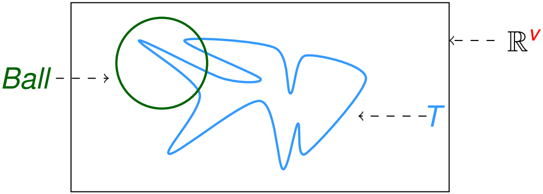
v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$
d : maximum degree of all polynomials
H : the bitsizes of the coefficients of the polynomials
T : $\{x \in \mathbb{R}^v | f_1(x) \geqslant 0, \cdots, f_m(x) \geqslant 0\}$
minimum value that nonnegative g takes over $T \cap$ *Ball* is



Lower Bound on the Cost
[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

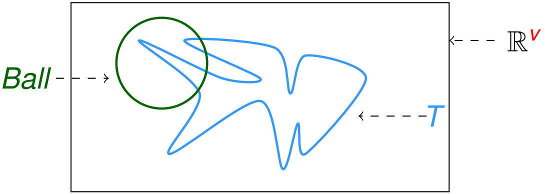
v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$
d : maximum degree of all polynomials
H : the bitsizes of the coefficients of the polynomials
T : $\{x \in \mathbb{R}^v | f_1(x) \geqslant 0, \cdots, f_m(x) \geqslant 0\}$
minimum value that nonnegative g takes over $T \cap Ball$ is



Lower Bound on the Cost
[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$
d : maximum degree of all polynomials
H : the bitsizes of the coefficients of the polynomials
T : $\{x \in \mathbb{R}^v | f_1(x) \geqslant 0, \cdots, f_m(x) \geqslant 0\}$
minimum value that nonnegative g takes over $T \cap Ball$ is either 0 or $\geqslant (2^H + m)^{-d^v}$

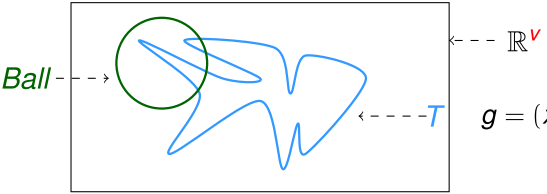


Lower Bound on the Cost
[Jeronimo-Perrucci-Tsigaridas'13]

Given: a real polynomial system $P(x)$

v : #variables, $x = (x_1, x_2, \cdots, x_v)$
m : #polynomials constraints $f_i(x) \geqslant 0, \forall i \in [m]$
d : maximum degree of all polynomials
H : the bitsizes of the coefficients of the polynomials
T : $\{x \in \mathbb{R}^v | f_1(x) \geqslant 0, \cdots, f_m(x) \geqslant 0\}$

minimum value that nonnegative g takes over $T \cap Ball$ is either 0 or $\geqslant (2^H + m)^{-d^v}$



$g = (x_1 x_2 - 1)^2 + x_2^2$

Multiple Regression Sketch

Multiple Regression Sketch

Given :

Multiple Regression Sketch

Given : $A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n \times k}$

Multiple Regression Sketch

Given :

$$A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n \times k}$$
$$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n \times 1}$$

Multiple Regression Sketch

Given : $$A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n\times k}$$
$$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n\times 1}$$

Let $x^{(j)} = \arg\min_{x\in\mathbb{R}^{k\times 1}} \|A^{(j)}x - b^{(j)}\|, \forall j \in [m]$

Multiple Regression Sketch

Given : $$A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n \times k}$$
$$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n \times 1}$$

Let $$x^{(j)} = \arg\min_{x \in \mathbb{R}^{k \times 1}} \| A^{(j)} x - b^{(j)} \|, \forall j \in [m]$$

Choose : S to be a random Gaussian matrix

Multiple Regression Sketch

Given : $A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n\times k}$

$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n\times 1}$

Let $x^{(j)} = \arg\min_{x\in\mathbb{R}^{k\times 1}} \|A^{(j)}x - b^{(j)}\|, \forall j \in [m]$

Choose : S to be a random Gaussian matrix

Denote $y^{(j)} = \arg\min_{y\in\mathbb{R}^{k\times 1}} \|SA^{(j)}y - Sb^{(j)}\|, \forall j \in [m]$

Multiple Regression Sketch

Given : $A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n\times k}$

$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n\times 1}$

Let $x^{(j)} = \arg\min_{x\in\mathbb{R}^{k\times 1}} \|A^{(j)}x - b^{(j)}\|, \forall j \in [m]$

Choose : S to be a random Gaussian matrix

Denote $y^{(j)} = \arg\min_{y\in\mathbb{R}^{k\times 1}} \|SA^{(j)}y - Sb^{(j)}\|, \forall j \in [m]$

Gaurantee :

Multiple Regression Sketch

Given : $A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n\times k}$

$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n\times 1}$

$$\text{Let } x^{(j)} = \arg\min_{x\in\mathbb{R}^{k\times 1}} \|A^{(j)}x - b^{(j)}\|, \forall j \in [m]$$

Choose : S to be a random Gaussian matrix

$$\text{Denote } y^{(j)} = \arg\min_{y\in\mathbb{R}^{k\times 1}} \|SA^{(j)}y - Sb^{(j)}\|, \forall j \in [m]$$

Gaurantee : For all $\epsilon \in (0, 1/2)$, one can set $t = O(k/\epsilon)$ such that

Multiple Regression Sketch

Given : $A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n \times k}$

$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n \times 1}$

Let $x^{(j)} = \arg\min_{x \in \mathbb{R}^{k \times 1}} \|A^{(j)} x - b^{(j)}\|, \forall j \in [m]$

Choose : S to be a random Gaussian matrix

Denote $y^{(j)} = \arg\min_{y \in \mathbb{R}^{k \times 1}} \|SA^{(j)} y - Sb^{(j)}\|, \forall j \in [m]$

Gaurantee : For all $\epsilon \in (0, 1/2)$, one can set $t = O(k/\epsilon)$ such that

$$\sum_{j=1}^{m} \|A^{(j)} y^{(j)} - b^{(j)}\|_2^2 \leqslant (1+\epsilon) \sum_{j=1}^{m} \|A^{(j)} x^{(j)} - b^{(j)}\|_2^2$$

Multiple Regression Sketch

Given : $A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n \times k}$

$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n \times 1}$

$$\text{Let } x^{(j)} = \arg\min_{x \in \mathbb{R}^{k \times 1}} \|A^{(j)} x - b^{(j)}\|, \forall j \in [m]$$

Choose : S to be a random Gaussian matrix

$$\text{Denote } y^{(j)} = \arg\min_{y \in \mathbb{R}^{k \times 1}} \|SA^{(j)} y - Sb^{(j)}\|, \forall j \in [m]$$

Gaurantee : For all $\epsilon \in (0, 1/2)$, one can set $t = O(k/\epsilon)$ such that

$$\sum_{j=1}^{m} \|A^{(j)} y^{(j)} - b^{(j)}\|_2^2 \leqslant (1+\epsilon) \sum_{j=1}^{m} \|A^{(j)} x^{(j)} - b^{(j)}\|_2^2$$

with prob. $9/10$

Multiple Regression Sketch

Given : $A^{(1)}, A^{(2)}, \cdots, A^{(m)} \in \mathbb{R}^{n \times k}$
$b^{(1)}, b^{(2)}, \cdots, b^{(m)} \in \mathbb{R}^{n \times 1}$

Let $x^{(j)} = \arg\min_{x \in \mathbb{R}^{k \times 1}} \|A^{(j)} x - b^{(j)}\|, \forall j \in [m]$

Choose : S to be a random Gaussian matrix

Denote $y^{(j)} = \arg\min_{y \in \mathbb{R}^{k \times 1}} \|SA^{(j)} y - Sb^{(j)}\|, \forall j \in [m]$

Gaurantee : For all $\epsilon \in (0, 1/2)$, one can set $\boxed{t = O(k/\epsilon)}$ such that

$$\sum_{j=1}^{m} \|A^{(j)} y^{(j)} - b^{(j)}\|_2^2 \leqslant (1 + \epsilon) \sum_{j=1}^{m} \|A^{(j)} x^{(j)} - b^{(j)}\|_2^2$$

with prob. $9/10$

Warmup, inefficient WLRA Algorithm

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$
$A_{ij} \in \{0, \pm 1, \pm 2, \cdots, \pm \Delta\}$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

$W_{ij} \in \{0, 1, 2, \cdots, \Delta\}$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm :

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for U, $V^\top \in \mathbb{R}^{n \times k}$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for U, $V^\top \in \mathbb{R}^{n\times k}$
2. write polynomial $g(x_1, \cdots, x_{2nk}) = \|W \circ (A - UV)\|_F^2$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for $U, V^\top \in \mathbb{R}^{n \times k}$
2. write polynomial $g(x_1, \cdots, x_{2nk}) = \| W \circ (A - UV) \|_F^2$
3. pick $C \in [L^-, L^+]$, run polynomial verifier $g(x) \leqslant C$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon)\, \mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for U, $V^\top \in \mathbb{R}^{n \times k}$
2. write polynomial $g(x_1, \cdots, x_{2nk}) = \| W \circ (A - UV) \|_F^2$
3. pick $C \in [L^-, L^+]$, run polynomial verifier $g(x) \leqslant C$
4. optimize C by binary search over $[L^-, L^+]$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon)\, \mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for $U, V^\top \in \mathbb{R}^{n \times k}$
2. write polynomial $g(x_1, \cdots, x_{2nk}) = \| W \circ (A - UV) \|_F^2$
3. pick $C \in [L^-, L^+]$, run polynomial verifier $g(x) \leqslant C$
4. optimize C by binary search over $[L^-, L^+]$

Time : $2^{\Omega(nk)}$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for $U, V^\top \in \mathbb{R}^{n \times k}$
2. write polynomial $g(x_1, \cdots, x_{2nk}) = \| W \circ (A - UV) \|_F^2$
3. pick $C \in [L^-, L^+]$, run polynomial verifier $g(x) \leqslant C$
4. optimize C by binary search over $[L^-, L^+]$

Time : $2^{\Omega(nk)}$

How can we do better?

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for $U, V^\top \in \mathbb{R}^{n \times k}$

Time : $2^{\Omega(nk)}$

How can we do better?

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\| W \circ (A - \widehat{A}) \|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for $U, V^\top \in \mathbb{R}^{n \times k}$

Time : $2^{\Omega(nk)}$

How can we do better?

polynomial verifier runs in $(\#\,\text{constraints} \cdot \text{degree})^{O(\#\,\text{variables})}$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$\boxed{\text{1. create } 2nk \text{ variables for } U, V^\top \in \mathbb{R}^{n\times k}}$

Time : $2^{\Omega(nk)}$

How can we do better?

polynomial verifier runs in $(\#\,\text{constraints} \cdot \text{degree})^{O(\#\,\text{variables})}$

lower bound on cost $(\#\,\text{constraints})^{-\text{degree}^{O(\#\,\text{variables})}}$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for $U, V^\top \in \mathbb{R}^{n\times k}$

Time : $2^{\Omega(nk)}$

How can we do better?

polynomial verifier runs in $(\#\,\text{constraints} \cdot \text{degree})^{O(\#\,\text{variables})}$

lower bound on cost $(\#\,\text{constraints})^{-\text{degree}^{O(\#\,\text{variables})}}$

write a polynomial with few #variables, i.e. $\mathrm{poly}(kr/\epsilon)$

Warmup, inefficient WLRA Algorithm

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : rank-k $\widehat{A}$ s.t. $\|W \circ (A - \widehat{A})\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

1. create $2nk$ variables for $U, V^\top \in \mathbb{R}^{n\times k}$

Time : $2^{\Omega(nk)}$

How can we do better?

polynomial verifier runs in $(\#\,\mathrm{constraints}\cdot\mathrm{degree})^{O(\#\,\mathrm{variables})}$

lower bound on cost $(\#\,\mathrm{constraints})^{-\mathrm{degree}^{O(\#\,\mathrm{variables})}}$

write a polynomial with few #variables, i.e. $\mathrm{poly}(kr/\epsilon)$

without blowing up degree and #constraints too much

Main Idea

To reduce the number of variables to $\mathrm{poly}(kr/\epsilon)$:

1. Multiple regression sketch with $O(k/\epsilon)$ rows

2. Weight matrix W has rank at most r

Guess a Sketch

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$
$A_{ij} \in \{0, \pm 1, \pm 2, \cdots, \pm\Delta\}$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$
$W_{ij} \in \{0, 1, 2, \cdots, \Delta\}$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\| W \circ (UV - A) \|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

D_{W_j} be diagonal matrix with vector W_j

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

D_{W_j} be diagonal matrix with vector W_j

$$W = \begin{bmatrix} \boxed{\begin{matrix}1\\0\\1\\0\\1\\0\end{matrix}} & \begin{matrix}1\\1\\1\\0\\0\\0\end{matrix} & \begin{matrix}1\\2\\3\\4\\5\\6\end{matrix} & \begin{matrix}2\\1\\2\\0\\1\\0\end{matrix} & \begin{matrix}2\\2\\4\\4\\6\\6\end{matrix} & \begin{matrix}2\\3\\4\\4\\5\\6\end{matrix} \end{bmatrix} \quad D_{W_1} = \begin{bmatrix} 1 & & & & & \\ & 0 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 1 & \\ & & & & & 0 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\| W \circ (UV - A) \|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

D_{W_j} be diagonal matrix with vector W_j

$$W = \begin{bmatrix} 1 & \boxed{1} & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix} \quad D_{W_2} = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 0 & \\ & & & & & 0 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\| W \circ (UV - A) \|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W
D_{W_j} be diagonal matrix with vector W_j

$$W = \begin{bmatrix} 1 & 1 & \boxed{1} & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix} \quad D_{W_3} = \begin{bmatrix} 1 & & & & & \\ & 2 & & & & \\ & & 3 & & & \\ & & & 4 & & \\ & & & & 5 & \\ & & & & & 6 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\| W \circ (UV - A) \|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W
D_{W_j} be diagonal matrix with vector W_j

$$W = \begin{bmatrix} 1 & 1 & 1 & \boxed{2} & 2 & 2 \\ 0 & 1 & 2 & \boxed{1} & 2 & 3 \\ 1 & 1 & 3 & \boxed{2} & 4 & 4 \\ 0 & 0 & 4 & \boxed{0} & 4 & 4 \\ 1 & 0 & 5 & \boxed{1} & 6 & 5 \\ 0 & 0 & 6 & \boxed{0} & 6 & 6 \end{bmatrix} \quad D_{W_4} = \begin{bmatrix} 2 & & & & & \\ & 1 & & & & \\ & & 2 & & & \\ & & & 0 & & \\ & & & & 1 & \\ & & & & & 0 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

D_{W_j} be diagonal matrix with vector W_j

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & \boxed{2} & 2 \\ 0 & 1 & 2 & 1 & \boxed{2} & 3 \\ 1 & 1 & 3 & 2 & \boxed{4} & 4 \\ 0 & 0 & 4 & 0 & \boxed{4} & 4 \\ 1 & 0 & 5 & 1 & \boxed{6} & 5 \\ 0 & 0 & 6 & 0 & \boxed{6} & 6 \end{bmatrix} \quad D_{W_5} = \begin{bmatrix} 2 & & & & & \\ & 2 & & & & \\ & & 4 & & & \\ & & & 4 & & \\ & & & & 6 & \\ & & & & & 6 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\| W \circ (UV - A) \|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W
D_{W_j} be diagonal matrix with vector W_j

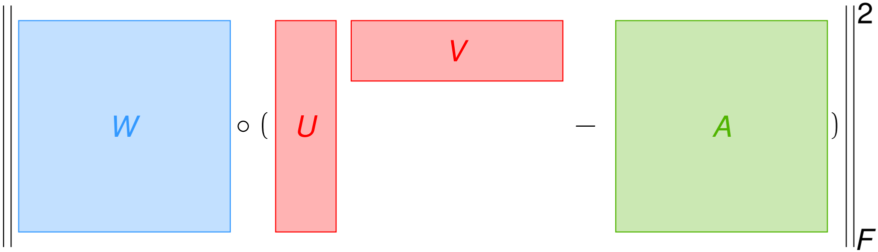
$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix} \quad D_{W_6} = \begin{bmatrix} 2 & & & & & \\ & 3 & & & & \\ & & 4 & & & \\ & & & 4 & & \\ & & & & 5 & \\ & & & & & 6 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

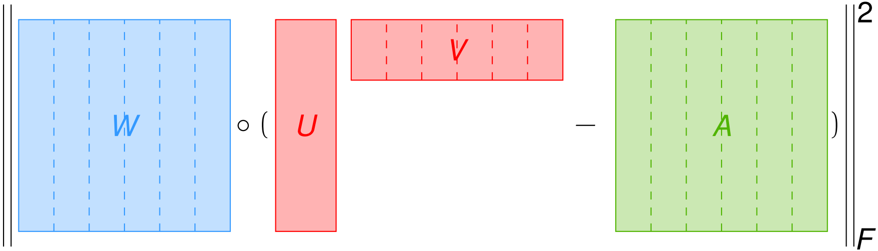


Guess a Sketch

Given : $\quad A \in \mathbb{R}^{n \times n}, W \in \mathbb{R}^{n \times n}, r \in \mathbb{N}, k \in \mathbb{N}, \epsilon > 0$

Output : $\quad U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

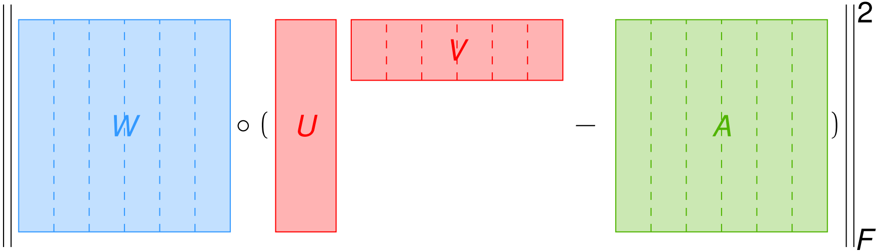


Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^{\top} \in \mathbb{R}^{n \times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2$

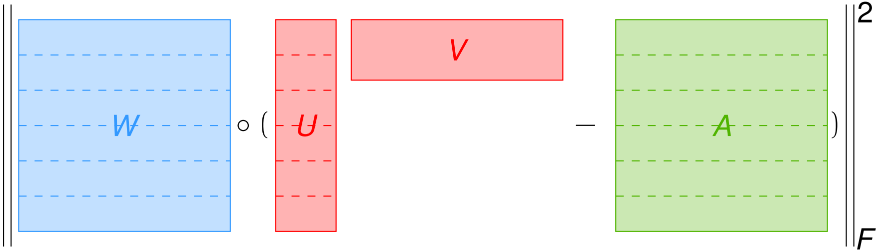


Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2$

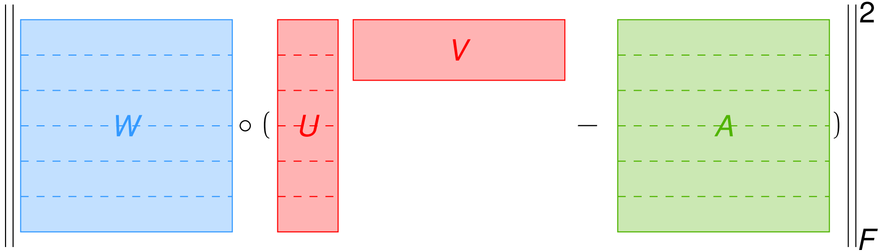


Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\| W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2 \longleftrightarrow \sum_{i=1}^{n} \| U^i V D_{W^i} - A^i D_{W^i} \|_2^2$



Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t \times n}$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

$\downarrow$ Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $S D_{W_j} U \in \mathbb{R}^{t\times k}$ and $V D_{W^i} T \in \mathbb{R}^{k\times t}$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $S D_{W_j} U \in \mathbb{R}^{t\times k}$ and $V D_{W^i} T \in \mathbb{R}^{k\times t}$

$$S D_{W_1} U = \boxed{S} \begin{bmatrix} 1 & & & & & \\ & 0 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 1 & \\ & & & & & 0 \end{bmatrix} \boxed{U}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $SD_{W_j}U \in \mathbb{R}^{t\times k}$ and $VD_{W^i}T \in \mathbb{R}^{k\times t}$

$$SD_{W_2}U = \boxed{S} \begin{bmatrix} 1 &&&&& \\ & 1 &&&& \\ && 1 &&& \\ &&& 0 && \\ &&&& 0 & \\ &&&&& 0 \end{bmatrix} \boxed{U}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\underbrace{\| W \circ (UV - A) \|_F^2} \leqslant (1 + \epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2 \longleftrightarrow \sum_{i=1}^{n} \| U^i V D_{W^i} - A^i D_{W^i} \|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t \times n}$

$$\sum_{j=1}^{n} \| S D_{W_j} U V_j - S D_{W_j} A_j \|_2^2 \qquad \sum_{i=1}^{n} \| U^i V D_{W^i} T - A^i D_{W^i} T \|_2^2$$

Guess $S D_{W_j} U \in \mathbb{R}^{t \times k}$ and $V D_{W^i} T \in \mathbb{R}^{k \times t}$

$$S D_{W_3} U = \boxed{\;S\;} \begin{bmatrix} 1 & & & & & \\ & 2 & & & & \\ & & 3 & & & \\ & & & 4 & & \\ & & & & 5 & \\ & & & & & 6 \end{bmatrix} \boxed{U}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $S D_{W_j} U \in \mathbb{R}^{t\times k}$ and $V D_{W^i} T \in \mathbb{R}^{k\times t}$

$$S D_{W_4} U = \boxed{\quad S \quad} \begin{bmatrix} 2 &&&&& \\ & 1 &&&& \\ && 2 &&& \\ &&& 0 && \\ &&&& 1 & \\ &&&&& 0 \end{bmatrix} \boxed{U}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $S D_{W_j} U \in \mathbb{R}^{t\times k}$ and $V D_{W^i} T \in \mathbb{R}^{k\times t}$

$$S D_{W_5} U = \boxed{S} \begin{bmatrix} 2 & & & & & \\ & 2 & & & & \\ & & 4 & & & \\ & & & 4 & & \\ & & & & 6 & \\ & & & & & 6 \end{bmatrix} \boxed{U}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$ $\qquad$ $\sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$

Guess $S D_{W_j} U \in \mathbb{R}^{t\times k}$ and $V D_{W^i} T \in \mathbb{R}^{k\times t}$

$$S D_{W_6} U = \boxed{S} \begin{bmatrix} 2 & & & & & \\ & 3 & & & & \\ & & 4 & & & \\ & & & 4 & & \\ & & & & 5 & \\ & & & & & 6 \end{bmatrix} \boxed{U}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $S D_{W_j} U \in \mathbb{R}^{t\times k}$ and $V D_{W^i} T \in \mathbb{R}^{k\times t}$

Create $t \times k$ variables for each of n $S D_{W_j} U$ s?

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $S D_{W_j} U \in \mathbb{R}^{t\times k}$ and $V D_{W^i} T \in \mathbb{R}^{k\times t}$

Create $t \times k$ variables for each of n $S D_{W_j} U$ s?

$$n \times t \times k$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} UV_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|SD_{W_j} UV_j - SD_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $SD_{W_j} U \in \mathbb{R}^{t\times k}$ and $VD_{W^i} T \in \mathbb{R}^{k\times t}$

Create $t \times k$ variables for each of n $SD_{W_j}U$ s?

$$? \times t \times k$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : $\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \longleftrightarrow \sum_{i=1}^{n} \|U^i V D_{W^i} - A^i D_{W^i}\|_2^2$

Sketch by Gaussian $S, T^\top \in \mathbb{R}^{t\times n}$

$$\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2 \qquad \sum_{i=1}^{n} \|U^i V D_{W^i} T - A^i D_{W^i} T\|_2^2$$

Guess $SD_{W_j}U \in \mathbb{R}^{t\times k}$ and $VD_{W^i}T \in \mathbb{R}^{k\times t}$

Create $t \times k$ variables for each of n $SD_{W_j}U$ s?

$$r \times t \times k$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\| W \circ (UV - A) \|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : U, $V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix} \qquad \overbrace{\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \\ 1 & 1 & 3 \\ 0 & 0 & 4 \\ 1 & 0 & 5 \\ 0 & 0 & 6 \end{bmatrix}}^{\text{column span of } W}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

$$W_1 = W_1$$

$$W = \begin{bmatrix} \boxed{\begin{matrix}1\\0\\1\\0\\1\\0\end{matrix}} & \begin{matrix}1\\1\\1\\0\\0\\0\end{matrix} & \begin{matrix}1\\2\\3\\4\\5\\6\end{matrix} & \begin{matrix}2\\1\\2\\0\\1\\0\end{matrix} & \begin{matrix}2\\2\\4\\4\\6\\6\end{matrix} & \begin{matrix}2\\3\\4\\4\\5\\6\end{matrix} \end{bmatrix} \qquad \begin{bmatrix}1\\0\\1\\0\\1\\0\end{bmatrix} = \begin{bmatrix}1\\0\\1\\0\\1\\0\end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

$$W_2 = W_2$$

$$W = \begin{bmatrix} 1 & \boxed{1} & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix} \qquad \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

$$W_3 = W_3$$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix} \qquad \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : U, $V^\top \in \mathbb{R}^{n\times k}$ s.t. $\|W \circ (UV - A)\|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

$$W_4 = W_1 + W_2$$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & 2 \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix} \qquad \begin{bmatrix} 2 \\ 1 \\ 2 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\| W \circ (UV - A) \|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

$$W_5 = W_1 + W_3$$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & \boxed{2} & 2 \\ 0 & 1 & 2 & 1 & \boxed{2} & 3 \\ 1 & 1 & 3 & 2 & \boxed{4} & 4 \\ 0 & 0 & 4 & 0 & \boxed{4} & 4 \\ 1 & 0 & 5 & 1 & \boxed{6} & 5 \\ 0 & 0 & 6 & 0 & \boxed{6} & 6 \end{bmatrix} \qquad \begin{bmatrix} 2 \\ 2 \\ 4 \\ 4 \\ 6 \\ 6 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\| W \circ (UV - A) \|_F^2 \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm : W_j be jth column of W

$$W_6 = W_2 + W_3$$

$$W = \begin{bmatrix} 1 & 1 & 1 & 2 & 2 & \boxed{2} \\ 0 & 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 2 & 4 & 4 \\ 0 & 0 & 4 & 0 & 4 & 4 \\ 1 & 0 & 5 & 1 & 6 & 5 \\ 0 & 0 & 6 & 0 & 6 & 6 \end{bmatrix} \qquad \begin{bmatrix} 2 \\ 3 \\ 4 \\ 4 \\ 5 \\ 6 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{bmatrix}$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{\sum_{j=1}^{n} \|D_{W_j} UV_j - D_{W_j} A_j\|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \|D_{W_j} UV_j - D_{W_j} A_j\|_2^2$$

$$\text{sketch} \sum_{j=1}^{n} \|SD_{W_j} UV_j - SD_{W_j} A_j\|_2^2$$

$1+\epsilon$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{\sum_{j=1}^{n} \|D_{W_j} UV_j - D_{W_j} A_j\|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \|D_{W_j} UV_j - D_{W_j} A_j\|_2^2$$

$$1+\epsilon$$

$$\text{sketch } \sum_{j=1}^{n} \|SD_{W_j} UV_j - SD_{W_j} A_j\|_2^2$$

create variables for $SD_{W_j}U, \forall j \in [r]$

Guess a Sketch

Given : $\quad A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $\quad U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\underbrace{\| W \circ (UV - A) \|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2$$

$$1+\epsilon$$

$$\text{sketch} \sum_{j=1}^{n} \| S D_{W_j} U V_j - S D_{W_j} A_j \|_2^2$$

create variables for $S D_{W_j} U, \forall j \in [r]$

create $t \times k$ variables for $S D_{W_1} U$

$$S \qquad \begin{bmatrix} 1 & & & & & \\ & 0 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 1 & \\ & & & & & 0 \end{bmatrix} \qquad U$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \quad 1+\epsilon$$

sketch $\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

create $t \times k$ variables for $S D_{W_2} U$

S

$$\begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 0 & \\ & & & & & 0 \end{bmatrix}$$

U

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2 \quad \xrightarrow{1+\epsilon}$$

sketch $\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

create $t \times k$ variables for $S D_{W_3} U$

S

$$\begin{bmatrix} 1 & & & & & \\ & 2 & & & & \\ & & 3 & & & \\ & & & 4 & & \\ & & & & 5 & \\ & & & & & 6 \end{bmatrix}$$

U

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\underbrace{\| W \circ (UV - A) \|_F^2}_{\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2 \qquad 1+\epsilon$$

sketch $\sum_{j=1}^{n} \| S D_{W_j} U V_j - S D_{W_j} A_j \|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_4} U$ as $S D_{W_1} U + S D_{W_2} U$

$$S \qquad \begin{bmatrix} 2 & & & & & \\ & 1 & & & & \\ & & 2 & & & \\ & & & 0 & & \\ & & & & 1 & \\ & & & & & 0 \end{bmatrix} \qquad U$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2$$

$1+\epsilon$

sketch $\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_4} U$ as $S D_{W_1} U + S D_{W_2} U$

$$S \begin{bmatrix} 1+1 & & & & & \\ & 0+1 & & & & \\ & & 1+1 & & & \\ & & & 0+0 & & \\ & & & & 1+0 & \\ & & & & & 0+0 \end{bmatrix} U$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\| W \circ (UV - A) \|_F^2}_{\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2$$

$1 + \epsilon$

sketch $\sum_{j=1}^{n} \| S D_{W_j} U V_j - S D_{W_j} A_j \|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_4} U$ as $S D_{W_1} U + S D_{W_2} U$

$$S \left(\begin{bmatrix} 1 & & & & & \\ & 0 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 1 & \\ & & & & & 0 \end{bmatrix} + \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 0 & \\ & & & & & 0 \end{bmatrix} \right) U$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2$$

$1+\epsilon$

sketch $\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_5} U$ as $S D_{W_1} U + S D_{W_3} U$

S

$$\begin{bmatrix} 2 & & & & & \\ & 2 & & & & \\ & & 4 & & & \\ & & & 4 & & \\ & & & & 6 & \\ & & & & & 6 \end{bmatrix}$$

U

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2$$

$$1+\epsilon$$

sketch $\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_5} U$ as $S D_{W_1} U + S D_{W_3} U$

$$S \begin{bmatrix} 1+1 & & & & & \\ & 0+2 & & & & \\ & & 1+3 & & & \\ & & & 0+4 & & \\ & & & & 1+5 & \\ & & & & & 0+6 \end{bmatrix} U$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{\sum_{j=1}^n \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^n \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2$$

$1+\epsilon$

sketch $\sum_{j=1}^n \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_5} U$ as $S D_{W_1} U + S D_{W_3} U$

$$S \left(\begin{bmatrix} 1 & & & & & \\ & 0 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 1 & \\ & & & & & 0 \end{bmatrix} + \begin{bmatrix} 1 & & & & & \\ & 2 & & & & \\ & & 3 & & & \\ & & & 4 & & \\ & & & & 5 & \\ & & & & & 6 \end{bmatrix} \right) U$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\| W \circ (UV - A) \|_F^2}$ $\leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2$$

$1+\epsilon$

sketch $\sum_{j=1}^{n} \| S D_{W_j} U V_j - S D_{W_j} A_j \|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_6} U$ as $S D_{W_2} U + S D_{W_3} U$

S

$$\begin{bmatrix} 2 & & & & & \\ & 3 & & & & \\ & & 4 & & & \\ & & & 4 & & \\ & & & & 5 & \\ & & & & & 6 \end{bmatrix}$$

U

Guess a Sketch

Given : $A \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{n\times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n\times k}$ s.t. $\underbrace{\|W \circ (UV - A)\|_F^2}_{\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \|D_{W_j} U V_j - D_{W_j} A_j\|_2^2$$

$1+\epsilon$

sketch $\sum_{j=1}^{n} \|S D_{W_j} U V_j - S D_{W_j} A_j\|_2^2$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_6} U$ as $S D_{W_2} U + S D_{W_3} U$

$$S \begin{bmatrix} 1+1 & & & & & \\ & 1+2 & & & & \\ & & 1+3 & & & \\ & & & 0+4 & & \\ & & & & 0+5 & \\ & & & & & 0+6 \end{bmatrix} U$$

Guess a Sketch

Given : $A \in \mathbb{R}^{n \times n}$, $W \in \mathbb{R}^{n \times n}$, $r \in \mathbb{N}$, $k \in \mathbb{N}$, $\epsilon > 0$

Output : $U, V^\top \in \mathbb{R}^{n \times k}$ s.t. $\underbrace{\| W \circ (UV - A) \|_F^2}_{\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2} \leqslant (1+\epsilon)\,\mathrm{OPT}$

Algorithm :

$$\sum_{j=1}^{n} \| D_{W_j} U V_j - D_{W_j} A_j \|_2^2 \;\xrightarrow{1+\epsilon}\; \text{sketch } \sum_{j=1}^{n} \| S D_{W_j} U V_j - S D_{W_j} A_j \|_2^2$$

create variables for $S D_{W_j} U, \forall j \in [r]$

write $S D_{W_6} U$ as $S D_{W_2} U + S D_{W_3} U$

$$S \left(\begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 0 & & \\ & & & & 0 & \\ & & & & & 0 \end{bmatrix} + \begin{bmatrix} 1 & & & & & \\ & 2 & & & & \\ & & 3 & & & \\ & & & 4 & & \\ & & & & 5 & \\ & & & & & 6 \end{bmatrix} \right) U$$

Wrapping Up

- Can use an explicit formula for the regression solution in terms of the variables created

Wrapping Up

- Can use an explicit formula for the regression solution in terms of the variables created
- The regression solution is a rational function, but can use tricks to clear the denominator

Wrapping Up

- Can use an explicit formula for the regression solution in terms of the variables created
- The regression solution is a rational function, but can use tricks to clear the denominator
- Multiple Regression Sketch + Bounded Rank Weight Matrix imply a small number of variables

Wrapping Up

- Can use an explicit formula for the regression solution in terms of the variables created
- The regression solution is a rational function, but can use tricks to clear the denominator
- Multiple Regression Sketch + Bounded Rank Weight Matrix imply a small number of variables
- Time : $n^{O(rk^2/\epsilon)}$

Wrapping Up

- Can use an explicit formula for the regression solution in terms of the variables created
- The regression solution is a rational function, but can use tricks to clear the denominator
- Multiple Regression Sketch + Bounded Rank Weight Matrix imply a small number of variables
- Time : $n^{O(rk^2/\epsilon)}$

$$(\#\,\text{constraints} \cdot \text{degree})^{O(\#\,\text{variables})} = (O(n) \cdot O(nk))^{O(krt)}$$

Open Problems

- For a rank-r weight matrix W, the upper bound is $n^{O(k^2 r/\epsilon)}$ but the lower bound is only $2^{\Omega(r)}$ - can we close this gap?

Open Problems

- For a rank-r weight matrix W, the upper bound is $n^{O(k^2 r/\epsilon)}$ but the lower bound is only $2^{\Omega(r)}$ - can we close this gap?
- Can we prove a hardness result with respect to the parameter k, e.g., a $2^{\Omega(k)}$ lower bound for WLRA problem?

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity
  - nonnegative matrix factorization

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity
  - nonnegative matrix factorization
  - weighted low rank approximation

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity
  - nonnegative matrix factorization
  - weighted low rank approximation
- Algorithm techniques

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity
  - nonnegative matrix factorization
  - weighted low rank approximation
- Algorithm techniques
  - polynomial system verifier

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity
  - nonnegative matrix factorization
  - weighted low rank approximation
- Algorithm techniques
  - polynomial system verifier
  - random matrices for dimensionality reduction to reduce the number of variables

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity
  - nonnegative matrix factorization
  - weighted low rank approximation
- Algorithm techniques
  - polynomial system verifier
  - random matrices for dimensionality reduction to reduce the number of variables
- Hardness techniques

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity
  - nonnegative matrix factorization
  - weighted low rank approximation
- Algorithm techniques
  - polynomial system verifier
  - random matrices for dimensionality reduction to reduce the number of variables
- Hardness techniques
  - random 4-SAT, ETH, etc.

Conclusions

- Studied intractable matrix factorization problems through the lens of parameterized complexity
  - nonnegative matrix factorization
  - weighted low rank approximation
- Algorithm techniques
  - polynomial system verifier
  - random matrices for dimensionality reduction to reduce the number of variables
- Hardness techniques
  - random 4-SAT, ETH, etc.
- Parameterized Complexity gives a way of coping with intractability for emerging machine learning problems

Topics at a Survey Level

- Projection onto Complicated Objects and Gaussian Mean Width
- M-Estimator Loss Functions for Regression
- Compressed Sensing

Projection onto other Objects

- Least squares regression finds the closest point y in a subspace K to a given point b
- Given a (possibly infinite) set of points K, and a point b, compute $\min_{y \in K} |y - b|$
  - All norms are Euclidean norms
- Let S be a sketching matrix, we want that if y' = $\mathrm{argmin}_{y \in K} |Sy - Sb|$, then

$$|y' - b| \leq (1 + \epsilon) \min_{y \in K} |y - b|$$

- More generally, want to preserve distances of all vectors in a set K, that is, |S(y-y')| = $(1 \pm \epsilon)|y - y'|$ for all y, y' ∈ K

What properties of K determine the dimension and sparsity of S?

Example: Preserving Distances in a Set

- More generally, want to preserve distances of all vectors in a set K, that is, |S(y-y')| = $(1 \pm \epsilon)|y - y'|$ for all y, y' ∈ K

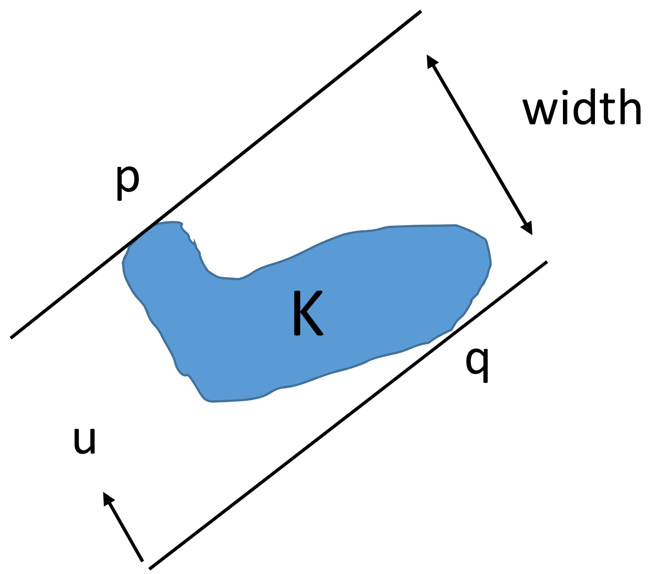
- What is the dimension of S needed if K is:

  - n arbitrary points in R^{d}?

  - n arbitrary points on a line in R^{d}?

Spherical Mean Width

- Let K be a bounded subset in R^n
- Consider the width in direction u for a unit vector u:



- Width in direction u = $\sup_{\mathrm{p,q\ in\ K}} < \mathrm{u}, \mathrm{p} - \mathrm{q} >$
- Spherical mean width = $\mathrm{E_u}[\sup_{\mathrm{p,q\ in\ K}} < \mathrm{u}, \mathrm{p} - \mathrm{q} >]$

Gaussian Mean Width

- Let $g \sim N(0, I_n)$ be an i.i.d. Gaussian vector

- Gaussian mean width g(K) = $E_g[\sup_{p,q \text{ in } K} < g, p - q >]$

  $= \Theta(n^{.5}) \cdot$ spherical mean width

- Examples

  - K = S^{n-1}
    - $\Theta(n^{.5})$
  - K = set of unit vectors in a d-dimensional subspace of R^n
    - $\Theta(d^{.5})$
  - K = t arbitrary unit vectors in R^n
    - $\Theta(\log^{.5} t)$

Gaussian Mean Width of t Arbitrary Unit Vectors

- Let $u^1, \ldots, u^t$ be t arbitrary unit vectors in R^n
- Let g in R^n have iid N(0,1) entries
- Define random variables $Z_j = < u^j, g >$ which are N(0,1) random variables
- Want to bound $E_g[\max_j Z_j]$
- Fact: for an N(0,1) random variable W, $E\left[e^{\lambda W}\right] = e^{\lambda^2/2}$
- For any $\lambda > 0$, $E\left[e^{\lambda \max_j Z_j}\right] \leq \sum_j E\left[e^{\lambda Z_j}\right] \leq t\, e^{\lambda^2/2}$
- For all $\lambda > 0$, $E_g[\max_j Z_j] \leq \left(\frac{1}{\lambda}\right) \log E[e^{\lambda \max_j Z_j}] \leq (\frac{\log t}{\lambda} + \frac{\lambda}{2}) = 2\sqrt{\log t}$

Sketching Bounds

- [Gordon] Let K be a subset of S^{n-1}. A random Gaussian matrix S with $g(K)^2/\epsilon^2$ rows satisfies
$$|S(y-y')|^2 = (1 \pm \epsilon)|y-y'|^2 \text{ for all y, y' in K}$$

- What about sparse sketching matrices S?

- [Bourgain, Dirksen, Nelson] S can have m = $g(K)^2 \text{poly}(\log n)/\epsilon^2$ rows and $s = \text{poly}(\log n)/\epsilon^2$ non- zeros per column if m and s satisfy a condition related to higher moments of $\sup_{p,q} < g, p-q >$
  - Applied to finite and infinite unions of subspaces

Topics at a Survey Level

- Projection onto Complicated Objects and Gaussian Mean Width
- M-Estimator Loss Functions for Regression
- Compressed Sensing

Other Fitness Measures

Example: Method of least absolute deviation (l_1 -regression)

- Find x* that minimizes $|Ax-b|_1 = \Sigma\ |b_i - \langle A_{i*}, x\rangle|$
- Cost is less sensitive to outliers than least squares
- Can solve via linear programming

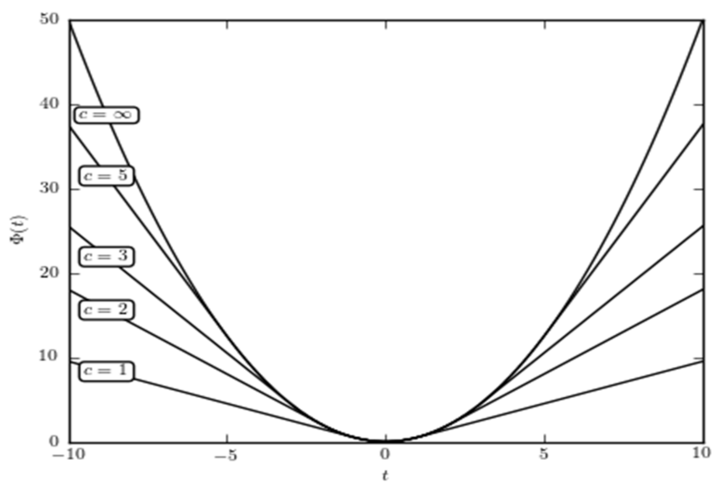
What about the many other fitness measures used in practice?

Huber Loss Function

$M(x) = x^2/(2c)$ for $|x| \leq c$

$M(x) = |x|-c/2$ for $|x| > c$

Enjoys smoothness properties of l_2^2 and robustness properties of l_1



Other Examples

- L_1-L_2

  $M(x) = 2((1+x^2/2)^{1/2} - 1)$

- Fair estimator

  $M(x) = c^2 [\ |x|/c - \log(1+|x|/c)\ ]$

- Tukey estimator

  $M(x) = c^2/6\ (1-[1-(x/c)^2]^3) \quad \text{if } |x| \leq c$

  $\quad = c^2/6 \quad \text{if } |x| > c$

Nice M-Estimators

- An M-Estimator is nice if it has at least linear growth and at most quadratic growth

- There is $C_M > 0$ so that for all a, a’ with $|a| \geq |a'| > 0$,

  $$|a/a'|^2 \geq M(a)/M(a') \geq C_M\, |a/a'|$$

- Any convex M satisfies the linear lower bound

  $$M(a') = M\left(\frac{a'}{a} \cdot a + \left(1 - \frac{a'}{a}\right) \cdot 0\right) \leq \frac{a'}{a} M(a) + \left(1 - \frac{a'}{a}\right) M(0) = \frac{a'}{a} M(a)$$

- Any sketchable M satisfies the quadratic upper bound
  - sketchable => there is a distribution on k x n matrices S for which $|Sx|_M = \Theta(|x|_M)$ with good probability and k is slow-growing function of n

Nice M-Estimator Theorem

[Nice M-Estimators] $O(nnz(A)) + poly(d \log n)$ time algorithm to output x'
so that for any constant $C > 1$, with probability 99%:

$$|Ax'-b|_M \leq C \min_x |Ax-b|_M$$

Remarks:

- For convex nice M-estimators can solve with convex programming, but slow – poly(nd) time

- The sketch is "universal"

M-Sketch

$$T = \begin{bmatrix} S^0 \cdot D^0 \\ S^1 \cdot D^1 \\ S^2 \cdot D^2 \\ \cdots \\ S^{\log n} \cdot D^{\log n} \end{bmatrix}$$

-The same M-Sketch works for all nice M-estimators!

$x' = \text{argmin}_x \ |TAx\text{-}Tb|_{w,M}$

- S^i are independent CountSketch matrices with poly(d) rows
- D^i is n x n diagonal and uniformly samples a $1/(d \log n)^i$ fraction of the n rows
- Want $|T(Ax - b)|_{w,M} \approx |Ax - b|_M$ for any given y = Ax-b
  – Example: y = (n, 1, …, 1): $|y|_1 = n + 1 \cdot (n - 1)$. Estimate both scales

Topics at a Survey Level

- Projection onto Complicated Objects and Gaussian Mean Width
- M-Estimator Loss Functions for Regression
- Compressed Sensing

Compressed Sensing

- We take random “linear measurements” of an n-dimensional vector x
- In our language, we choose a random r x n sketching matrix S and observe $\mathrm{S} \cdot \mathrm{x}$
- Output a vector x’ with $|\mathrm{x}-\mathrm{x}'|_{\mathrm{p}} = \mathrm{D} \cdot \min_{\mathrm{k-sparse\ z}} |\mathrm{x}-\mathrm{z}|_{\mathrm{q}}$, where D is the distortion (the $\ell_{\mathrm{p}}/\ell_{\mathrm{q}}$-guarantee)
- Let x_{k} be the best k-sparse approximation to x, i.e., the largest k coordinates in magnitude
- Randomized (“for-each”) scheme versus deterministic (“for-all”) scheme
- CountSketch is a randomized scheme achieving ℓ_2/ℓ_2 w.h.p.

$$|\mathrm{x}-\mathrm{x}'|_2 = \mathrm{O}(1)\cdot |\mathrm{x}-\mathrm{x}_{\mathrm{k}}|_2$$

CountSketch for Compressed Sensing

- CountSketch had O(log n) repetitions of hashing into O(k) buckets
- S is a random linear map S with O(k log n) rows
- For an n-dimensional vector x, estimate every x_i up to additive error $\frac{|x-x_k|_2}{\sqrt{k}}$
- Output a 2k-sparse x' consisting of the top 2k estimates given by CountSketch
- Say coordinate i is heavy if $|x_i| \geq |x - x_k|_2/\sqrt{k}$
  - How many heavy coordinates can there be?
- Say a coordinate i is super-heavy if $|x_i| \geq 3|x - x_k|_2/\sqrt{k}$
  - Claim: the set T of super-heavy coordinates is in the support of x'
- $|x - x'|_2 \leq |(x - x')_T|_2 + \left|(x - x')_{[n]\setminus T}\right|_2$

$$\leq \sqrt{2k} \cdot \frac{|x-x_k|_2}{\sqrt{k}} + |(x - x_k)_{[n]\setminus T}|_2 + \left|(x_k - x')_{[n]\setminus T}\right|_2 = O(|x - x_k|_2)$$

No Deterministic Algorithm Achieves ℓ_2/ℓ_2

- Recall ℓ_2/ℓ_2: output x' with $|x - x'|_2 = O(1) \cdot |x - x_k|_2$
- Consider k = 1
- Suppose S is a deterministic sketching matrix with r = o(n) rows
- Suffices to show there is a vector x in kernel(S) with $|x|_\infty \geq C|x - x_1|_2$ for any constant C > 0
- W.l.o.g., can assume S has orthonormal rows
- $\sum_i |Se_i|_2^2 = r$, so there exists an i with $|Se_i|_2^2 \leq \frac{r}{n}$
- Let $x = e_i - S^T S e_i$, so x is in kernel(S)
- But $|x|_\infty^2 \geq |x_i|^2 = \left(e_i^T e_i - e_i^T S^T S e_i\right)^2 \geq \left(1 - \frac{r}{n}\right)^2$, while
- $|x - x_1|_2 \leq |x - e_i|_2 = \left|S^T S e_i\right|_2 = |Se_i|_2 \leq \sqrt{\frac{r}{n}} = o(1)$

Deterministic Algorithms Achieve ℓ_2/ℓ_1

- ℓ_2/ℓ_1: output x' with $|x - x'|_2 = O(1/k^{.5}) \cdot |x - x_k|_1$
- S has the (ϵ, k)-restricted isometry property (RIP) if for all k-sparse vectors x,

$$(1-\epsilon)|x|_2^2 \le |Sx|_2^2 \le (1+\epsilon)|x|_2^2$$

- What are some matrices S with O(k log(n/k)) rows that have the (ϵ, k)-RIP property for constant ϵ?
- Deterministic, but not explicit!
- Major open question: explicit matrix with (ϵ, k)-RIP with $o(k^2)$ rows
- Bourgain et al.: can get $k^{2-\gamma}$ rows for a constant $\gamma > 0$ and $k \approx n^{.5}$

Deterministic Algorithms Achieve ℓ_2/ℓ_1

- If S has the (ϵ, k)-RIP then one can efficiently output an x' for which
$$|\mathrm{x} - \mathrm{x}'|_2 = \mathrm{O}(1/\mathrm{k}^{.5}) \cdot |\mathrm{x} - \mathrm{x}_\mathrm{k}|_1$$

- In fact, can just solve a linear program!

$$\min_{\mathrm{z} \in \mathrm{R}^\mathrm{n}} |\mathrm{z}|_1$$
s.t. Sz = Sx

- If x' is the solution, then $|\mathrm{x} - \mathrm{x}'|_2 \leq \mathrm{O}\left(\frac{1}{\mathrm{k}^{.5}}\right) |\mathrm{x} - \mathrm{x}_\mathrm{k}|_1$

- Proof uses (ϵ, k)-RIP and elementary norm manipulations