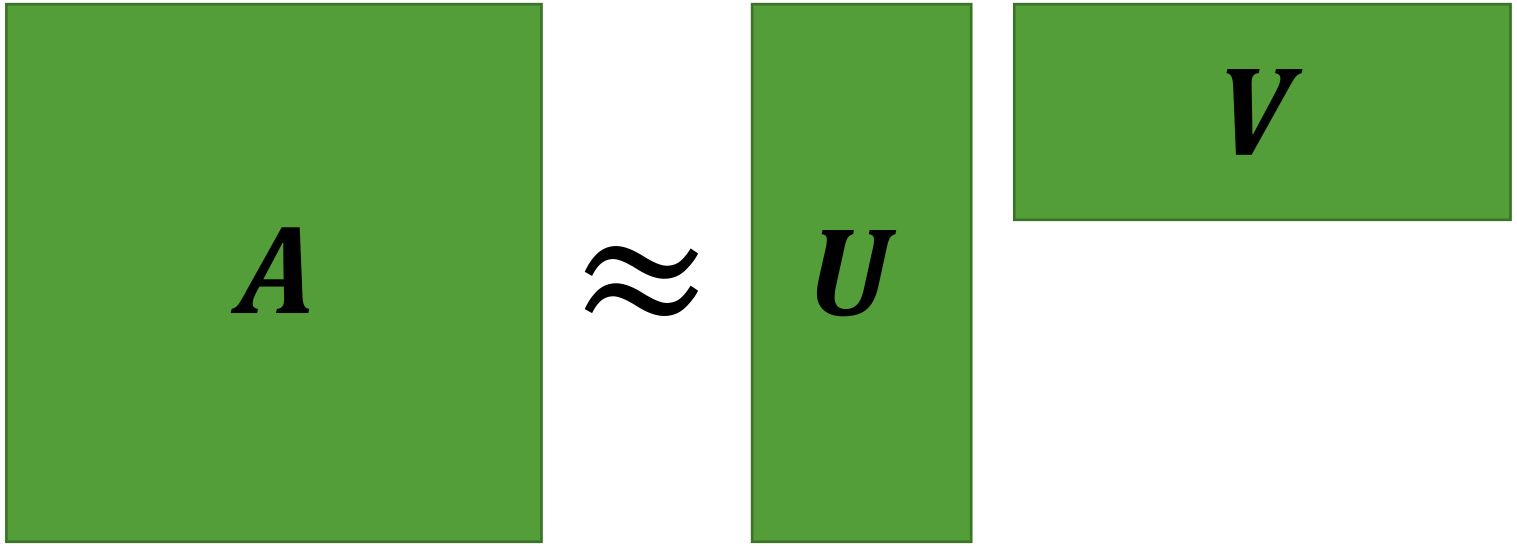
Sample-Optimal Low-Rank Approximation of Distance Matrices

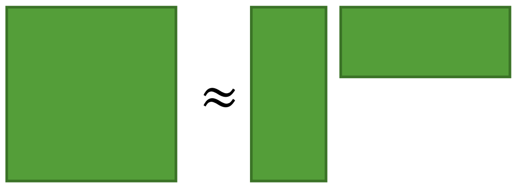
Piotr Indyk	Ali Vakilian	Tal Wagner	David Woodruff
MIT	MIT	MIT	CMU

Matrix Low-Rank Approximation

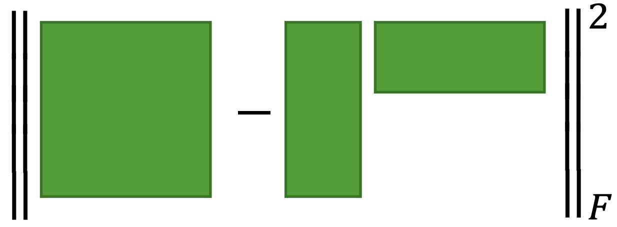
- Input: $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, integer $0 < k \ll n$
- Output: $\boldsymbol{U}, \boldsymbol{V}^T \in \mathbb{R}^{n \times k}$ such that $\boldsymbol{A} \approx \boldsymbol{U}\boldsymbol{V}$
- **Why?** Matrix operations are space and time intensive



Matrix Low-Rank Approximation



- Baseline: **SVD**
  - Returns: $\boldsymbol{A}_k$ s.t. $||\boldsymbol{A}-\boldsymbol{A}_k||_F^2 = \min_{\boldsymbol{U},\boldsymbol{V}} ||\boldsymbol{A}-\boldsymbol{U}\boldsymbol{V}||_F^2$
  - Runtime: $O(n^3)$, **too slow**



- Faster algorithms?

  Find $\boldsymbol{U},\boldsymbol{V}$ s.t. $||\boldsymbol{A}-\boldsymbol{U}\boldsymbol{V}||_F^2 \le ||\boldsymbol{A}-\boldsymbol{A}_k||_F^2 + err$

  Runtime: **nearly linear**, $\tilde{O}(n^2)$ or $\tilde{O}(nnz(\boldsymbol{A}))$

  [Frieze-Kannan-Vempala’04] [Drineas-Kannan-Mahoney’06] [Sarlos’06] [Clarkson-Woodruff’09, ‘13] [**many more**]

Can we do better?

Sublinear-Time Algorithms?

- Arbitrary matrices: **No**

- Some families of matrices: **Yes**
  - Incoherent matrices [Candes-Recht'09]
  - PSD matrices [Musco-Woodruff'17]
  - **Distance matrices** [Bakshi-Woodruff'18], **this work**

Distance Matrices

- Let $(\mathcal{X}, \mathbf{d})$ be a metric space:

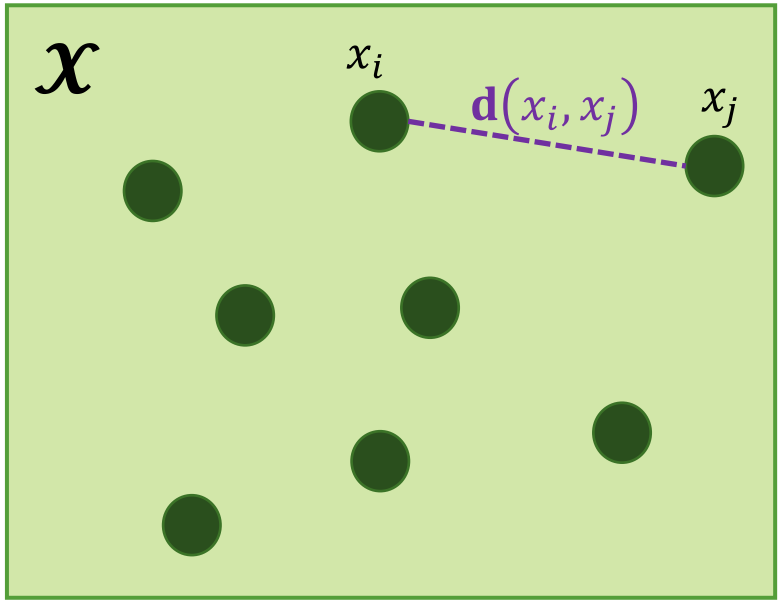
  $\mathcal{X} = \{x_1, \dots, x_n\}$

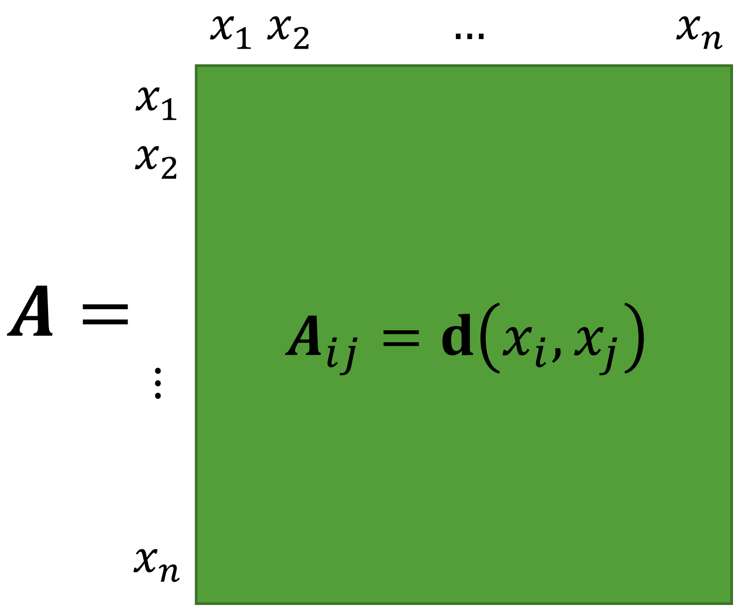
  $\mathbf{d}: \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$, symmetric, triangle inequality

- Distance matrix: $A_{ij} = \mathbf{d}(x_i, x_j)$

- Many applications
  - E.g. survey: [Dokmanic-Parhizkar-Ranieri-Vetterli’15]

- **<u>This work</u>: Low-rank approximation of distance matrices**





Our Results: Upper Bound

Nearly linear time **in $\boldsymbol{n}$**:

Algorithm: Given distance matrix $\boldsymbol{A} \in \mathbb{R}^{n \times n}$, $k \in \mathbb{N}$, $\epsilon > 0$,

- Runtime: $\tilde{O}(n) \cdot poly(k, \epsilon^{-1})$
- Returns: $\boldsymbol{U}, \boldsymbol{V}^{\boldsymbol{T}} \in \mathbb{R}^{n \times k}$ s.t. $||\boldsymbol{A} - \boldsymbol{U}\boldsymbol{V}||_F^2 \leq \underbrace{||\boldsymbol{A} - \boldsymbol{A}_k||_F^2}_{\text{Optimal (SVD) error}} + \underbrace{\epsilon||\boldsymbol{A}||_F^2}_{\text{Additional error}}$
- Simple

Previous work: $\tilde{O}(n^{1+\gamma}) \cdot poly(k, \epsilon^{-1})$ for $\gamma > 0$ [Bakshi-Woodruff'18]

Our Results: Lower Bound

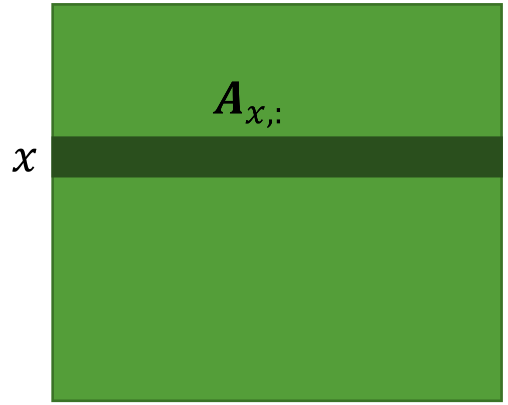
Tight query complexity:

- Our algorithm reads $O(nk\epsilon^{-1})$ entries of $\boldsymbol{A}$
- **Theorem:** Any algorithm must read $\Omega(nk\epsilon^{-1})$ entries of $\boldsymbol{A}$

Algorithm

- **Theorem** [Frieze-Kannan-Vempala’04]: For any matrix,

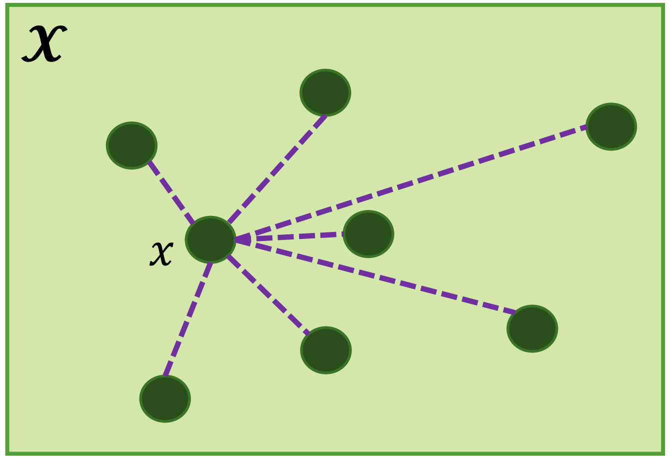
Sampling rows proportionally to their ℓ_2-norm $\Rightarrow$ Low-rank approximation



- **New problem:**

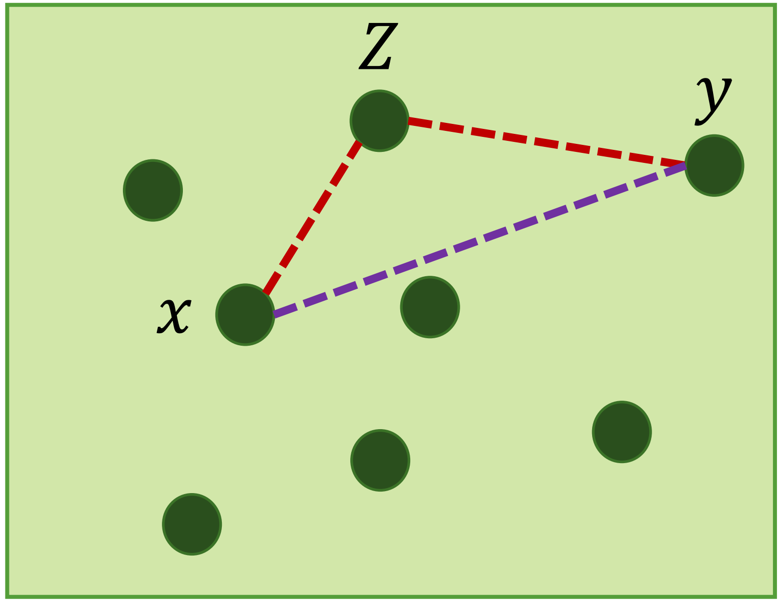
For all $x \in \mathcal{X}$, estimate $\|A_{x,:}\|_2^2 = \sum_y \mathbf{d}(x,y)^2$

[Bakshi-Woodruff’18], [Indyk’99], [Chen’06], **this work**



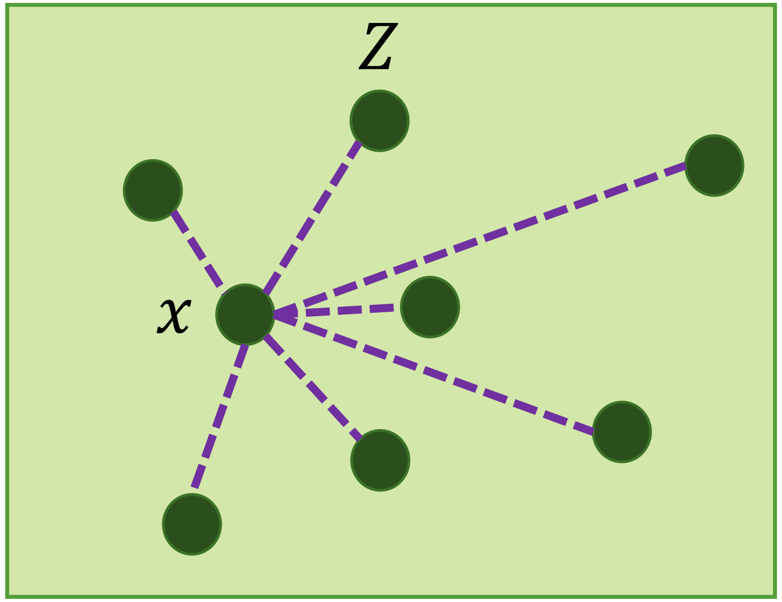
Estimating Sums of Squared Distances

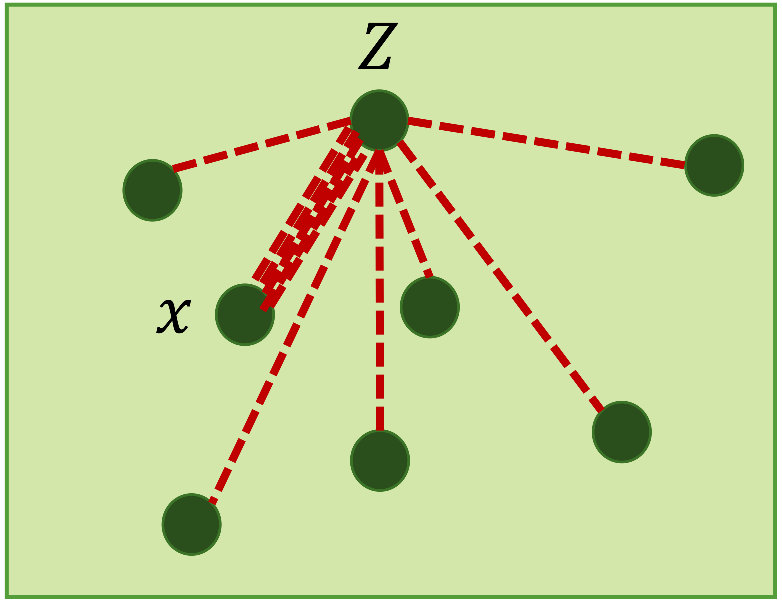
- Pick $Z \in \boldsymbol{\mathcal{X}}$ u.a.r.
- For all x, y, estimate $\mathbf{d}(x, y)^2$ by $\mathbf{d}(x, Z)^2 + \mathbf{d}(Z, y)^2$



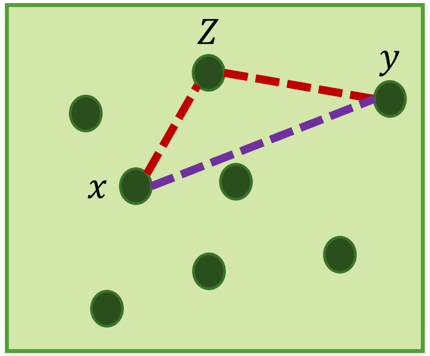
Estimating Sums of Squared Distances

- Pick $Z \in \mathcal{X}$ u.a.r.
- For all x, y, estimate $\mathbf{d}(x,y)^2$ by $\mathbf{d}(x,Z)^2 + \mathbf{d}(Z,y)^2$
- For all x, estimate $\sum_y \mathbf{d}(x,y)^2$ by $n \cdot \mathbf{d}(x,Z)^2 + \sum_y \mathbf{d}(Z,y)^2$





Analysis



On one hand:

Triangle inequality

$$\forall x, \quad \sum_y \mathbf{d}(x,y)^2 \le 2\left(n \cdot \mathbf{d}(x,Z)^2 + \sum_y \mathbf{d}(Z,y)^2\right)$$

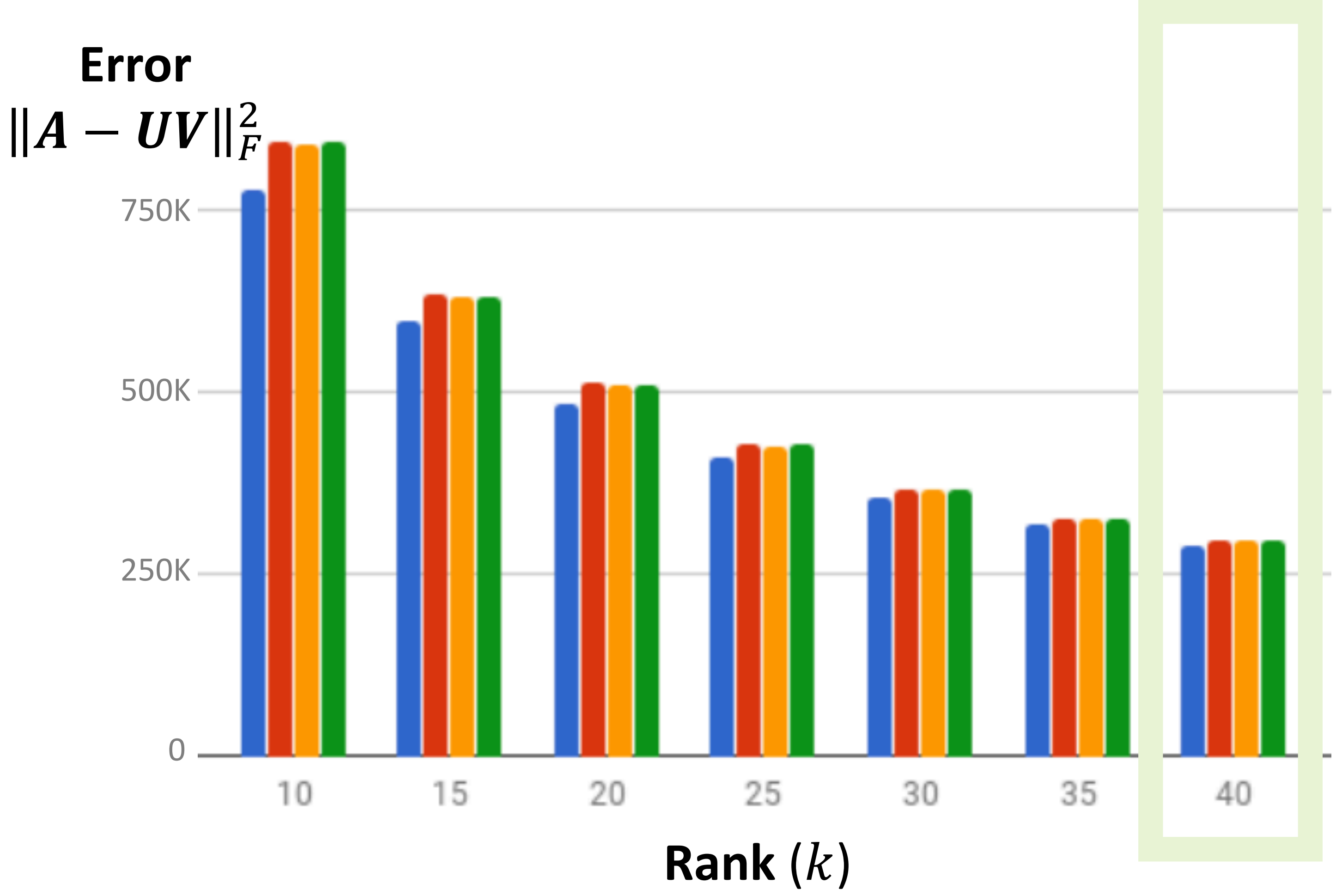
On the other hand:

$$\mathbb{E}_Z\left[\sum_x \left(n \cdot \mathbf{d}(x,Z)^2 + \sum_y \mathbf{d}(Z,y)^2\right)\right] = 2\sum_x \sum_y d(x,y)^2$$

Z is uniformly random

$\Rightarrow p_x \sim \sum_y \mathbf{d}(x,y)^2$ and $\hat{p}_x \sim n \cdot \mathbf{d}(x,Z)^2 + \sum_y \mathbf{d}(Z,y)^2$ are equivalent distributions. ■

Experiment: MNIST, Euclidean Distance



Method	Analytic runtime	Empirical runtime (secs, $k = 40$)
SVD	$O(n^3)$	398.50
[CW'13]	$\tilde{O}(n^2)$	34.32
[BW'18]	$\tilde{O}(n^{1+\gamma})$	4.17
Ours	$\tilde{O}(n)$	1.23

Thank you