

# COMPSCI 514: Algorithms for Data Science

---

Cameron Musco

University of Massachusetts Amherst. Spring 2026.

Lecture 13

- Problem Set 3 posted last night. Due next Friday 4/10 at 11:59pm.
- Given the timing of Midterm 2, Problem Set 4 might be 'split' into two halves, due before and after the midterm.
- Based on feedback, I am going to try to pick specific topics to do review and practice questions on during office hours. Today I will review **eigenvectors and eigenvalues**.
- You can still come to office hours with general questions unrelated to this topic, as usual.

# Summary

## Last Class:

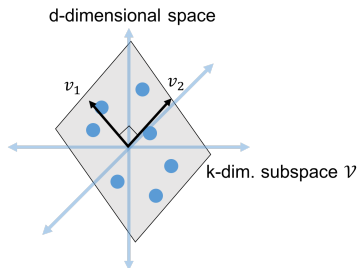
- No-distortion embeddings for data lying in a  $k$ -dimensional subspace via an orthonormal basis  $\mathbf{V} \in \mathbb{R}^{d \times k}$  for that subspace.
- View as low-rank matrix factorization. Introduce concept of low-rank approximation.
- Idea of approximating a data matrix  $\mathbf{X}$  with  $\mathbf{XV}^T$  when the data points lie close to the subspace spanned by  $\mathbf{V}$ 's columns.

## This Class:

- 'Dual view' of low-rank approximation: data points that can be approximately reconstructed from a few basis vectors vs. linearly dependent features.
- Finding an optimal orthogonal basis  $\mathbf{V} \in \mathbb{R}^{d \times k}$  to minimize  $\|\mathbf{X} - \mathbf{XV}^T\|_F^2$  when our data does not exactly lie in a low-dimensional subspace, via eigendecomposition.

## Last Class: Embedding with Assumptions

**Set Up:** Assume that data points  $\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$  lie in some  $k$ -dimensional subspace  $\mathcal{V}$  of  $\mathbb{R}^d$ .



Let  $\vec{v}_1, \dots, \vec{v}_k$  be an orthonormal basis for  $\mathcal{V}$  and  $\mathbf{V} \in \mathbb{R}^{d \times k}$  be the matrix with these vectors as its columns.

$$\|\mathbf{V}^T \vec{x}_i - \mathbf{V}^T \vec{x}_j\|_2^2 = \|\vec{x}_i - \vec{x}_j\|_2^2.$$

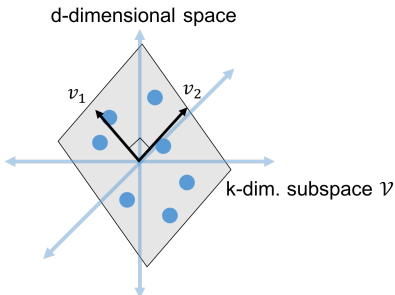
Letting  $\tilde{x}_i = \mathbf{V}^T \vec{x}_i$ , we have a perfect embedding from  $\mathcal{V}$  into  $\mathbb{R}^k$ .

# Last Class: Low-Rank Factorization

**Claim:** If  $\vec{x}_1, \dots, \vec{x}_n$  lie in a  $k$ -dimensional subspace  $\mathcal{V}$  with orthonormal basis  $\mathbf{V} \in \mathbb{R}^{d \times k}$ , the data matrix can be written as

$$\mathbf{X} = \mathbf{X}\mathbf{V}\mathbf{V}^T \text{ (implies } \text{rank}(\mathbf{X}) \leq k \text{)}$$

- $\mathbf{V}\mathbf{V}^T$  is a **projection matrix**, which projects the rows of  $\mathbf{X}$  (the data points  $\vec{x}_1, \dots, \vec{x}_n$ ) onto the subspace  $\mathcal{V}$ .

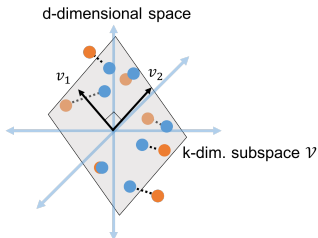


$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $\mathbf{X} \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : orthogonal basis for subspace  $\mathcal{V}$ .  $\mathbf{V} \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$ .

# Low-Rank Approximation

**Claim:** If  $\vec{x}_1, \dots, \vec{x}_n$  lie **close to** a  $k$ -dimensional subspace  $\mathcal{V}$  with orthonormal basis  $\mathbf{V} \in \mathbb{R}^{d \times k}$ , the data matrix can be **approximated as:**

$$\mathbf{X} \approx \mathbf{XV}^T$$



**Note:**  $\mathbf{XV}^T$  has rank  $k$ . It is a **low-rank approximation** of  $\mathbf{X}$ .

$$\mathbf{XV}^T = \underset{\mathbf{B} \text{ with rows in } \mathcal{V}}{\arg \min} \|\mathbf{X} - \mathbf{B}\|_F^2 = \sum_{i,j} (X_{i,j} - B_{i,j})^2.$$

$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $\mathbf{X} \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : orthogonal basis for subspace  $\mathcal{V}$ .  $\mathbf{V} \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$ .

# This Class: Low-Rank Approximation

**So Far:** If  $\vec{x}_1, \dots, \vec{x}_n$  lie close to a  $k$ -dimensional subspace  $\mathcal{V}$  with orthonormal basis  $\mathbf{V} \in \mathbb{R}^{d \times k}$ , the data matrix can be approximated as:

$$\mathbf{X} \approx \mathbf{XV}^T.$$

This is the closest approximation to  $\mathbf{X}$  with rows in  $\mathcal{V}$  (i.e., in the column span of  $\mathbf{V}$ ).

- Letting  $\mathbf{V}\mathbf{V}^T\vec{x}_i, \mathbf{V}\mathbf{V}^T\vec{x}_j$  be the  $i^{\text{th}}$  and  $j^{\text{th}}$  projected data points,

$$\|\mathbf{V}\mathbf{V}^T\vec{x}_i - \mathbf{V}\mathbf{V}^T\vec{x}_j\|_2 = \|\mathbf{V}^T\mathbf{V}\mathbf{V}^T\vec{x}_i - \mathbf{V}^T\mathbf{V}\mathbf{V}^T\vec{x}_j\|_2 = \|\mathbf{V}^T\vec{x}_i - \mathbf{V}^T\vec{x}_j\|_2.$$

- I.e., we can use the rows of  $\mathbf{XV} \in \mathbb{R}^{n \times k}$  as a compressed approximate data set.

Key question is how to find the subspace  $\mathcal{V}$  and correspondingly  $\mathbf{V}$ .

$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $\mathbf{X} \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : orthogonal basis for subspace  $\mathcal{V}$ .  $\mathbf{V} \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$ .

## Properties of Projection Matrices

**Quick Exercise 1:** Show that  $\mathbf{V}\mathbf{V}^T$  is idempotent. I.e.,  $(\mathbf{V}\mathbf{V}^T)(\mathbf{V}\mathbf{V}^T)\vec{y} = (\mathbf{V}\mathbf{V}^T)\vec{y}$  for any  $\vec{y} \in \mathbb{R}^d$ .

**Quick Exercise 2:** Show that  $\mathbf{V}\mathbf{V}^T(\mathbf{I} - \mathbf{V}\mathbf{V}^T) = \mathbf{0}$  ( the projection is orthogonal to its complement).

# Pythagorean Theorem

**Pythagorean Theorem:** For any orthonormal  $\mathbf{V} \in \mathbb{R}^{d \times k}$  and any  $\vec{y} \in \mathbb{R}^d$ ,

$$\|\vec{y}\|_2^2 = \|(\mathbf{V}\mathbf{V}^T)\vec{y}\|_2^2 + \|\vec{y} - (\mathbf{V}\mathbf{V}^T)\vec{y}\|_2^2.$$

# A Step Back: Why Low-Rank Approximation?

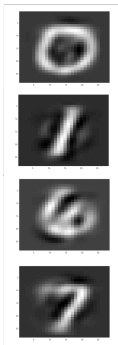
**Question:** Why might we expect  $\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$  to lie close to a  $k$ -dimensional subspace?

- The rows of  $X$  can be approximately reconstructed from a basis of  $k$  vectors.

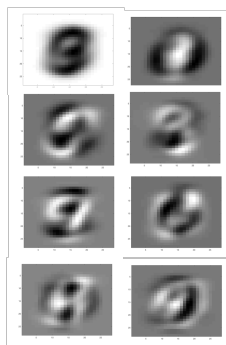
784 dimensional vectors



projections onto 15  
dimensional space



orthonormal basis  $v_1, \dots, v_{15}$



# Dual View of Low-Rank Approximation

**Question:** Why might we expect  $\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$  to lie close to a  $k$ -dimensional subspace?

- Equivalently, the columns of  $\mathbf{X}$  are approx. spanned by  $k$  vectors.

**Linearly Dependent Variables:**

	bedrooms	bathrooms	sq.ft.	floors	list price	sale price
home 1	2	2	1800	2	200,000	195,000
home 2	4	2.5	2700	1	300,000	310,000
⋮	⋮	⋮	⋮	⋮	⋮	⋮
home n	5	3.5	3600	3	450,000	450,000

	bedrooms
home 1	2
home 2	4
⋮	⋮
home n	5 <sup>11</sup>

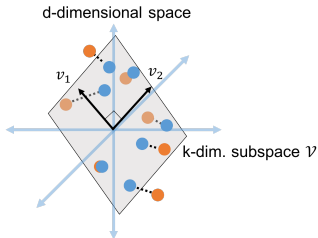
# Best Fit Subspaces

# Best Fit Subspace

If  $\vec{x}_1, \dots, \vec{x}_n$  are close to a  $k$ -dimensional subspace  $\mathcal{V}$  with orthonormal basis  $\mathbf{V} \in \mathbb{R}^{d \times k}$ , the data matrix can be approximated as  $\mathbf{X}\mathbf{V}\mathbf{V}^T$ .  $\mathbf{X}\mathbf{V}$  gives optimal embedding of  $\mathbf{X}$  in  $\mathcal{V}$ .

How do we find  $\mathcal{V}$  (equivalently  $\mathbf{V}$ )?

$$\arg \min_{\text{orthonormal } \mathbf{V} \in \mathbb{R}^{d \times k}} \|\mathbf{X} - \mathbf{X}\mathbf{V}\mathbf{V}^T\|_F^2 = \sum_{i,j} (\mathbf{X}_{i,j} - (\mathbf{X}\mathbf{V}\mathbf{V}^T)_{i,j})^2 = \sum_{i=1}^n \|\vec{x}_i - \mathbf{V}\mathbf{V}^T\vec{x}_i\|_2^2$$



$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $\mathbf{X} \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : orthogonal basis for subspace  $\mathcal{V}$ .  $\mathbf{V} \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$ .

# Solution via Eigendecomposition

$\mathbf{V}$  minimizing  $\|\mathbf{X} - \mathbf{X}\mathbf{V}\mathbf{V}^T\|_F^2$  is given by:

$$\arg \max_{\text{orthonormal } \mathbf{V} \in \mathbb{R}^{d \times k}} \|\mathbf{X}\mathbf{V}\|_F^2 = \sum_{i=1}^n \|\mathbf{V}^T \vec{x}_i\|_2^2 = \sum_{j=1}^k \|\mathbf{X}\vec{v}_j\|_2^2$$

Surprisingly, can find the columns of  $\mathbf{V}$ ,  $\vec{v}_1, \dots, \vec{v}_k$  **greedily**.

$$\vec{v}_1 = \arg \max_{\vec{v} \text{ with } \|\vec{v}\|_2=1} \|\mathbf{X}\vec{v}\|_2^2$$

$$\vec{v}_2 = \arg \max_{\vec{v} \text{ with } \|\vec{v}\|_2=1, \langle \vec{v}, \vec{v}_1 \rangle = 0} \|\mathbf{X}\vec{v}\|_2^2$$

...

$$\vec{v}_k = \arg \max_{\vec{v} \text{ with } \|\vec{v}\|_2=1, \langle \vec{v}, \vec{v}_j \rangle = 0 \forall j < k} \|\mathbf{X}\vec{v}\|_2^2.$$

$\vec{v}_1, \dots, \vec{v}_k$  are the top  $k$  eigenvectors of  $\mathbf{X}^T\mathbf{X}$  by the *Courant-Fischer Principle*.

$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $\mathbf{X} \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : orthogonal basis for subspace  $\mathcal{V}$ .  $\mathbf{V} \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$ .

# Review of Eigenvectors and Eigendecomposition

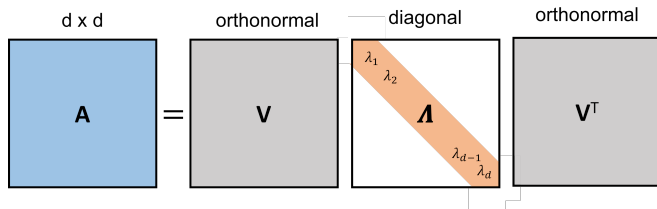
**Eigenvector:**  $\vec{x} \in \mathbb{R}^d$  is an eigenvector of a matrix  $\mathbf{A} \in \mathbb{R}^{d \times d}$  if  $\mathbf{A}\vec{x} = \lambda\vec{x}$  for some scalar  $\lambda$  (the eigenvalue corresponding to  $\vec{x}$ ).

- That is,  $\mathbf{A}$  just 'stretches'  $x$ .
- If  $\mathbf{A}$  is **symmetric**, can find  $d$  orthonormal eigenvectors  $\vec{v}_1, \dots, \vec{v}_d$ . Let  $\mathbf{V} \in \mathbb{R}^{d \times d}$  have these vectors as columns.

$$\mathbf{AV} = \begin{bmatrix} | & | & | & | \\ \mathbf{A}\vec{v}_1 & \mathbf{A}\vec{v}_2 & \cdots & \mathbf{A}\vec{v}_d \\ | & | & | & | \end{bmatrix} = \begin{bmatrix} | & | & | & | \\ \lambda_1\vec{v}_1 & \lambda_2\vec{v}_2 & \cdots & \lambda\vec{v}_d \\ | & | & | & | \end{bmatrix} = \mathbf{V}\mathbf{\Lambda}$$

Yields eigendecomposition:  $\mathbf{AVV}^T = \mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ .

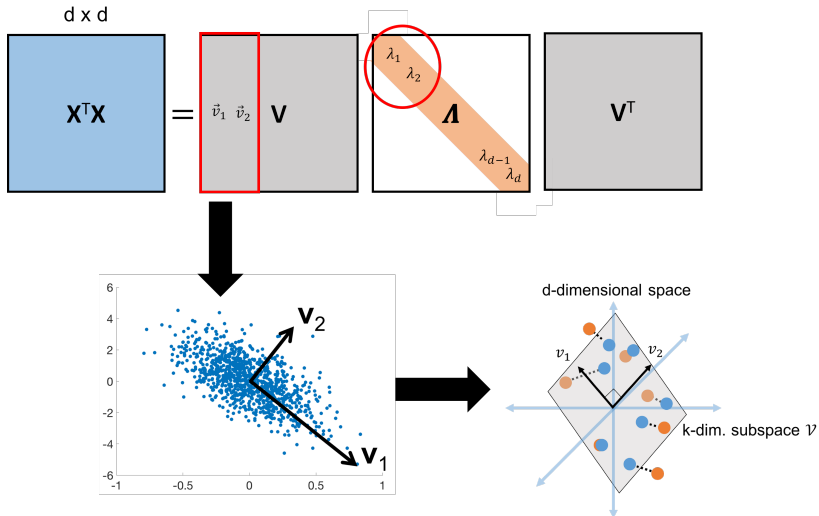
# Review of Eigenvectors and Eigendecomposition



Typically order the eigenvectors in decreasing order:

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d.$$

# Low-Rank Approximation via Eigendecomposition



# Low-Rank Approximation via Eigendecomposition

**Upshot:** Letting  $\mathbf{V}_k$  have columns  $\vec{v}_1, \dots, \vec{v}_k$  corresponding to the top  $k$  eigenvectors of the covariance matrix  $\mathbf{X}^T\mathbf{X}$ ,  $\mathbf{V}_k$  is the orthogonal basis minimizing

$$\|\mathbf{X} - \mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2.$$

This is principal component analysis (PCA).

How accurate is this low-rank approximation? Can understand using eigenvalues of  $\mathbf{X}^T\mathbf{X}$ .

$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $\mathbf{X} \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : top eigenvectors of  $\mathbf{X}^T\mathbf{X}$ ,  $\mathbf{V}_k \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$ .

# Spectrum Analysis

Let  $\vec{v}_1, \dots, \vec{v}_k$  be the top  $k$  eigenvectors of  $\mathbf{X}^T\mathbf{X}$  (the top  $k$  principal components). Approximation error is:

$$\begin{aligned}\|\mathbf{X} - \mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2 &= \|\mathbf{X}\|_F^2 \operatorname{tr}(\mathbf{X}^T\mathbf{X}) - \|\mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2 \operatorname{tr}(\mathbf{V}_k^T\mathbf{X}^T\mathbf{X}\mathbf{V}_k) \\ &= \sum_{i=1}^d \lambda_i(\mathbf{X}^T\mathbf{X}) - \sum_{i=1}^k \vec{v}_i^T \mathbf{X}^T\mathbf{X} \vec{v}_i \\ &= \sum_{i=1}^d \lambda_i(\mathbf{X}^T\mathbf{X}) - \sum_{i=1}^k \lambda_i(\mathbf{X}^T\mathbf{X}) = \sum_{i=k+1}^d \lambda_i(\mathbf{X}^T\mathbf{X})\end{aligned}$$

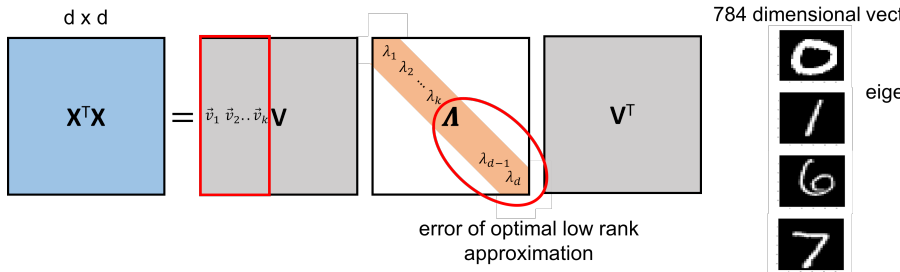
- **Exercise:** For any matrix  $\mathbf{A}$ ,  $\|\mathbf{A}\|_F^2 = \sum_{i=1}^d \|\vec{a}_i\|_2^2 = \operatorname{tr}(\mathbf{A}^T\mathbf{A})$  (sum of diagonal entries = sum eigenvalues).

$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $\mathbf{X} \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : top eigenvectors of  $\mathbf{X}^T\mathbf{X}$ ,  $\mathbf{V}_k \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$ .

# Spectrum Analysis

**Claim:** The error in approximating  $X$  with the best rank  $k$  approximation (projecting onto the top  $k$  eigenvectors of  $X^T X$ ) is:

$$\|X - X V_k V_k^T\|_F^2 = \sum_{i=k+1}^d \lambda_i(X^T X)$$



- Choose  $k$  to balance accuracy/compression – often at an ‘elbow’.

$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $X \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : top eigenvectors of  $X^T X$ ,  $V_k \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$

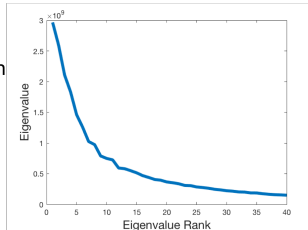
# Spectrum Analysis

Plotting the **spectrum** of  $X^T X$  (its eigenvalues) shows how compressible  $X$  is using low-rank approximation (i.e., how close  $\vec{x}_1, \dots, \vec{x}_n$  are to a low-dimensional subspace).

784 dimensional vectors



eigendecomposition



784 dimensional vectors



eigende

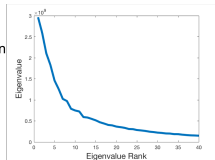
$\vec{x}_1, \dots, \vec{x}_n \in \mathbb{R}^d$ : data points,  $X \in \mathbb{R}^{n \times d}$ : data matrix,  $\vec{v}_1, \dots, \vec{v}_k \in \mathbb{R}^d$ : top eigenvectors of  $X^T X$ ,  $V_k \in \mathbb{R}^{d \times k}$ : matrix with columns  $\vec{v}_1, \dots, \vec{v}_k$ .

# Spectrum Analysis

784 dimensional vectors



eigendecomposition



## Exercises:

1. Show that the eigenvalues of  $X^T X$  are always positive. **Hint:** Use that  $\lambda_j = \vec{v}_j^T X^T X \vec{v}_j$ .
2. Show that for symmetric  $A$ , the trace is the sum of eigenvalues:  $\text{tr}(A) = \sum_{i=1}^n \lambda_i(A)$ . **Hint:** First prove the **cyclic property** of trace, that for any  $MN$ ,  $\text{tr}(MN) = \text{tr}(NM)$  and then apply this to  $A$ 's eigendecomposition

# Summary

- Many (most) datasets can be approximated via projection onto a low-dimensional subspace.
- Find this subspace via a maximization problem:

$$\max_{\text{orthonormal } \mathbf{V}} \|\mathbf{XV}\|_F^2.$$

- Greedy solution via eigendecomposition of  $\mathbf{X}^T\mathbf{X}$ .
- Columns of  $\mathbf{V}$  are the top eigenvectors of  $\mathbf{X}^T\mathbf{X}$ .
- Error of best low-rank approximation (compressibility of data) is determined by the tail of  $\mathbf{X}^T\mathbf{X}$ 's eigenvalue spectrum.