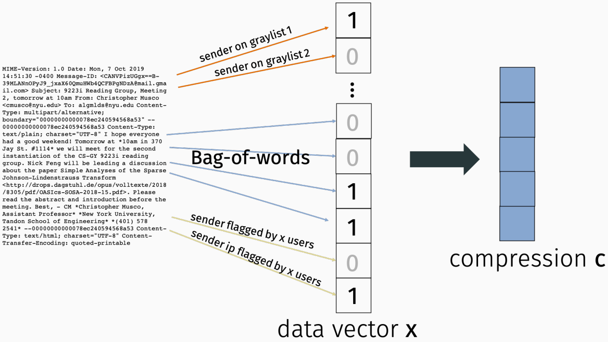
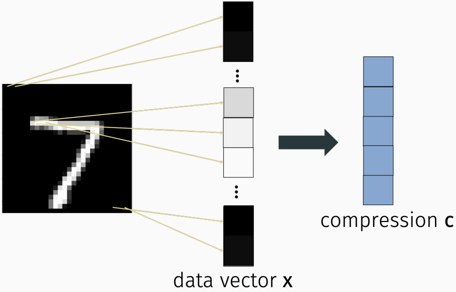
CS-GY 9223 D: Lecture 9
Low-rank approximation and singular value decomposition

NYU Tandon School of Engineering, Prof. Christopher Musco

Return to data compression:

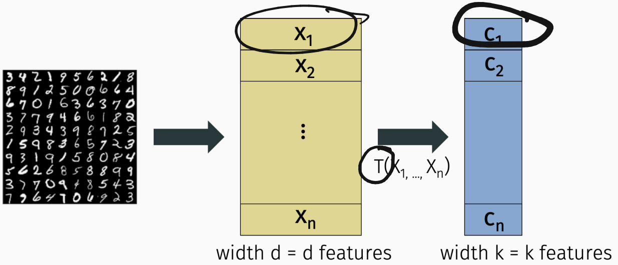


Return to data compression:



Main difference from randomized methods:

$c_1 = \Pi x_1$



In this section, we will discuss data dependent transformations. Johnson-Lindenstrauss, MinHash, SimHash were all data oblivious.

Advantages of data independent methods:

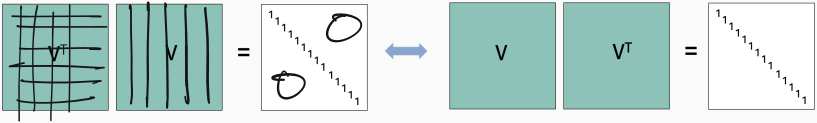
- less computational cost
- easy to analyze, clean + general bounds
- more easily distributed + streams

Advantages of data dependent methods:

- better compression, in practice
- don't use randomness
- more interpretable.

If a square matrix has orthonormal rows, it also have orthonormal columns:

orthogonal matrix



$$V^T V = I = V V^T$$

$$(Vx)^T Vx = x^T \underbrace{V^T V}_{I} x = x^T x = \|x\|_2^2$$

Implies that for any vector x, $\|Vx\|_2^2 = \|x\|_2^2$ and $\|V^T x\|_2^2$.

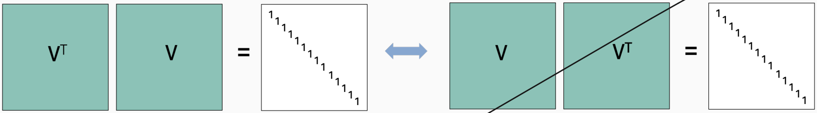
Equivalently, any vector x, $\|x^T V^T\|_2^2 = \|x\|_2^2$ and $\|x^T V\|_2^2 = \|x\|_2^2$.

Same thing goes for Frobenius norm: for any matrix X, $\|VX\|_F^2 = \|X\|_F^2$ and $\|V^T X\|_F^2 = \|X\|_F^2$.

$\|x_1\|_2^2 + \|x_2\|_2^2 + \dots \|x_n\|_2^2 = \|X\|_F^2$

$= \|Vx_1\|_2^2 + \|Vx_2\|_2^2 + \dots \|Vx_n\|_2^2$ where x_i is ith column of X.

If a square matrix has orthonormal rows, it also have orthonormal columns:



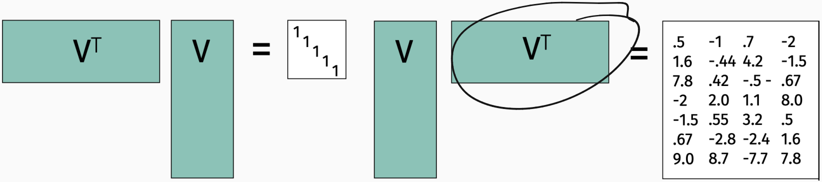
$$\mathbf{V}^T\mathbf{V} = \mathbf{I} = \mathbf{V}\mathbf{V}^T$$

Implies that for any vector $\mathbf{x}$, $\|\mathbf{V}\mathbf{x}\|_2^2 = \|\mathbf{x}\|_2^2$ and $\|\mathbf{V}^T\mathbf{x}\|_2^2$.

Equivalently, any vector $\mathbf{x}$, $\|\mathbf{x}^T\mathbf{V}^T\|_2^2 = \|\mathbf{x}\|_2^2$ and $\|\mathbf{x}^T\mathbf{V}\|_2^2 = \|\mathbf{x}\|_2^2$.

Same thing goes for Frobenius norm: for any matrix $\mathbf{X}$, $\|\mathbf{V}\mathbf{X}\|_F^2 = \|\mathbf{X}\|_F^2$ and $\|\mathbf{V}^T\mathbf{X}\|_F^2 = \|\mathbf{X}\|_F^2$.

The same is not true for rectangular matrices:



$\mathbf{V}^T\mathbf{V} = \mathbf{I}$ but $\mathbf{V}\mathbf{V}^T \neq \mathbf{I}$

$\|Vx\|_2^2 = x^T V^T V x = x^T x$

For any $\mathbf{x}$, $\|\mathbf{V}\mathbf{x}\|_2^2 = \|\mathbf{x}\|_2^2$ but $\|\mathbf{V}^T\mathbf{x}\|_2^2 \neq \|\mathbf{x}\|_2^2$ in general.

Equivalently, $\mathbf{x}$, $\|\mathbf{x}^T\mathbf{V}^T\|_2^2 = \|\mathbf{x}\|_2^2$ but $\|\mathbf{x}^T\mathbf{V}\|_2^2 \neq \|\mathbf{x}\|_2^2$ in general.

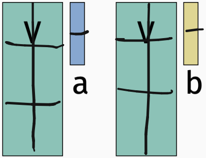
$(V^T x)^T (V^T x) = x^T V V^T x \neq x^T x$

Multiplying a vector by **V** with orthonormal columns <u>rotates and/or reflects</u> the vector.

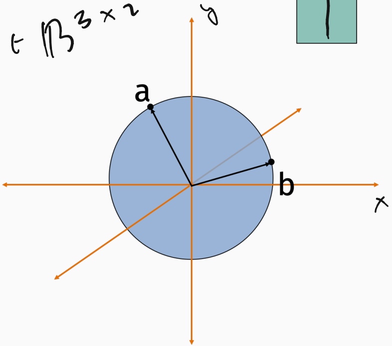
$a \in \mathbb{R}^2$

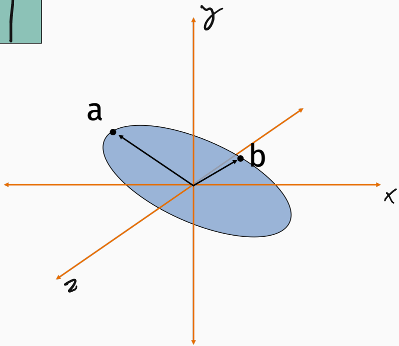
$Va \in \mathbb{R}^3$

$V \in \mathbb{R}^{3 \times 2}$

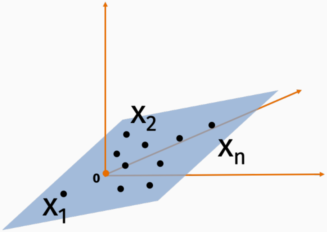


$\langle Va, Vb \rangle = a^T V^T V b = a^T b = \langle a, b \rangle$

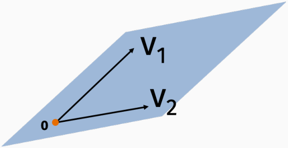




Suppose $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ lie on a low-dimensional subspace S through the origin. I.e. our data set is rank k for $k < d$.

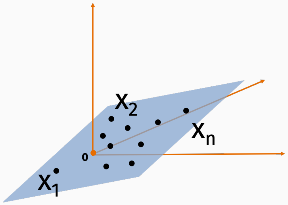


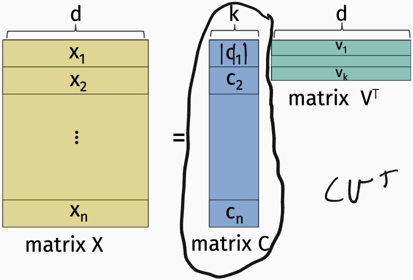
Let $\mathbf{v}_1, \ldots, \mathbf{v}_k$ be orthogonal unit vectors spanning S.



For all i, we can write:

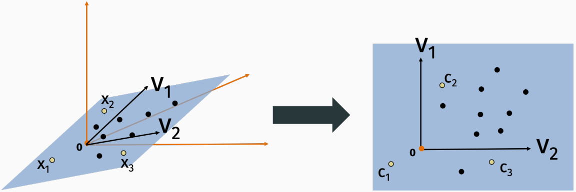
$$\mathbf{x}_i = c_{i,1}\mathbf{v}_1 + \ldots + c_{i,k}\mathbf{v}_k.$$





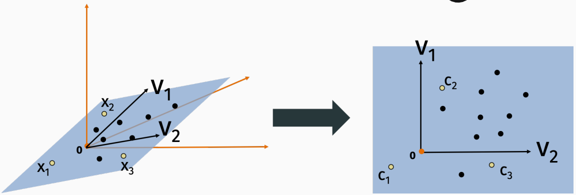
CV^T

What are $\mathbf{c}_1, \ldots, \mathbf{c}_n$?



$X = CV^T \qquad XV = CV^TV \qquad XV = C$

$V^TV = I$



$y = Vz$

Lots of information preserved:

- $\|\mathbf{x}_i - \mathbf{x}_j\|_2 = \|\mathbf{c}_i - \mathbf{c}_j\|_2$ for all i, j.
- $\mathbf{x}_i^T\mathbf{x}_j = \mathbf{c}_i^T\mathbf{c}_j$ for all i, j.
- Norms preserved, linear separability preserved, $\min \|\mathbf{X}\mathbf{y} - \mathbf{b}\| = \min \|\mathbf{C}\mathbf{z} - \mathbf{b}\|$, etc., etc.

$c_i V^T$

$\|c_i V^T - c_j V^T\|_2^2$

$\|(c_i - c_j)V^T\|_2^2$

$= \|V(c_i - c_j)^T\|_2^2$

$= \|c_i - c_j\|_2^2$

$\|Xy - b\|_2^2 = \|CV^TVz - b\|_2^2$

$= \|Cz - b\|_2^2$

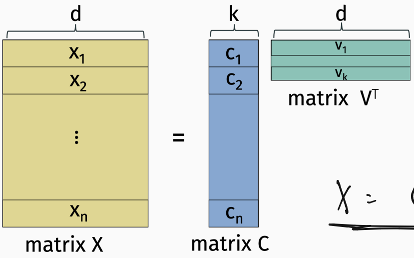
$X = CV^T$

V^T

y

$= V^Ty$

$y = Vz$



$X = CV^T$

$XV = CV^TV$

$XV = C$

Formally, $C = XV$.

$$X = CV^T \Rightarrow XV = CV^TV$$

Since V's columns are an orthonormal basis, $V^TV = I$.

$X = CV^T$

$\downarrow$

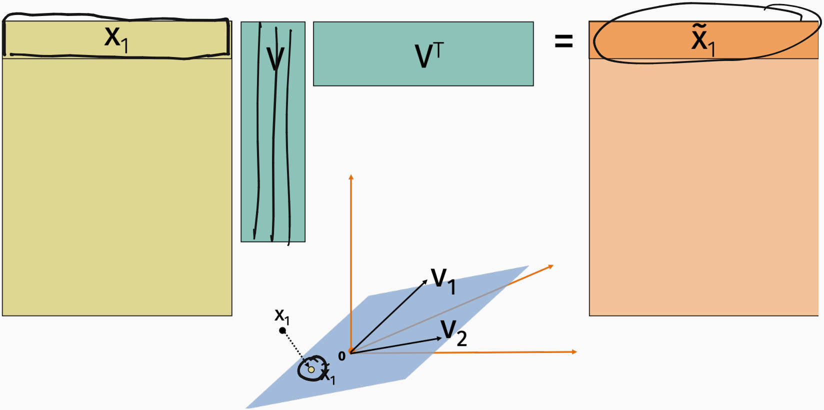
XV $\quad X = XVV^T$

So $X = XVV^T$.

$A = AVV^T$

VV^T is a symmetric projection matrix.

$x_i^T (VV^T)$



When all data points already lie in the subspace spanned by V's columns, projection doesn't do anything. So $X = XVV^T$.

When $\mathbf{X}$'s rows lie close to a k dimensional subspace, we can still approximate

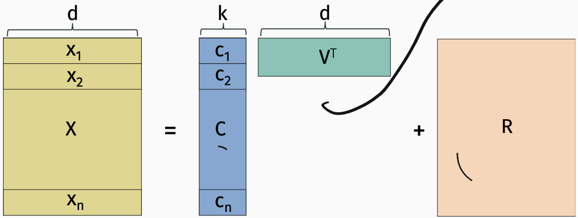
$$\mathbf{X} \approx \mathbf{X}\mathbf{V}\mathbf{V}^T.$$

$\mathbf{X}\mathbf{V}\mathbf{V}^T$ is a low-rank approximation for $\mathbf{X}$.

For a given subspace $\mathcal{V}$ spanned by the columns in $\mathbf{V}$,

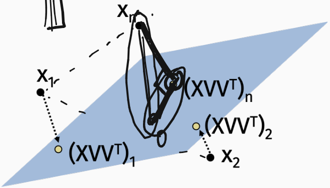
$$\mathbf{X}\mathbf{V} = \arg\min_{\mathbf{C}} \|\mathbf{X} - \mathbf{C}\mathbf{V}^T\|_F^2 = \sum_{i,j} (\mathbf{X}_{i,j} - (\mathbf{C}\mathbf{V}^T)_{i,j})^2.$$

rank k matrix



$\|Va\|_2^2 = \|a\|_2^2$

$\|a^T V^T\|_2^2 = \|a^T\|_2^2$



$(XV)_i$

$= x_i^T V$

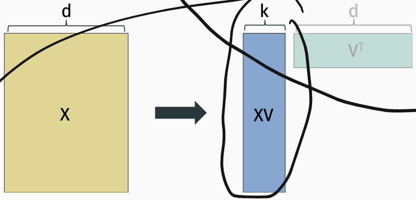
$\|x_i^T V - x_j^T V\|_2$

$= \|(x_i - x_j)^T V\|_2^2$

$$\|\mathbf{x}_i - \mathbf{x}_j\|_2 \approx \|(\mathbf{XVV}^T)_i - (\mathbf{XVV}^T)_j\|_2 = \|(\mathbf{XV})_i - (\mathbf{XV})_j\|_2$$

$\|a^T V^T\|_2$

$\|a^T\|_2$



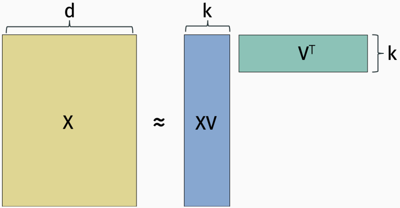
$\|(x_i - x_j^T) V V^T\|_2^2$

$a^T = (x_i - x_j)^T V$

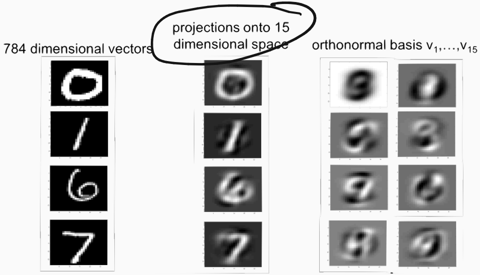
$\mathbf{XV}$ can be used as a compressed version of data matrix $\mathbf{X}$.

WHY IS DATA APPROXIMATELY LOW-RANK?

Rows of $\mathbf{X}$ (data points) are approximately spanned by k vectors. Columns of $\mathbf{X}$ (data features) are approximately spanned by k vectors.



If a data set only had k unique data points, it would be exactly rank k. If it has k "clusters" of data points (e.g. the 10 digits) it's often very close to rank k.



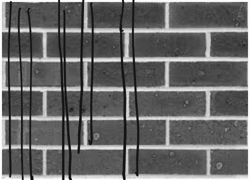
Colinearity/correlation of data features leads to a low-rank data matrix.

	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

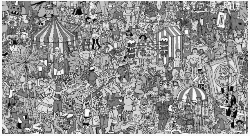
$$\|\underbrace{(c_i - c_j) V^T}_{\text{transpose}}\|_2^2 \qquad \|V \underbrace{(c_i - c_j)^T}_{\alpha}\|_2^2 = \|V\alpha\|_2^2 = \alpha^T V^T V \alpha = \|\alpha\|_2^2$$

When encoded as a matrix, which image has lower approximate rank?

1

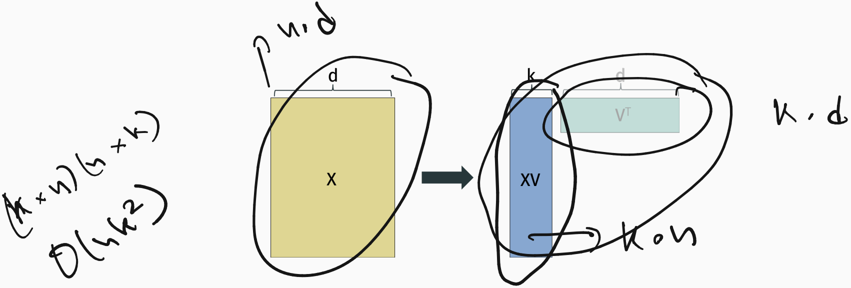


2



3

2, 1, 3

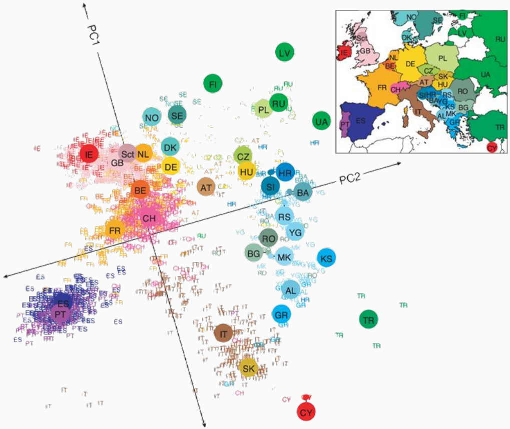


- $\mathbf{XV} \cdot \mathbf{V}^T$ takes $O(k(n+d))$ space to store instead of $O(nd)$.
- Regression problems involving $\mathbf{XV} \cdot \mathbf{V}^T$ can be solved in $O(nk^2)$ instead of $O(nd^2)$ time.
- $\mathbf{XV}$ can be used for visualization when $k = 2, 3$.

$\min \|XVV^T z - b\|_2^2$ $\qquad$ $\min_z \|XVz - b\|_2^2 \rightarrow O(nk)$

$y = Vz$ $\qquad$ A $\qquad$ $(A^T A)^{-1} A^T b$

"Genes Mirror Geography Within Europe" – Nature, 2008.



Each data vector $\mathbf{x}_i$ contains genetic information for one person in Europe. Set $k = 2$ and plot $(XV)_i$ for each i on a 2-d plane. Color points by what country they are from.

Given a subspace $\mathcal{V}$ spanned by the k columns in $\mathbf{V}$,

$$\|\mathbf{X} - \mathbf{X}\mathbf{V}\mathbf{V}^T\|_F^2 = \min_{\mathbf{C}} \|\mathbf{X} - \mathbf{C}\mathbf{V}^T\|_F^2$$

✓

$\|X\|_F^2 - \|XVV^T\|_F^2$

=

We want to find the best $\mathbf{V} \in \mathbb{R}^{d \times k}$:

$$\min_{\text{orthonormal } \mathbf{V} \in \mathbb{R}^{d \times k}} \|\mathbf{X} - \mathbf{X}\mathbf{V}\mathbf{V}^T\|_F^2 \tag{1}$$

Note that $\|\mathbf{X} - \mathbf{X}\mathbf{V}\mathbf{V}^T\|_F^2 = \|\mathbf{X}\|_F^2 - \|\mathbf{X}\mathbf{V}\mathbf{V}^T\|_F^2$ for all orthonormal $\mathbf{V}$ (since $\mathbf{V}\mathbf{V}^T$ is a projection). Equivalent form:

$$\max_{\text{orthonormal } \mathbf{V} \in \mathbb{R}^{d \times k}} \|\mathbf{X}\mathbf{V}\mathbf{V}^T\|_F^2 = \|\mathbf{X}\mathbf{V}\|_F^2 \tag{2}$$

$tr[(X - XVV^T)^T(X - XVV^T))$

If $k = 1$, want to find a single vector $\mathbf{v}_1$ which maximizes:

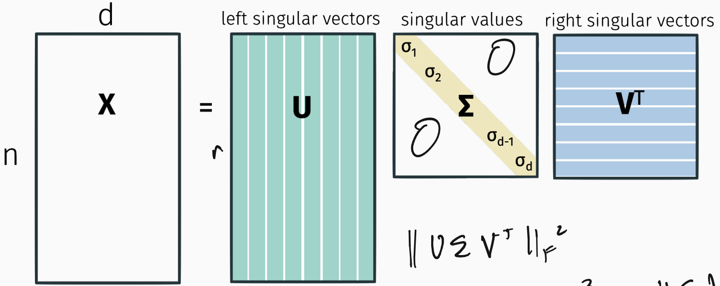
$$\|\mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2 = \|\mathbf{X}\mathbf{v}_1\|_F^2 = \|\mathbf{X}\mathbf{v}_1\|_2^2 = \mathbf{v}_1^T\mathbf{X}^T\mathbf{X}\mathbf{v}_1.$$

Choose $\mathbf{v}_1$ to be the top eigenvector of $\mathbf{X}^T\mathbf{X}$.

What about higher k?

One-stop shop for computing optimal low-rank approximations.

Any matrix **X** can be written:



$$\|U\Sigma V^T\|_F^2 = \|\Sigma V^T\|_F^2 = \|\Sigma\|_F^2$$

Where $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, and $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_d \geq 0$.

Note that $\sum_{i=1}^{d} \sigma_i^2 = \|\mathbf{X}\|_F^2$.

- $\mathbf{V}_k$'s columns are called the "top right singular vectors of $\mathbf{X}$"
- $\mathbf{U}_k$'s columns are called the "top left singular vectors of $\mathbf{X}$"
- $\sigma_1, \ldots, \sigma_k$ are the "top singular values". $\sigma_1, \ldots, \sigma_d$ are sometimes called the "spectrum of $\mathbf{X}$" (although this is more typically used to refer to eigenvalues).

- $\mathbf{U}$ contains the orthonormal eigenvectors of $\mathbf{X}\mathbf{X}^T$
- $\mathbf{V}$ contains the orthonormal eigenvectors of $\mathbf{X}^T\mathbf{X}$.
- $\sigma_i^2 = \lambda_i(\mathbf{X}\mathbf{X}^T) = \lambda_i(\mathbf{X}^T\mathbf{X})$

Exercise: Check this can be checked directly.

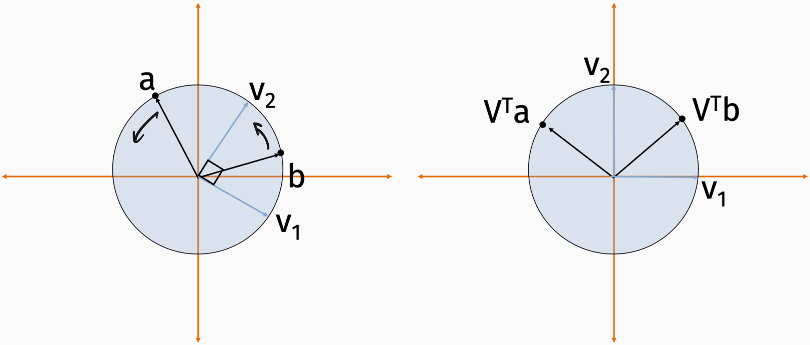
Important take away from singular value decomposition.

$(U \cdot (S \cdot (V^T a)))$

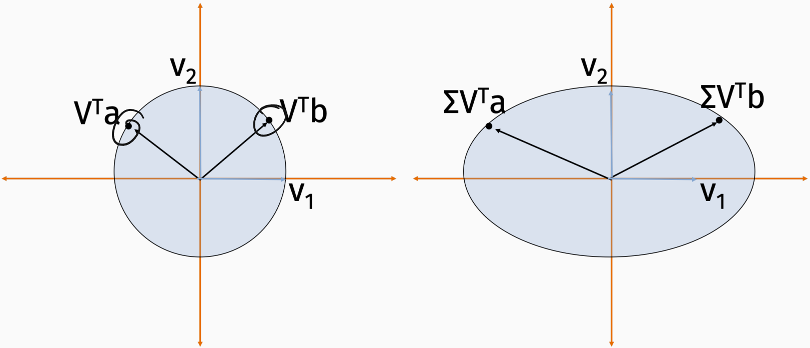
Multiplying any vector $\mathbf{a}$ by a matrix $\mathbf{X}$ to form $\mathbf{Xa}$ can be viewed as a composition of 3 operations:

1. Rotate/reflect the vector (multiplication by to $\mathbf{V}^T$).
2. Scale the coordinates (multiplication by $\mathbf{\Sigma}$.
3. Rotate/reflect the vector again (multiplication by $\mathbf{U}$).

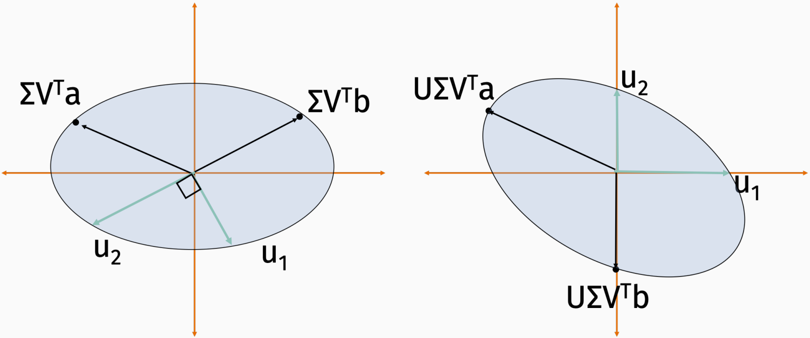
SINGULAR VALUE DECOMPOSITION: ROTATE/REFLECT

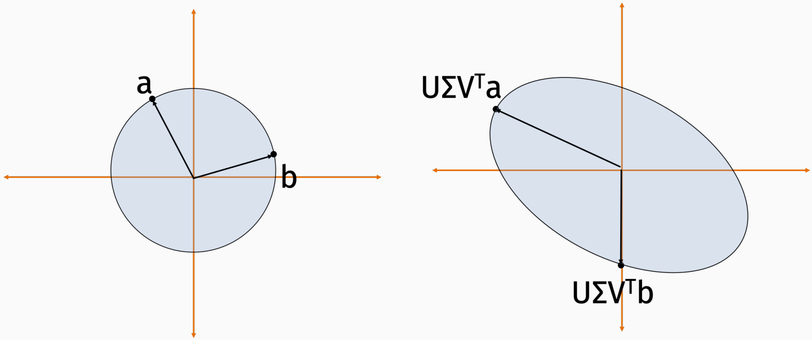


σ_1



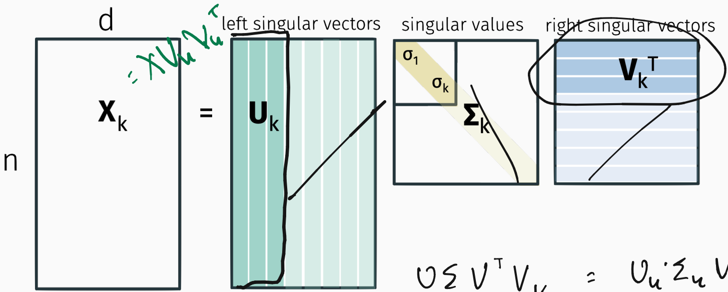
SINGULAR VALUE DECOMPOSITION: ROTATE/REFLECT




a
b
UΣV^Ta
UΣV^Tb

$XV_k = U \Sigma W = U \; [\;] = [u_1\sigma_1 \; u_2\sigma_2 \; \ldots \; u_k\sigma_k]$

Can read off optimal low-rank approximations from the SVD:



$U \Sigma V^T V_k = U_k \Sigma_k V_k^T$

$$X_k = U_k \Sigma_k V_k^T = U_k U_k^T X = X V_k V_k^T.$$

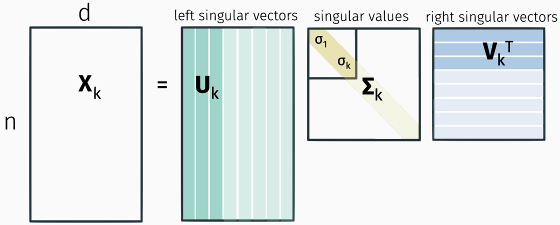
$$V_k = \underset{\text{orthonormal } V \in \mathbb{R}^{d \times k}}{\arg\min} \|X - XVV^T\|_F^2 = \underset{\text{orthonormal } V \in \mathbb{R}^{d \times k}}{\arg\max} \|XVV^T\|_F^2$$

$U_k \quad \Sigma_k$

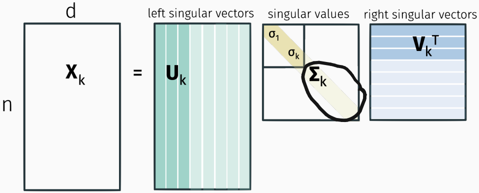
$X V_k V_k^T \qquad U \Sigma V^T V_k \qquad [V_k^T] \, [V_k] = [\;] \to W$

Connection to Principal Component Analysis:

- Let $\bar{\mathbf{X}} = \mathbf{X} - \mathbf{1}\boldsymbol{\mu}^T$ where $\boldsymbol{\mu} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$. I.e. $\bar{\mathbf{X}}$ is obtained by <u>mean centering</u> $\mathbf{X}$'s rows.
- Let $\bar{\mathbf{U}}\bar{\boldsymbol{\Sigma}}\bar{\mathbf{V}}^T$ be the SVD of $\bar{\mathbf{X}}$. $\bar{\mathbf{U}}$'s first columns are the "top principal components" of $\mathbf{X}$. $\mathbf{V}$'s first columns are the "weight vectors" for these principal components.



Observation 1: The optimal compression $\mathbf{X}\mathbf{V}_k$ has orthogonal columns.



Observation 2: The optimal low-rank approximation error $E_k = \underbrace{\|X - XV_kV_k^T\|_F^2}_{E_k} = \|X\|_F^2 - \|XV_kV_k^T\|_F^2$ can be written:

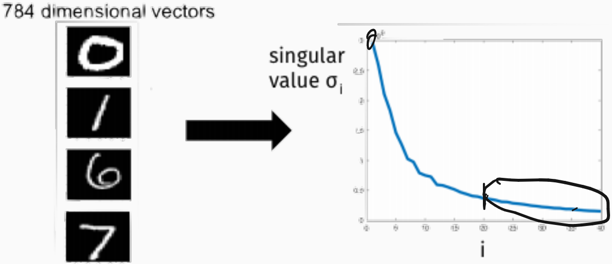
$$E_k = \sum_{i=k+1}^{d} \sigma_i^2.$$

$$\sum_{i=1}^{d} \sigma_i^2 - \sum_{i=1}^{k} \sigma_i^2 = \sum_{i=k+1}^{d} \sigma_i^2$$

Observation 2: The optimal low-rank approximation error $E_k = \|\mathbf{X} - \mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2 = \|\mathbf{X}\|_F^2 - \|\mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2$ can be written:

$$E_k = \sum_{i=k+1}^{d} \sigma_i^2.$$

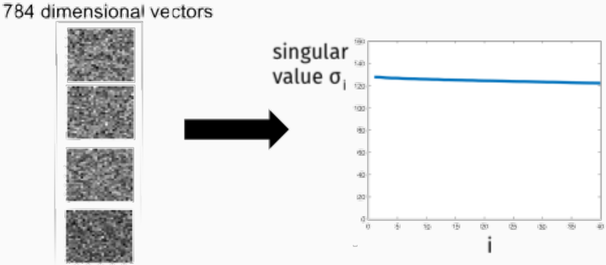
Can immediately get a sense of "how low-rank" a matrix is from it's spectrum:



Observation 2: The optimal low-rank approximation error $E_k = \|\mathbf{X} - \mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2 = \|\mathbf{X}\|_F^2 - \|\mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2$ can be written:

$$E_k = \sum_{i=k+1}^{d} \sigma_i^2.$$

Can immediately get a sense of "how low-rank" a matrix is from it's spectrum:

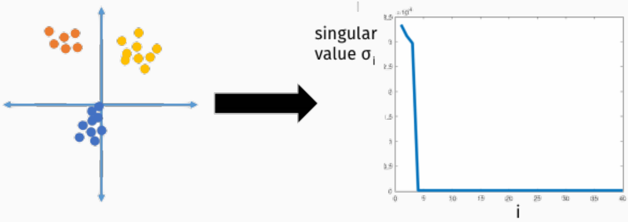


SPECTRAL PLOTS

Observation 2: The optimal low-rank approximation error $E_k = \|\mathbf{X} - \mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2 = \|\mathbf{X}\|_F^2 - \|\mathbf{X}\mathbf{V}_k\mathbf{V}_k^T\|_F^2$ can be written:

$$E_k = \sum_{i=k+1}^{d} \sigma_i^2.$$

Can immediately get a sense of "how low-rank" a matrix is from it's spectrum:



Suffices to compute right singular vectors $\mathbf{V}$:

$$(U\Sigma V^T)^T U \Sigma V^T = V^T \Sigma \cancel{U^T U} \Sigma V^T$$

$$= V^T \Sigma^2 V^T$$

- Compute $\mathbf{X}^T\mathbf{X}$.
- Find eigendecomposition $\mathbf{V}\mathbf{\Lambda}\mathbf{V}^T = \mathbf{X}^T\mathbf{X}$.
- Compute $\mathbf{L} = \mathbf{X}\mathbf{V}$. Set $\sigma_i = \|\mathbf{L}_i\|_2$ and $\mathbf{U}_i = \mathbf{L}_i / \|\mathbf{L}_i\|_2$.

$$XV = U\Sigma V^T V = U\Sigma \cdot \Sigma^{-1} = U$$

Total runtime $\approx$ $O(nd^2)$

$+ \; O(d^3)$

$O(nd \cdot (\text{something small}))$

- Compute approximate solution.
- Only compute top k singular vectors/values. Runtime will depend on k. When $k = d$ we can't do any better than classical algorithms based on eigendecomposition.
- Iterative algorithms achieve runtime $\approx O(ndk)$ vs. $O(nd^2)$ time.
  - **Krylov subspace methods** like the Lanczos method are most commonly used in practice.
  - **Power method** is the simplest Krylov subspace method, and still works very well.

$k = 1$

What we won't discuss today: sketching methods and stochastic methods (which are faster in some settings).

Today: What about when $k = 1$?

Goal: Find some $\mathbf{z} \approx \mathbf{v}_1$. $\|z - v_1\|_2 \leq \varepsilon$

Input: $\mathbf{X} \in \mathbb{R}^{n \times d}$ with SVD $\mathbf{U\Sigma V}^T$.

t - 1

Power method:

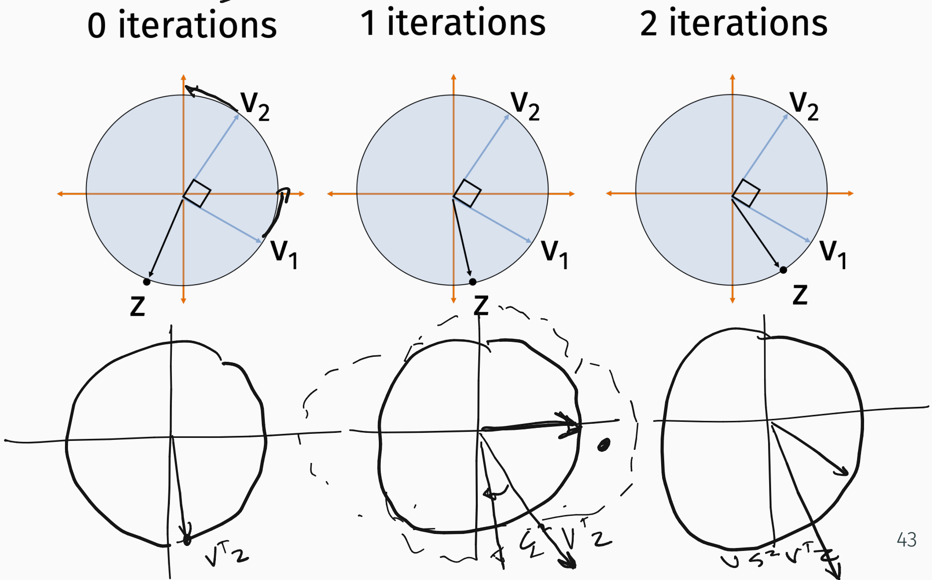
- Choose $\mathbf{z}^{(0)}$ randomly. E.g. $\mathbf{z}_0 \sim \mathcal{N}(0, 1)$.
- $\mathbf{z}^{(0)} = \mathbf{z}^{(0)} / \|\mathbf{z}^{(0)}\|_2$
- For $i = 1, \ldots, T$
    - $\mathbf{z}^{(i)} = \mathbf{X}^T \cdot (\mathbf{X}\mathbf{z}^{(i-1)})$ $\quad z^{(i)} = \underbrace{X^T X}\, z^{(i-1)}$
    - $n_i = \|\mathbf{z}^{(i)}\|_2$
    - $\mathbf{z}^{(i)} = \mathbf{z}^{(i)} / n_i$

  Return $\mathbf{z}^{(T)}$

$V \Sigma^2 V^T$

$X^T X z \quad V \Sigma^2 V^T z$

$\sigma^2 = .001$



Theorem (Basic Power Method Convergence)

Let $\gamma = \frac{\sigma_1 - \sigma_2}{\sigma_1}$ be parameter capturing the "gap" between the first and second largest singular values. If Power Method is initialized with a random Gaussian vector then, with high probability, after $T = O\left(\frac{\log d/\epsilon}{\gamma}\right)$ steps, we have either:

$$\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2 \leq \epsilon \qquad \text{or} \qquad \|\mathbf{v}_1 - (-\mathbf{z}^{(T)})\|_2 \leq \epsilon.$$

Total runtime: $O\left(nd \cdot \frac{\log d/\epsilon}{\gamma}\right)$

$1 - \frac{\sigma_2}{\sigma_1}$

Write $\mathbf{z}^{(i)}$ in the right singular vector basis:

$$\begin{aligned}
\mathbf{z}^{(0)} &= c_1^{(0)}\mathbf{v}_1 + c_2^{(0)}\mathbf{v}_2 + \ldots + c_d^{(0)}\mathbf{v}_d \\
\mathbf{z}^{(1)} &= c_1^{(1)}\mathbf{v}_1 + c_2^{(1)}\mathbf{v}_2 + \ldots + c_d^{(1)}\mathbf{v}_d \\
&\vdots \\
\mathbf{z}^{(i)} &= c_1^{(i)}\mathbf{v}_1 + c_2^{(i)}\mathbf{v}_2 + \ldots + c_d^{(i)}\mathbf{v}_d
\end{aligned}$$

Note: $[c_1^{(i)}, \ldots, c_d^{(i)}] = \mathbf{c}^{(i)} = \mathbf{V}^T\mathbf{z}^{(i)}$.

Also: $\sum_{j=1}^{d} \left(c_j^{(i)}\right)^2 = 1$.

Claim: After update $\mathbf{z}^{(i)} = \mathbf{X}^T\mathbf{X}\mathbf{z}^{(i-1)}$,

$$c_j^{(i)} = \sigma_j^2 c_j^{(i-1)}$$

$(U\Sigma V^T)^T U\Sigma V^T$

$= V \Sigma U^T U \Sigma V^T$

$= V \Sigma^2 V^T$

$$\mathbf{z}^{(i)} = \frac{1}{n_i}\left[c_1^{(i-1)}\sigma_1^2 \cdot \mathbf{v}_1 + c_2^{(i-1)}\sigma_2^2 \cdot \mathbf{v}_2 + \ldots + c_d^{(i-1)}\sigma_d^2 \cdot \mathbf{v}_d\right]$$

$V \Sigma^2 V^T z^{(i-1)}$

$[c_1^{(i-1)} \ldots c_d^{(i-1)}]$

$V [\sigma_1^2 c_1^{(i-1)} \ldots \sigma_d^2 c_d^{(i-1)}]$

Claim: After T updates:

$$\mathbf{z}^{(T)} = \frac{1}{\prod_{i=1}^{T} n_i}\left[c_1^{(0)}\sigma_1^{2T}\cdot \mathbf{v}_1 + c_2^{(0)}\sigma_2^{2T}\cdot \mathbf{v}_2 + \ldots + c_d^{(0)}\sigma_d^{2T}\cdot \mathbf{v}_d\right]$$

$\|z^{(T)}\|_2^2$

$\sum_{j=1}^{d} \alpha_j^2 = 1$

$z^{(T)}$

v_j

Let $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)}\sigma_j^{2T}$. **Goal:** Show that $\alpha_j \ll \alpha_1$ for all $j \neq 1$.

Since $\mathbf{z}^{(T)}$ is a unit vector, $\sum_{i=1}^{d} \alpha_i^2 = 1$. So $\alpha_1 \leq 1$.

If we can prove that $\frac{\alpha_j}{\alpha_1} \leq \sqrt{\frac{\epsilon}{d}}$ then:

$\alpha_j \leq \sqrt{\frac{\epsilon}{d}}$

$$\alpha_1^2 \geq 1 - d \cdot \left(\sqrt{\frac{\epsilon}{d}}\right)^2 \implies |\alpha_1| \geq 1 - \epsilon$$

$\alpha_1^2 = 1 - \sum_{j=2}^{d} \alpha_j^2 \geq 1 - d \cdot \left(\sqrt{\frac{\epsilon}{d}}\right)^2$

$$\|\mathbf{v}_1 - \mathbf{z}^{(T)}\|_2 = 2 - 2\langle \mathbf{v}_1, \mathbf{z}^{(T)}\rangle \leq 2\epsilon$$

$\alpha_1 \approx 1-\epsilon$

$2 - 2(1-\epsilon) \leq 2\epsilon$

Lets proves that $\frac{\alpha_j}{\alpha_1} \leq \sqrt{\frac{\epsilon}{d}}$ where $\alpha_j = \frac{1}{\prod_{i=1}^{T} n_i} c_j^{(0)} \sigma_j^{2T}$

First observation: Starting coefficients are all roughly equal.

For all j $\qquad O(1/d^3) \leq c_j^{(0)} \leq 1$

with probability $1 - \frac{1}{d}$. This is a very loose bound, but it's all that we will need. **Prove at home.**

$(1-x)^{1/x} \approx \frac{1}{e}$

$$\frac{\alpha_j}{\alpha_1} = \frac{\sigma_j^{2T}}{\sigma_1^{2T}} \frac{c_j^{(0)}}{c_1^{(0)}} \leq \frac{\sigma_j^{2T}}{\sigma_1^{2T}} \cdot O(d^3) \leq \sqrt{\frac{\epsilon}{d}}$$

with high probability

$$\left(\frac{\sigma_j}{\sigma_1}\right)^{2T} \leq \frac{\sqrt{\epsilon}}{d^{3.5}}$$

$$\left(1-\left(1-\frac{\sigma_j}{\sigma_1}\right)\right)^{\frac{1}{\left(1-\frac{\sigma_j}{\sigma_1}\right)} \cdot \log\left(d^{3.5}/\sqrt{\epsilon}\right)}$$

Need $T = O\left(\frac{1}{1-\frac{\sigma_j}{\sigma_1}} \cdot \log(d/\epsilon)\right)$

Theorem (Gapless Power Method Convergence)

If Power Method is initialized with a random Gaussian vector then, with high probability, after $T = O\left(\frac{\log d/\epsilon}{\epsilon}\right)$ steps, we obtain a $\mathbf{z}$ *satisfying:*

$$\|\mathbf{X} - \mathbf{X}\mathbf{z}\mathbf{z}^T\|_F^2 \leq (1+\epsilon)\|\mathbf{X} - \mathbf{X}\mathbf{v}_1\mathbf{v}_1^T\|_F^2$$

- Block Power Method aka Simultaneous Iteration aka Subspace Iteration aka Orthogonal Iteration

Power method:

- Choose $\mathbf{G} \in \mathbb{R}^{d \times k}$ be a random Gaussian matrix.
- $\mathbf{Z}_0 = \text{orth}(\mathbf{G})$.
- For $i = 1, \ldots, T$
  - $\mathbf{Z}^{(i)} = \mathbf{X}^T \cdot (\mathbf{X}\mathbf{Z}^{(i-1)})$
  - $\mathbf{Z}^{(i)} = \text{orth}(\mathbf{Z}^{(i)})$

  Return $\mathbf{Z}^{(T)}$

Runtime: $O\left(\frac{\log d/\epsilon}{\epsilon}\right)$ iterations to obtain a nearly optimal low-rank approximation:

$$\|\mathbf{X} - \mathbf{X}\mathbf{Z}\mathbf{Z}^T\|_F^2 \leq (1+\epsilon)\|\mathbf{X} - \mathbf{X}\mathbf{V_k}\mathbf{V_k}^T\|_F^2.$$