

# CS 383 – Computational Text Analysis

## Lecture 4

### Document Representation, Matrix Factorization

Adam Poliak

01/30/2023

Slides adapted from Dan Jurafsky, Dirk Hovy, Jure Leskovec

# Announcements (1/2)

- Office Hours:
  - This week: Thursday 3:30-4:30pm
- HW01 due tonight Monday 01/30
  - Based on Monday's lecture
- HW02 released tonight, due Monday 02/06

# Outline

- Document Representations - recap
- tf-idf
- Linear Algebra:
  - Matrix multiplication
  - Matrix factorization
    - SVD
- Latent Semantic Analysis
- PCA

# Recap so far

The first class was all about counting words

2<sup>nd</sup> class was about the power of counting words.

By counting words we can \_\_\_\_ \_\_\_\_

learn about language

generate language

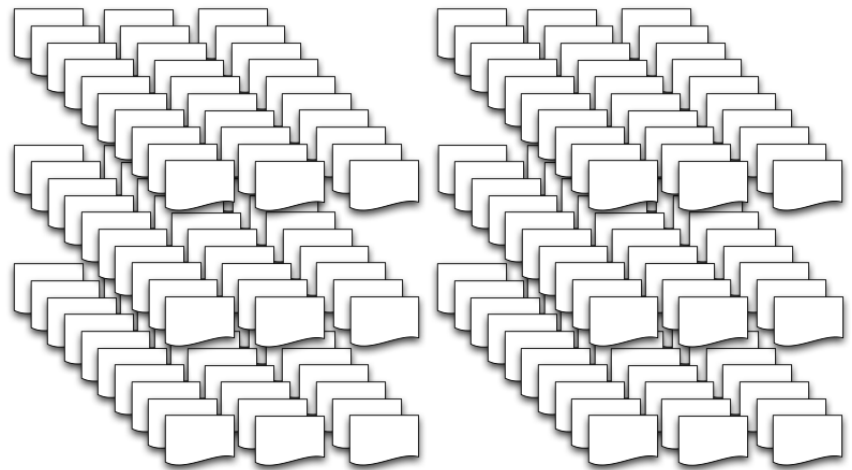
categorize language

represent documents as vectors

# Terminology

- **Corpus:**

- A collection of documents
- *Corpora* – plural of corpus

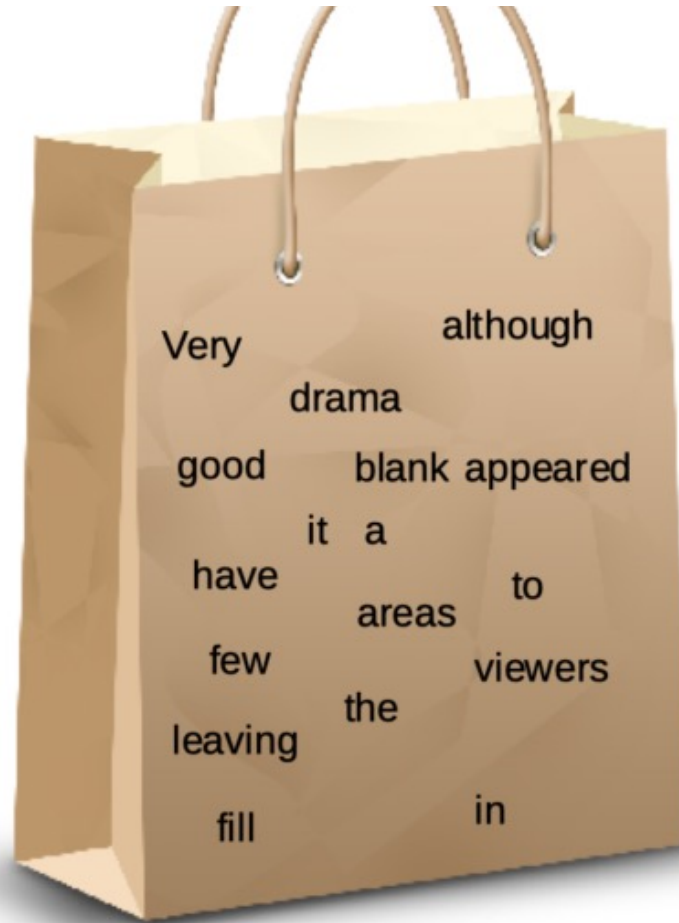


- **Document:**

- Often unit of text of interest (dependent on RQ)
- Often represents one data point

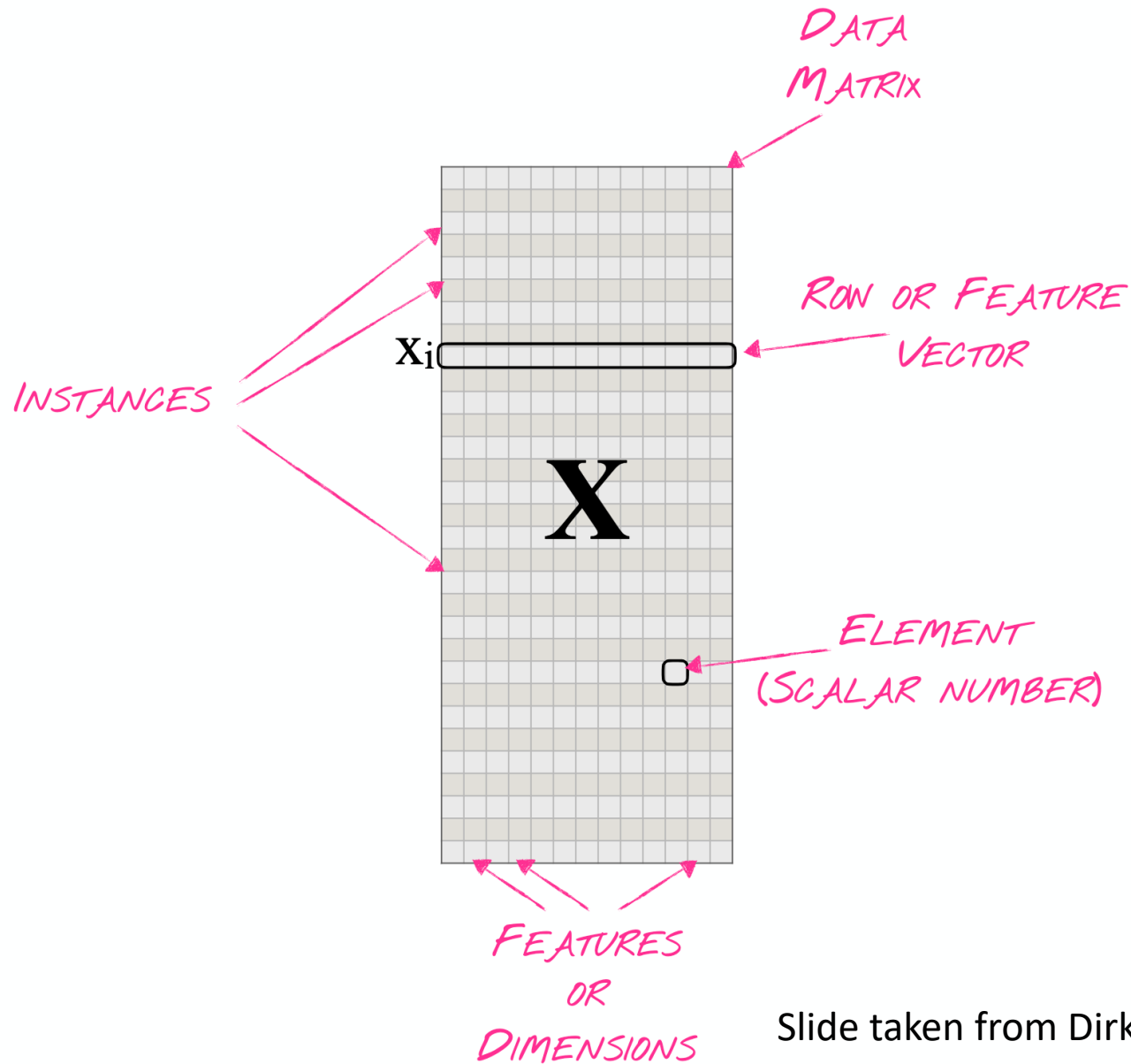
# Bag of Words

Very good drama although it appeared to have a few blank areas leaving the viewers to fill in the action for themselves. I can imagine life being this way for someone who can neither read nor write. This film simply smacked of the real world: the wife who is suddenly the sole supporter, the live-in relatives and their quarrels, the troubled child who gets knocked up and then, typically, drops out of school, a jackass husband who takes the nest egg and buys beer with it. 2 thumbs up... very very very good movie.



('the', 8),  
(',', 5),  
('very', 4),  
('.', 4),  
('who', 4),  
('and', 3),  
('good', 2),  
('it', 2),  
('to', 2),  
('a', 2),  
('for', 2),  
('can', 2),  
('this', 2),  
('of', 2),  
('drama', 1),  
('although', 1),  
('appeared', 1),  
('have', 1),  
('few', 1),  
('blank', 1)  
.....

# Document Matrix



Slide taken from Dirk Hovy

# Outline

- Document Representations - recap
- tf-idf
- Linear Algebra:
  - Matrix multiplication
  - Matrix factorization
    - SVD
- Latent Semantic Analysis
- PCA



# What to count?

# How to count?

Next lecture

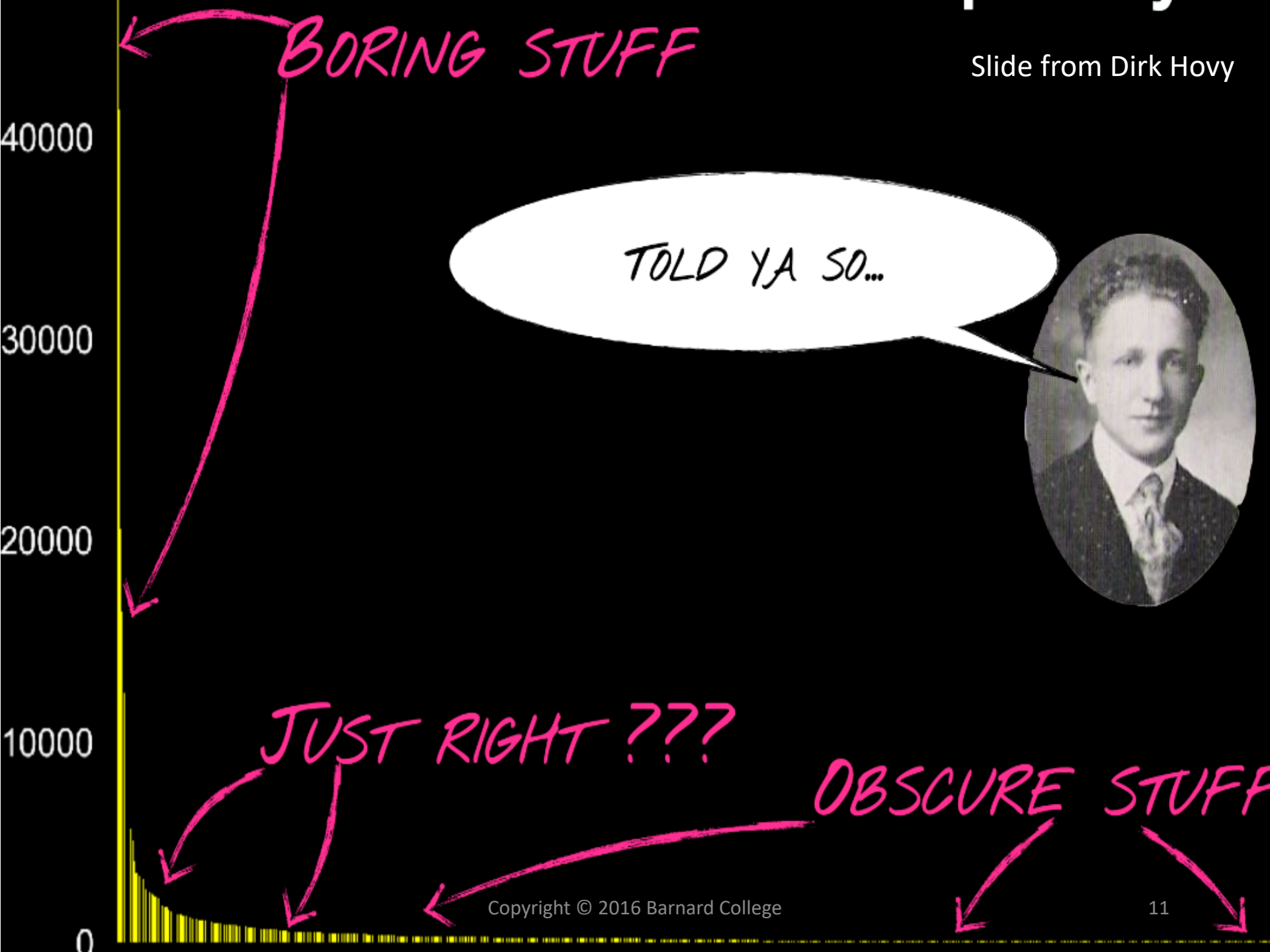
HW02

# Term Frequency (tf):

**tf** of word **w** in document **d**:

$$\frac{|w|}{|\text{Document}|}$$

*number of times **w** appears in **D**  
divided by of number tokens in **D***



# **Inverse Document Frequency**

# Some words are more interesting

The image consists of eight panels arranged in a 2x4 grid, separated by thick black vertical bars. Each panel contains several horizontal gray lines representing text. The word 'the' is placed on these lines in various positions and frequencies. The word 'sustainable' is highlighted in red in two panels, indicating it is more interesting despite its lower frequency.

Row	Column	Frequency of 'the'	Frequency of 'sustainable'
1	1	4	0
1	2	2	0
1	3	2	0
1	4	1	1
2	1	2	0
2	2	1	1
2	3	2	0
2	4	3	0

# Inverse Document Frequency (idf)

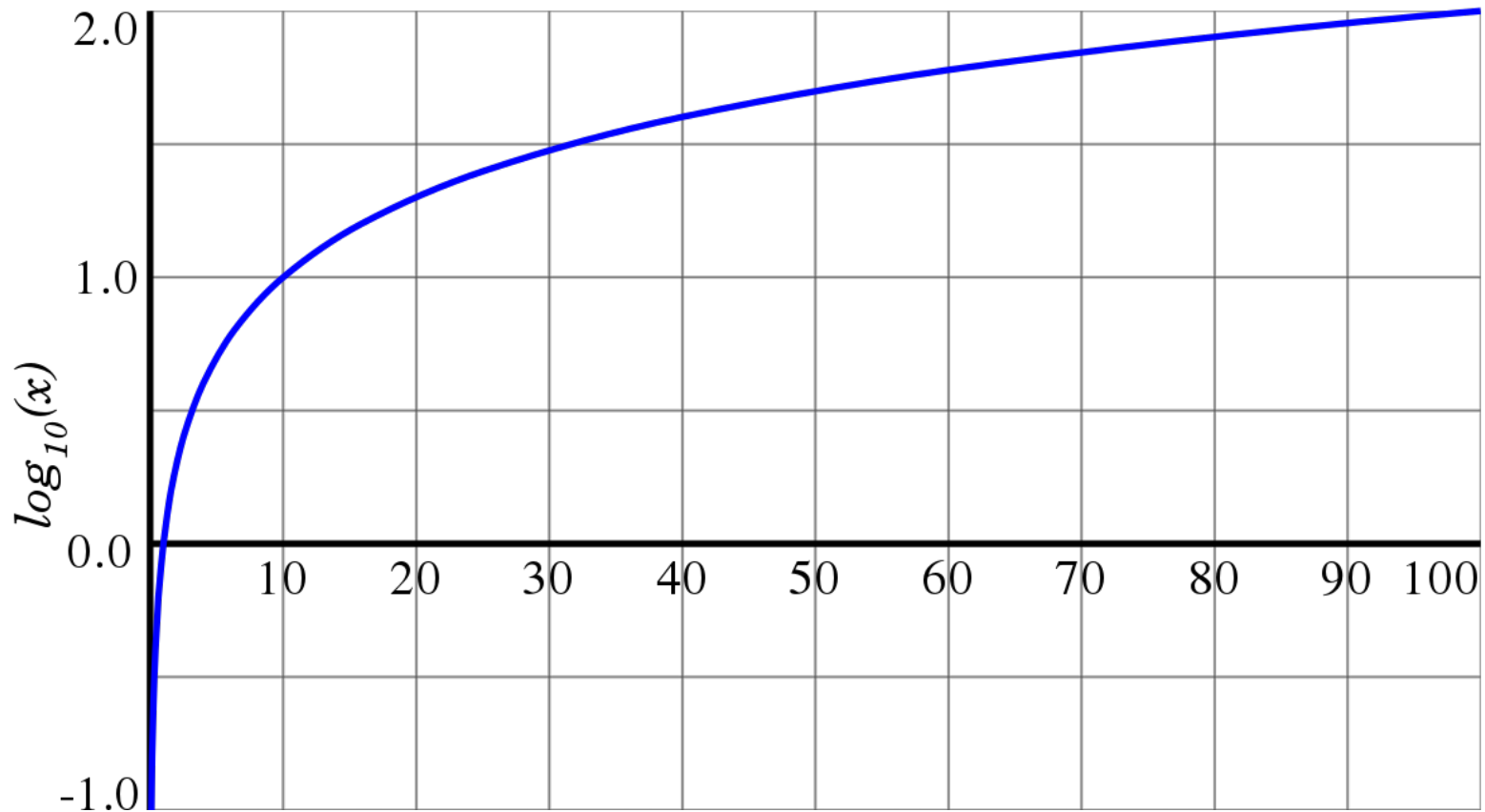
*idf* of word **w** in document **D**:

$$\log \frac{|D|}{|tf(w,d) \neq 0|}$$

*number of documents divided  
by number of documents that  
contain **w***

# Scaling down IDF

log function is a way to scale down idf



# TF-IDF



TF-IDF:

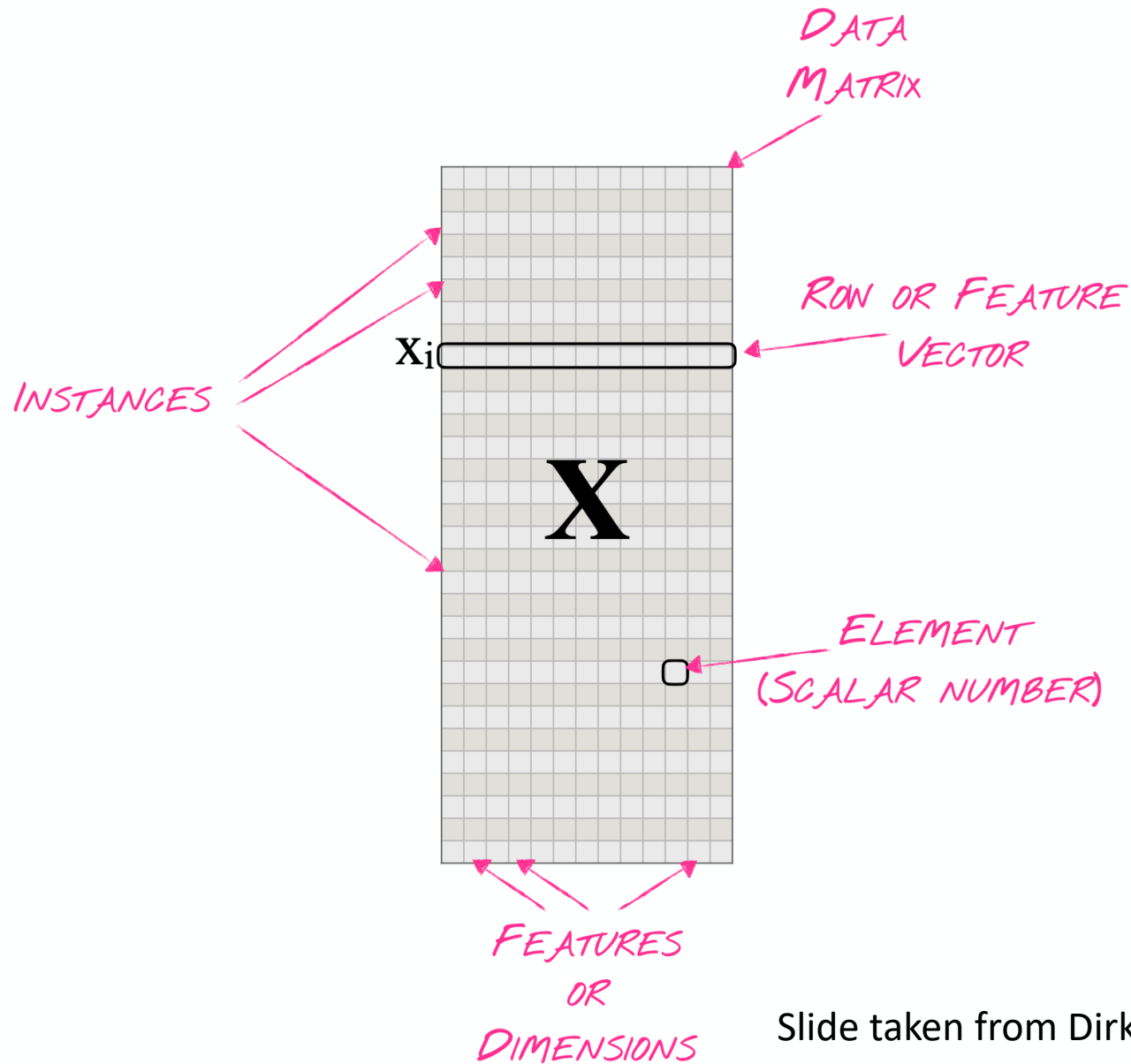
Term Frequency - Inverse Document  
Frequency

***TF-IDF*** of word ***w*** in document ***D***:

Term Frequency \* Inverse Document Frequency

Captures terms that are frequent in a document and specific to the document in the corpus

# Document Matrix



Slide taken from Dirk Hovy

# Document Matrix Characteristics

- Sparse:
  - Most common value is 0
- Size:
  - $|D| \times |V|$
  - Vocab can be very large, especially if we use n-grams
- Values are positive

# Outline

- Document Representations - recap
- tf-idf
- Dimensionality Reduction
  - Linear Algebra Review:
    - Matrix multiplication
    - Matrix factorization
      - SVD
- Latent Semantic Analysis
- PCA

# Matrix Multiplication

We can multiply two matrices A and B if ....

number of columns in A = number of rows in B

The size of the resulting matrix is ....

number of rows in A & the number of columns in B

[Khan Academy](#)

# Matrix Multiplication

$$\begin{bmatrix} 1 & 7 \\ 2 & 4 \end{bmatrix} \cdot \begin{bmatrix} 3 & 3 \\ 5 & 2 \end{bmatrix}$$

*A*

*B*

# Matrix Multiplication

$$\begin{array}{c} \vec{a}_1 \rightarrow \\ \vec{a}_2 \rightarrow \end{array} \begin{array}{c} \vec{b}_1 \\ \vec{b}_2 \\ \downarrow \\ \downarrow \end{array} \begin{array}{c} \left[ \begin{array}{cc} 1 & 7 \\ 2 & 4 \end{array} \right] \cdot \left[ \begin{array}{cc} 3 & 3 \\ 5 & 2 \end{array} \right] \\ \\ A \quad B \end{array}$$

# Matrix Multiplication

$$\begin{array}{c} \vec{a}_1 \rightarrow \\ \vec{a}_2 \rightarrow \end{array} \begin{array}{c} A \\ \left[ \begin{array}{cc} 1 & 7 \\ 2 & 4 \end{array} \right] \end{array} \cdot \begin{array}{c} \vec{b}_1 \quad \vec{b}_2 \\ \downarrow \quad \downarrow \\ B \\ \left[ \begin{array}{cc} 3 & 3 \\ 5 & 2 \end{array} \right] \end{array} = \begin{array}{c} C \\ \left[ \begin{array}{cc} \vec{a}_1 \cdot \vec{b}_1 & \vec{a}_1 \cdot \vec{b}_2 \\ \vec{a}_2 \cdot \vec{b}_1 & \vec{a}_2 \cdot \vec{b}_2 \end{array} \right] \end{array}$$



# Rank of a Matrix

- **Q:** What is **rank** of a matrix **A**?
- **A:** Number of **linearly independent** columns of **A**
- **For example:**
  - Matrix **A** =  $\begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix}$  has rank **r=2**
    - **Why?** The first two rows are linearly independent, so the rank is at least 2, but all three rows are linearly dependent (the first is equal to the sum of the second and third) so the rank must be less than 3.
- **Why do we care about low rank?**
  - We can write **A** as two “basis” vectors:  $[1 \ 2 \ 1] \ [-2 \ -3 \ 1]$
  - And new coordinates of :  $[1 \ 0] \ [0 \ 1] \ [1 \ -1]$

# Rank is “Dimensionality”

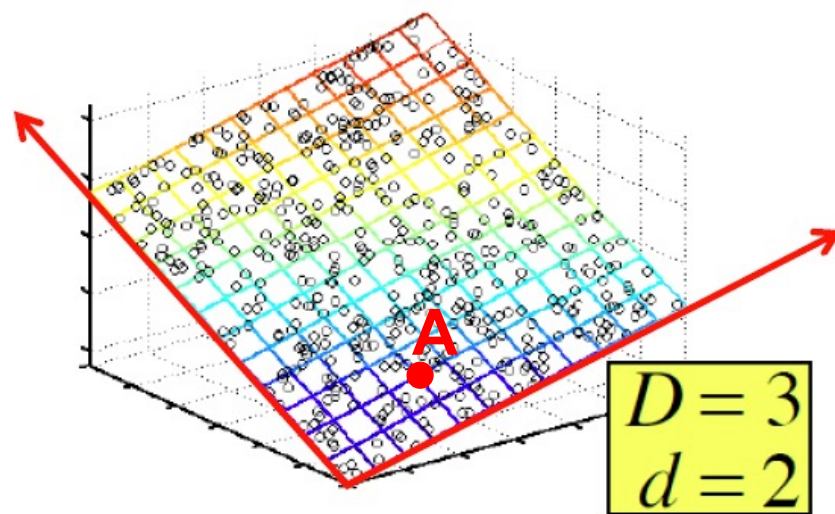
- **Cloud of points 3D space:**

- Think of point positions

as a matrix:

1 row per point:

$$\begin{bmatrix} 1 & 2 & 1 \\ -2 & -3 & 1 \\ 3 & 5 & 0 \end{bmatrix} \begin{matrix} \mathbf{A} \\ \mathbf{B} \\ \mathbf{C} \end{matrix}$$



- **We can rewrite coordinates more efficiently!**

- Old basis vectors:  $[1 \ 0 \ 0]$   $[0 \ 1 \ 0]$   $[0 \ 0 \ 1]$

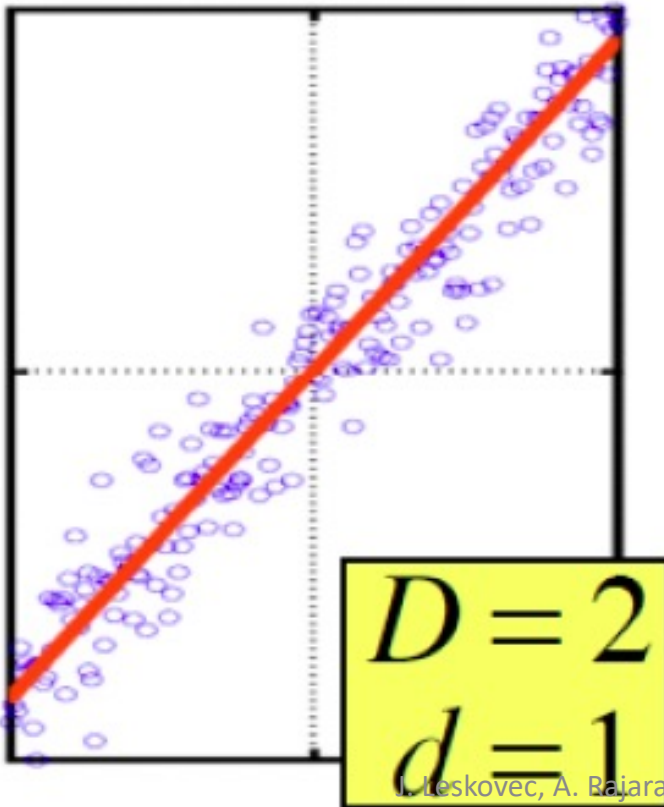
- **New basis vectors:**  $[1 \ 2 \ 1]$   $[-2 \ -3 \ 1]$

- Then **A** has new coordinates:  $[1 \ 0]$ . **B**:  $[0 \ 1]$ , **C**:  $[1 \ 1]$

- **Notice: We reduced the number of coordinates!**

# Dimensionality Reduction

- **Goal of dimensionality reduction is to discover the axis of data!**

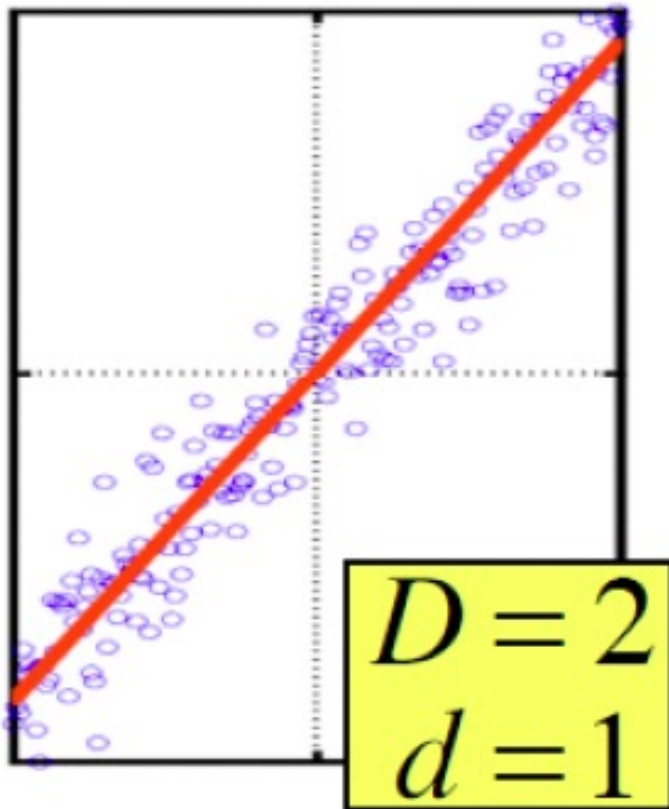


Rather than representing every point with 2 coordinates we represent each point with 1 coordinate (corresponding to the position of the point on the red line).

By doing this we incur a bit of **error** as the points do not exactly lie on the line

# Dimensionality Reduction in NLP

- **Goal of dimensionality reduction is to discover axes of data!**



Rather than representing every point with  $|V|$  coordinates, we represent each point with  $k$  coordinates (corresponding to the position of the point on the red line).

What are the  $k$  coordinates?

# Why Reduce Dimensions?

## Discover hidden correlations/topics

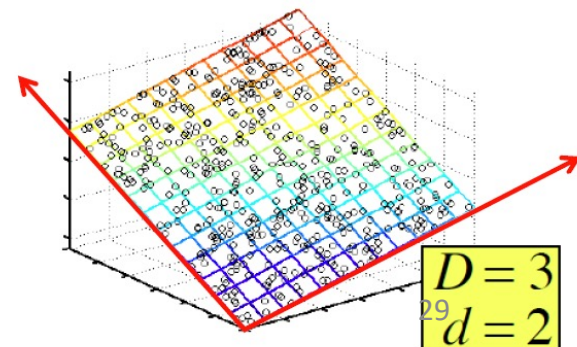
- Words that occur commonly together

## Remove redundant and noisy features

- Not all words are useful

## Interpretation and visualization

## Easier storage and processing of the data



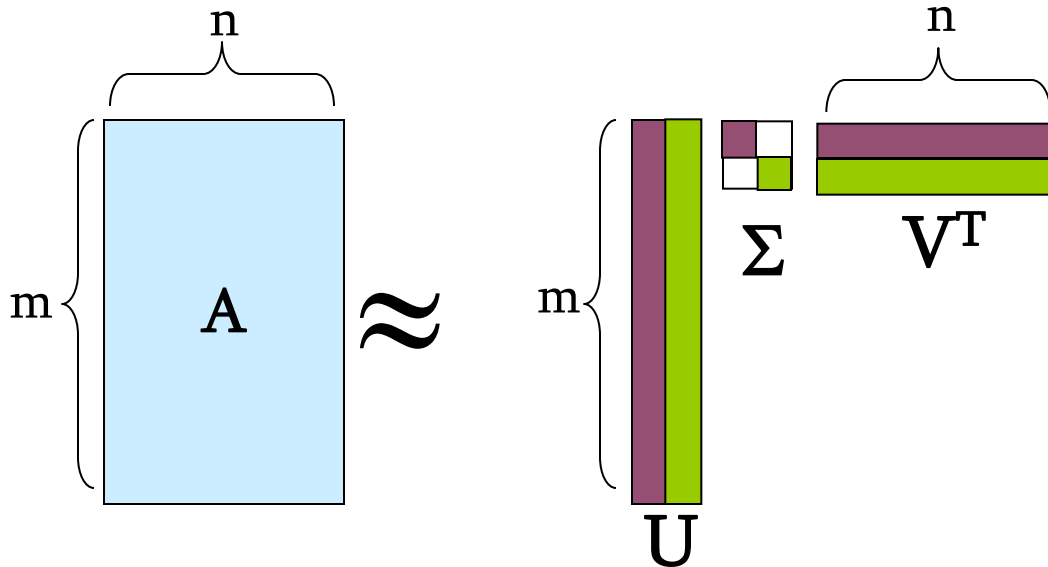
# SVD - Definition

$$\mathbf{A}_{[m \times n]} = \mathbf{U}_{[m \times r]} \mathbf{\Sigma}_{[r \times r]} (\mathbf{V}_{[n \times r]})^T$$

- **A: Input data matrix**
  - $m \times n$  matrix (e.g.,  $m$  documents,  $n$  terms)
- **U: Left singular vectors**
  - $m \times k$  matrix ( $m$  documents,  $r$  concepts)  $r \ll n$
- **$\Sigma$ : Singular values**
  - $r \times r$  diagonal matrix (strength of each 'concept')  
( $r$  : rank of the matrix **A**)
- **V: Right singular vectors**
  - $n \times r$  matrix ( $n$  terms,  $r$  concepts)

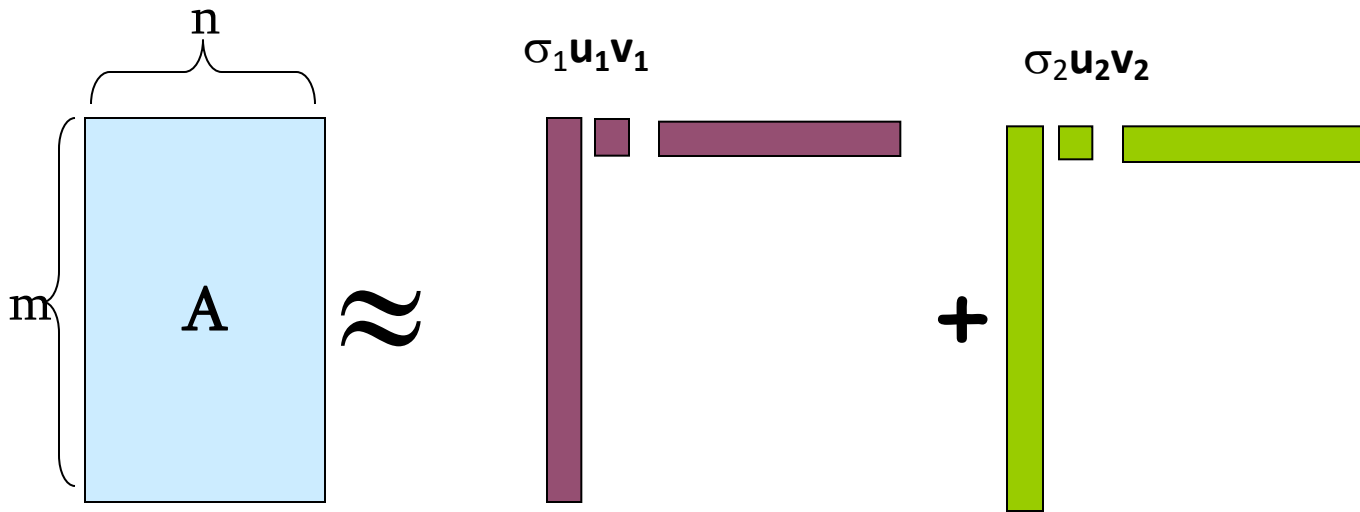
# SVD

$$\mathbf{A} \approx \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_i \sigma_i \mathbf{u}_i \circ \mathbf{v}_i^T$$



# SVD

$$\mathbf{A} \approx \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \sum_i \sigma_i \mathbf{u}_i \circ \mathbf{v}_i^T$$



$\sigma_i$  ... scalar  
 $\mathbf{u}_i$  ... vector  
 $\mathbf{v}_i$  ... vector



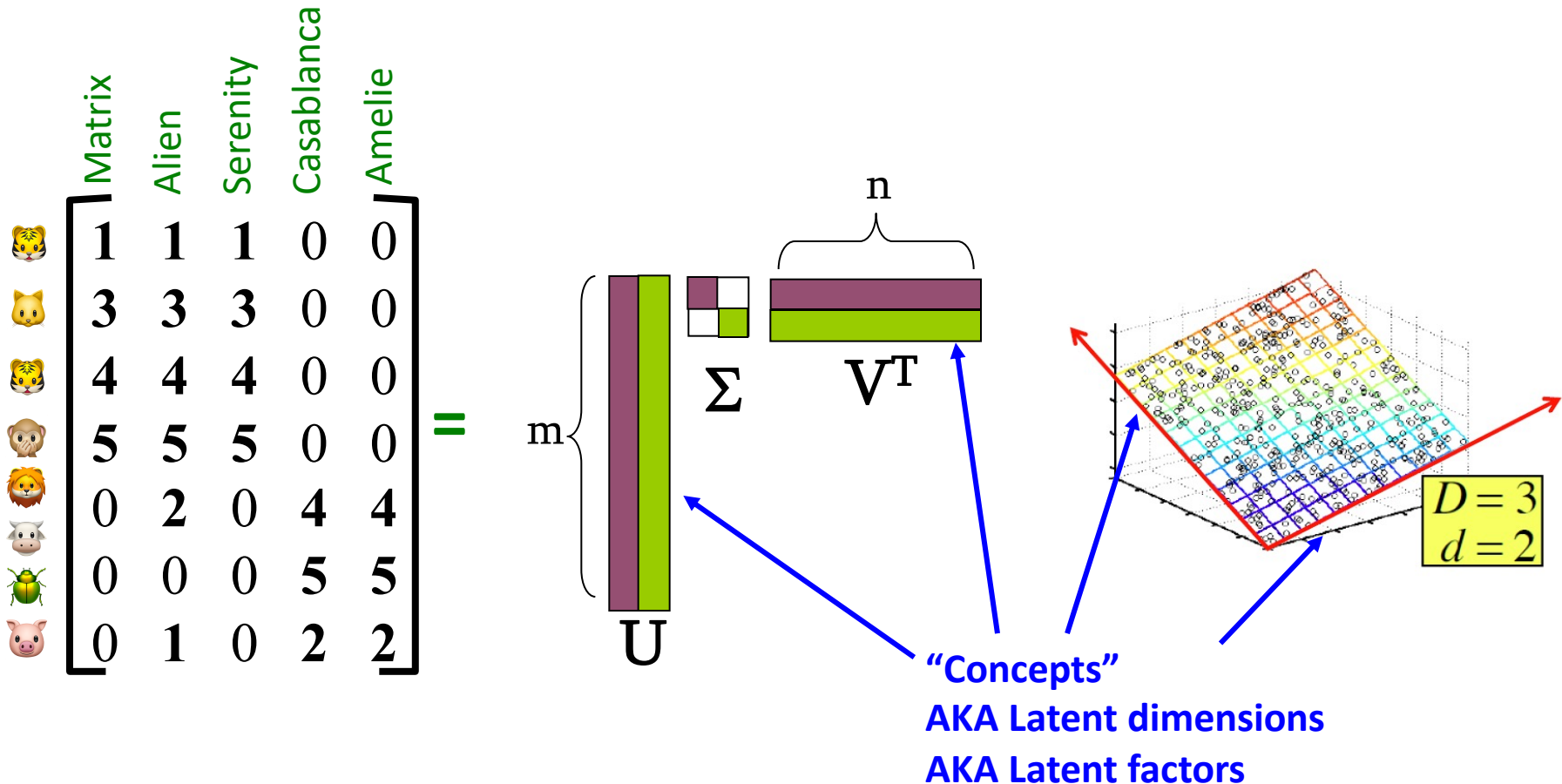
# SVD - Properties

It is **always** possible to decompose a real matrix  $A$  into  $A = U \Sigma V^T$ , where

- $U, \Sigma, V$ : **unique**
- $U, V$ : **column orthonormal**
  - $U^T U = I; V^T V = I$  ( $I$ : identity matrix)
  - (Columns are orthogonal unit vectors)
- $\Sigma$ : **diagonal**
  - Entries (**singular values**) are **positive**, and sorted in decreasing order ( $\sigma_1 \geq \sigma_2 \geq \dots \geq 0$ )

# SVD – Example: Users-to-Movies

- $A = U \Sigma V^T$  - example: **Users to Movies**



# SVD – Example: Users-to-Movies

- $A = U \Sigma V^T$  - example: Users to Movies

	Matrix	Alien	Serenity	Casablanca	Amelie		U		$\Sigma$		$V^T$				
🐅	1	1	1	0	0	=	0.13	0.02	-0.01	x	12.4	0	0	x	
🐱	3	3	3	0	0		0.41	0.07	-0.03		0	9.5	0		
🐅	4	4	4	0	0		0.55	0.09	-0.04		0	0	1.3		
🐵	5	5	5	0	0		0.68	0.11	-0.05						
🦁	0	2	0	4	4		0.15	-0.59	0.65						
🐮	0	0	0	5	5		0.07	-0.73	-0.67						
🐛	0	0	0	5	5		0.07	-0.73	-0.67						
🐷	0	1	0	2	2		0.07	-0.29	0.32						
											0.56	0.59	0.56	0.09	0.09
											0.12	-0.02	0.12	-0.69	-0.69
											0.40	-0.80	0.40	0.09	0.09

# SVD – Example: Users-to-Movies

- $A = U \Sigma V^T$  - example: Users to Movies

Matrix	Alien	Serenity	Casablanca	Amelie		SciFi-concept		Romance-concept						
1	1	1	0	0	=	0.13	0.02	-0.01		12.4	0	0		
3	3	3	0	0		0.41	0.07	-0.03		0	9.5	0		
4	4	4	0	0		0.55	0.09	-0.04	x	0	0	1.3		
5	5	5	0	0		0.68	0.11	-0.05		0	0	0		
0	2	0	4	4		0.15	-0.59	0.65		0	0	0		
0	0	0	5	5		0.07	-0.73	-0.67		0	0	0		
0	1	0	2	2		0.07	-0.29	0.32		0	0	0		
										0.56	0.59	0.56	0.09	0.09
										0.12	-0.02	0.12	-0.69	-0.69
										0.40	-0.80	0.40	0.09	0.09

# SVD – Example: Users-to-Movies

- $A = U \Sigma V^T$  - example:  $U$  is “user-to-concept” similarity matrix

Matrix	Alien	Serenity	Casablanca	Amelie		SciFi-concept	Romance-concept																											
1	1	1	0	0	=	0.13	0.02	-0.01	$\times$ <table border="1" style="margin: 0 auto;"> <tr><td>12.4</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>9.5</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1.3</td></tr> </table> $\times$	12.4	0	0	0	9.5	0	0	0	1.3	<table border="1" style="margin: 0 auto;"> <tr><td>0.56</td><td>0.59</td><td>0.56</td><td>0.09</td><td>0.09</td></tr> <tr><td>0.12</td><td>-0.02</td><td>0.12</td><td>-0.69</td><td>-0.69</td></tr> <tr><td>0.40</td><td>-0.80</td><td>0.40</td><td>0.09</td><td>0.09</td></tr> </table>	0.56	0.59	0.56	0.09	0.09	0.12	-0.02	0.12	-0.69	-0.69	0.40	-0.80	0.40	0.09	0.09
12.4	0	0																																
0	9.5	0																																
0	0	1.3																																
0.56	0.59	0.56	0.09	0.09																														
0.12	-0.02	0.12	-0.69	-0.69																														
0.40	-0.80	0.40	0.09	0.09																														
3	3	3	0	0	0.41	0.07	-0.03																											
4	4	4	0	0	0.55	0.09	-0.04																											
5	5	5	0	0	0.68	0.11	-0.05																											
0	2	0	4	4	0.15	-0.59	0.65																											
0	0	0	5	5	0.07	-0.73	-0.67																											
0	1	0	2	2	0.07	-0.29	0.32																											

# SVD – Example: Users-to-Movies

- $A = U \Sigma V^T$  - example:

Matrix	Alien	Serenity	Casablanca	Amelie		SciFi-concept		“strength” of the SciFi-concept		
1	1	1	0	0	=	0.13	0.02	-0.01		
3	3	3	0	0		0.41	0.07	-0.03		
4	4	4	0	0		0.55	0.09	-0.04		
5	5	5	0	0		0.68	0.11	-0.05	x	
0	2	0	4	4		0.15	-0.59	0.65		
0	0	0	5	5		0.07	-0.73	-0.67		
0	1	0	2	2		0.07	-0.29	0.32		
								12.4	0	
								0	9.5	
								0	0	
									1.3	
									x	
						0.56	0.59	0.56	0.09	0.09
						0.12	-0.02	0.12	-0.69	-0.69
						0.40	-0.80	0.40	0.09	0.09

# SVD – Example: Users-to-Movies

- $A = U \Sigma V^T$  - example:

**V** is “movie-to-concept” similarity matrix

Matrix	Alien	Serenity	Casablanca	Amelie
1	1	1	0	0
3	3	3	0	0
4	4	4	0	0
5	5	5	0	0
0	2	0	4	4
0	0	0	5	5
0	1	0	2	2

SciFi-concept

0.13	0.02	-0.01
0.41	0.07	-0.03
0.55	0.09	-0.04
0.68	0.11	-0.05
0.15	-0.59	0.65
0.07	-0.73	-0.67
0.07	-0.29	0.32

$\times$

12.4	0	0
0	9.5	0
0	0	1.3

$\times$

0.56	0.59	0.56	0.09	0.09
0.12	-0.02	0.12	-0.69	-0.69
0.40	-0.80	0.40	0.09	0.09

SciFi-concept

J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, <http://www.mmds.org>

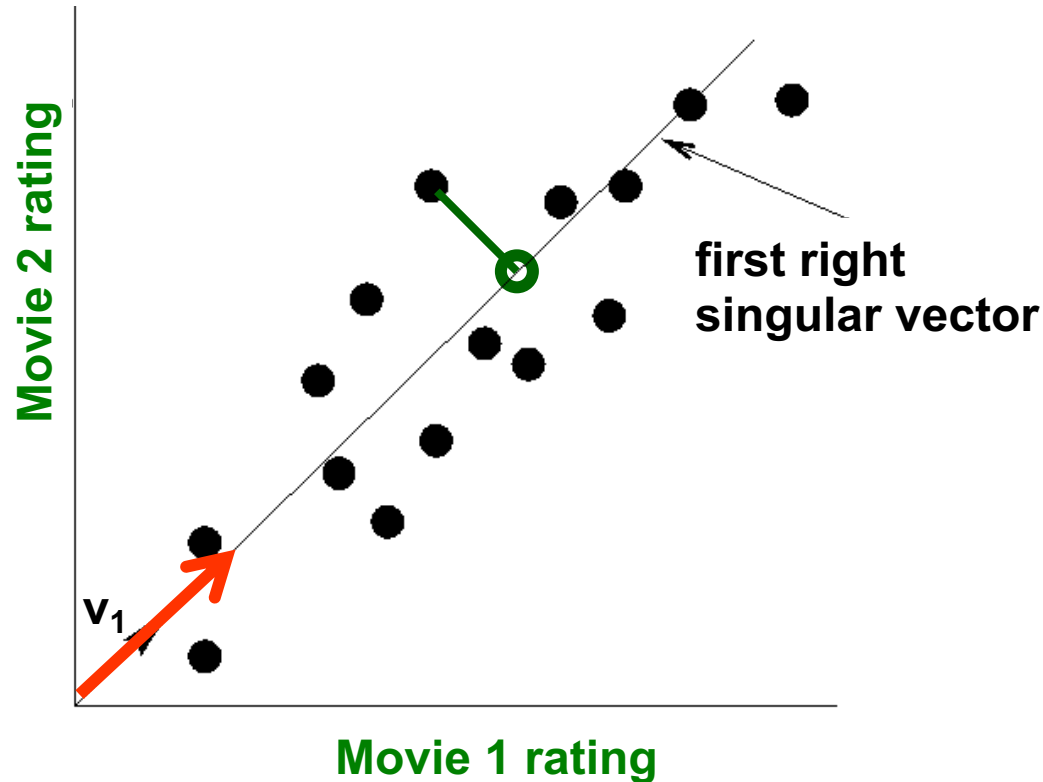
# SVD - Interpretation #1

**‘movies’, ‘users’ and ‘concepts’:**

- **$U$** : user-to-concept similarity matrix
- **$V$** : movie-to-concept similarity matrix
- **$\Sigma$** : its diagonal elements:  
‘strength’ of each concept



# SVD – Dimensionality Reduction



J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, <http://www.mmms.org>

- Instead of using two coordinates  $(x, y)$  to describe point locations, let's use only one coordinate  $(z)$
- Point's position is its location along vector  $v_1$
- **How to choose  $v_1$ ? Minimize reconstruction error**

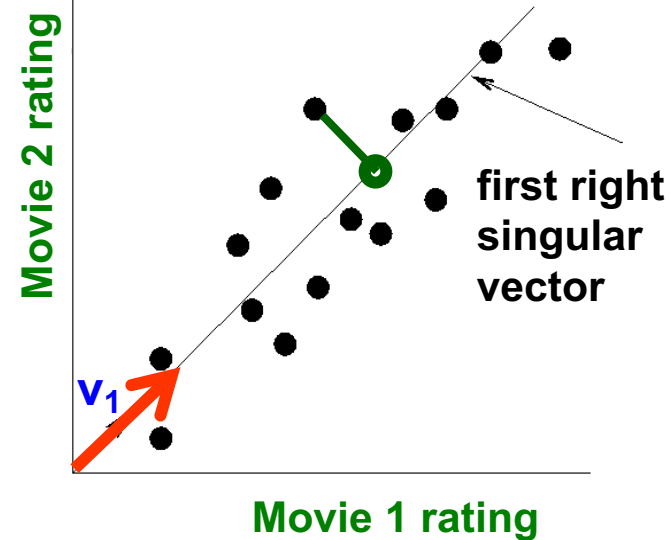
# SVD – Dimensionality Reduction

- **Goal:** Minimize the sum of reconstruction errors:

$$\sum_{i=1}^N \sum_{j=1}^D \|x_{ij} - z_{ij}\|^2$$

- where  $x_{ij}$  are the “old” and  $z_{ij}$  are the “new” coordinates

- **SVD gives ‘best’ axis to project on:**
  - ‘best’ = minimizing the reconstruction errors
- In other words, **minimum reconstruction error**



# SVD - Interpretation #2

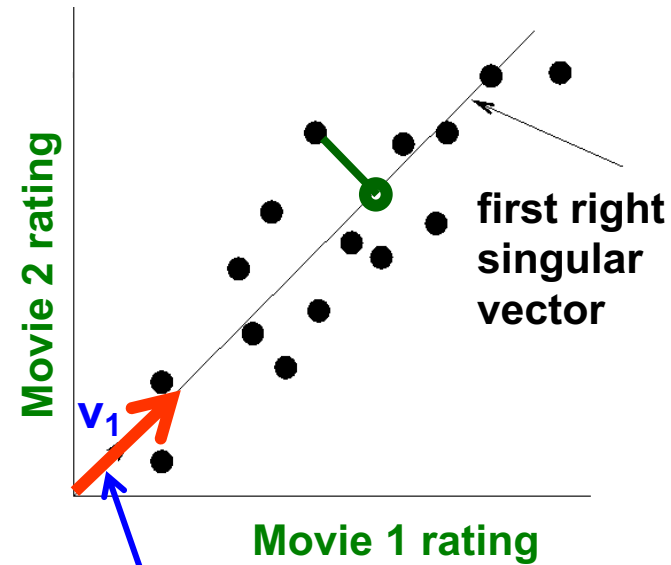
## • $A = U \Sigma V^T$ - example:

- $V$ : “movie-to-concept” matrix
- $U$ : “user-to-concept” matrix

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times$$

$$\begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times$$

$$\begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

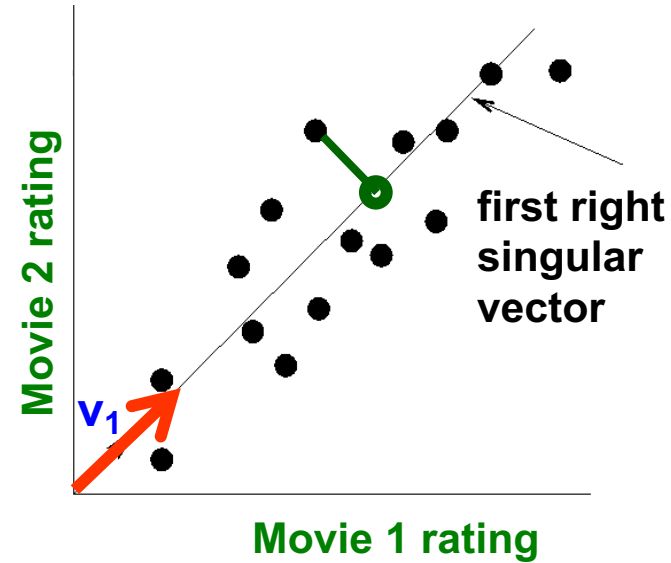


# SVD - Interpretation #2

- $A = U \Sigma V^T$  - example:

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

variance ('spread')  
on the  $v_1$  axis



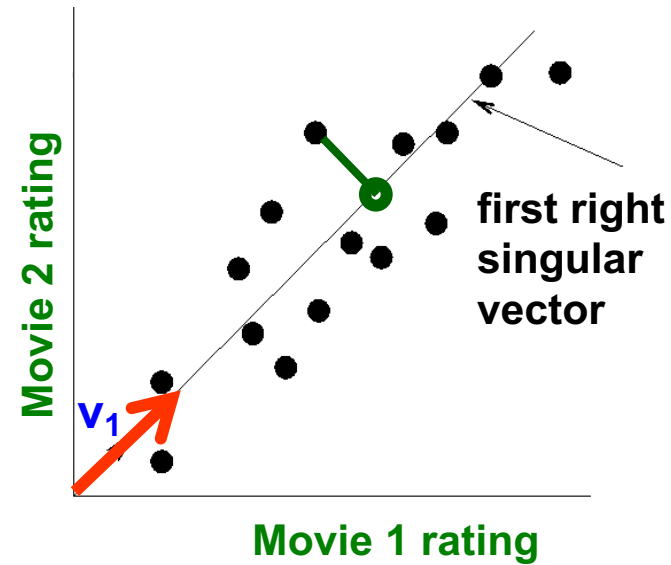
# SVD - Interpretation #2

## $A = U \Sigma V^T$ - example:

- $U \Sigma$ : Gives the coordinates of the points in the projection axis

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

Projection of users  
on the “Sci-Fi” axis  
 $(U \Sigma)^T$ :



$$\begin{bmatrix} 1.61 & 0.19 & -0.01 \\ 5.08 & 0.66 & -0.03 \\ 6.82 & 0.85 & -0.05 \\ 8.43 & 1.04 & -0.06 \\ 1.86 & -5.60 & 0.84 \\ 0.86 & -6.93 & -0.87 \\ 0.86 & -2.75 & 0.41 \end{bmatrix}$$

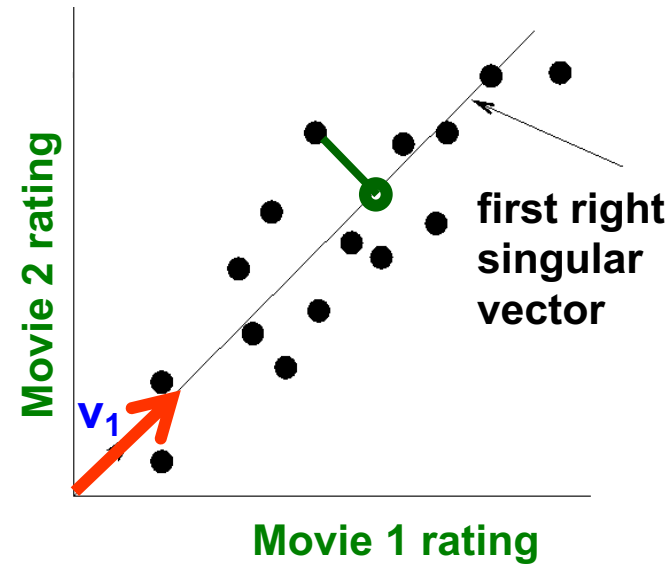
# SVD - Interpretation #2

## $A = U \Sigma V^T$ - example:

- $U \Sigma$ : Gives the coordinates of the points in the projection axis

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

Low variation around the third axis



		Movie 1 rating
1.61	0.19	-0.01
5.08	0.66	-0.03
6.82	0.85	-0.05
8.43	1.04	-0.06
1.86	-5.60	0.84
0.86	-6.93	-0.87
0.86	-2.75	0.41

# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**
- **A: Set smallest singular values to zero**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & \del{1.3} \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$



# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**
- **A: Set smallest singular values to zero**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} \approx \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**
- **A: Set smallest singular values to zero**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} \approx \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

The diagram illustrates the SVD decomposition of a matrix. The first matrix is a 7x5 matrix. The second matrix is a 7x3 matrix of singular values, with the smallest value (1.3) crossed out in red. The third matrix is a 7x5 matrix of right singular vectors, with the columns corresponding to the smallest singular value crossed out in red. The matrix is multiplied by the singular value matrix and then by the right singular vectors matrix.

# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**
- **A: Set smallest singular values to zero**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} \approx \begin{bmatrix} 0.13 & 0.02 \\ 0.41 & 0.07 \\ 0.55 & 0.09 \\ 0.68 & 0.11 \\ 0.15 & -0.59 \\ 0.07 & -0.73 \\ 0.07 & -0.29 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \end{bmatrix}$$

# SVD - Interpretation #2

## More details

- Q: How exactly is dim. reduction done?
- A: Set smallest singular values to zero

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} \approx \begin{bmatrix} 0.92 & 0.95 & 0.92 & 0.01 & 0.01 \\ 2.91 & 3.01 & 2.91 & -0.01 & -0.01 \\ 3.90 & 4.04 & 3.90 & 0.01 & 0.01 \\ 4.82 & 5.00 & 4.82 & 0.03 & 0.03 \\ 0.70 & 0.53 & 0.70 & 4.11 & 4.11 \\ -0.69 & 1.34 & -0.69 & 4.78 & 4.78 \\ 0.32 & 0.23 & 0.32 & 2.01 & 2.01 \end{bmatrix}$$

Frobenius norm:

$$\|M\|_F = \sqrt{\sum_{ij} M_{ij}^2}$$

$$\|A-B\|_F = \sqrt{\sum_{ij} (A_{ij}-B_{ij})^2}$$

is "small"

# SVD - Interpretation #2

## More details

- Q: How exactly is dim. reduction done?

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**
- **A: Set smallest singular values to zero**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & \cancel{1.3} \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**
- **A: Set smallest singular values to zero**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} \approx \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**
- **A: Set smallest singular values to zero**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} \approx \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

The diagram illustrates the SVD decomposition of a matrix. The first matrix is a 7x5 matrix. The second matrix is a 7x3 matrix of singular values, with the smallest value (1.3) crossed out in red. The third matrix is a 7x5 matrix of right singular vectors, with the columns corresponding to the smallest singular value crossed out in red. The matrices are multiplied together to reconstruct the original matrix.



# SVD - Interpretation #2

## More details

- **Q: How exactly is dim. reduction done?**
- **A: Set smallest singular values to zero**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} \approx \begin{bmatrix} 0.13 & 0.02 \\ 0.41 & 0.07 \\ 0.55 & 0.09 \\ 0.68 & 0.11 \\ 0.15 & -0.59 \\ 0.07 & -0.73 \\ 0.07 & -0.29 \end{bmatrix} \times \begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \end{bmatrix}$$

# SVD - Interpretation #2

## More details

- **Q:** How exactly is dim. reduction done?
- **A:** Set smallest singular values to zero

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} \approx \begin{bmatrix} 0.92 & 0.95 & 0.92 & 0.01 & 0.01 \\ 2.91 & 3.01 & 2.91 & -0.01 & -0.01 \\ 3.90 & 4.04 & 3.90 & 0.01 & 0.01 \\ 4.82 & 5.00 & 4.82 & 0.03 & 0.03 \\ 0.70 & 0.53 & 0.70 & 4.11 & 4.11 \\ -0.69 & 1.34 & -0.69 & 4.78 & 4.78 \\ 0.32 & 0.23 & 0.32 & 2.01 & 2.01 \end{bmatrix}$$

**A** **B**

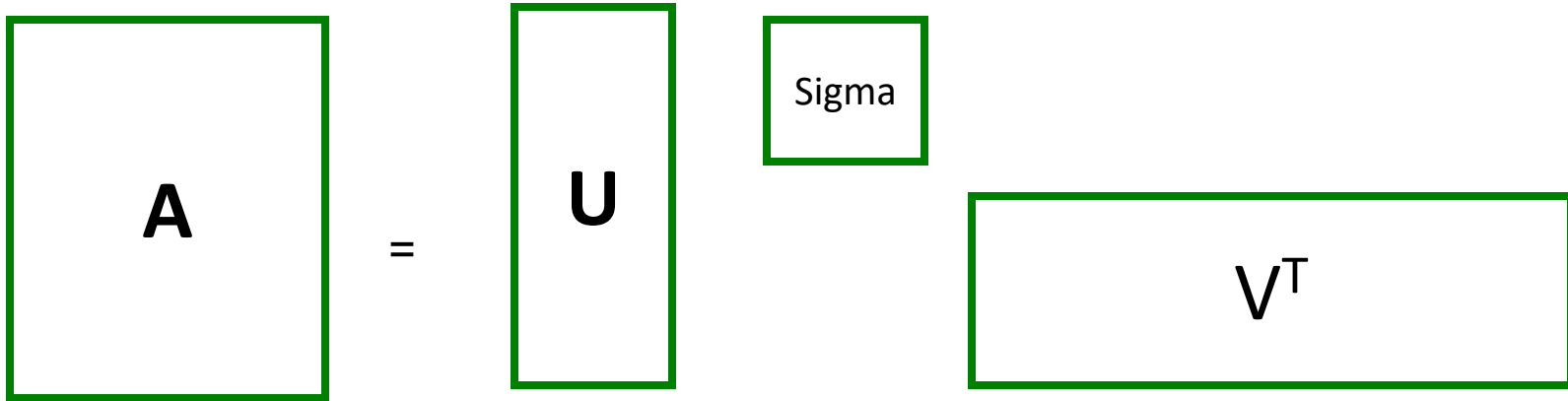
**Frobenius norm:**

$$\|M\|_F = \sqrt{\sum_{ij} M_{ij}^2}$$

$$\|A-B\|_F = \sqrt{\sum_{ij} (A_{ij}-B_{ij})^2}$$

is "small"

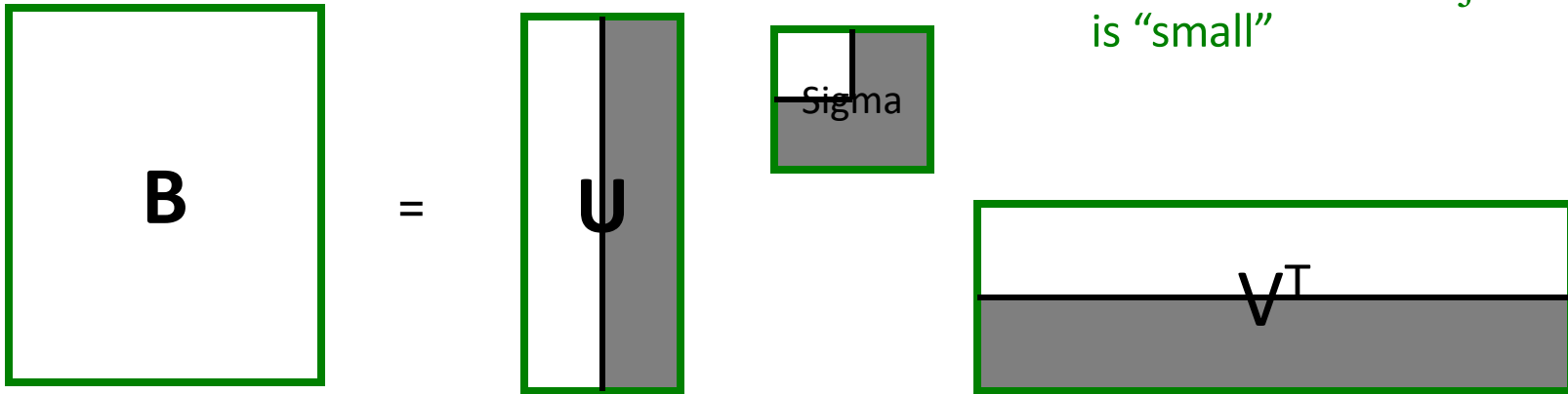
# SVD – Best Low Rank Approx.



**B is best approximation of A**

$$\|A-B\|_F = \sqrt{\sum_{ij} (A_{ij}-B_{ij})^2}$$

is "small"



# SVD – Best Low Rank Approx.

- Theorem:**

Let  $A = U \Sigma V^T$  and  $B = U S V^T$  where

$S =$  diagonal  $r \times r$  matrix with  $s_i = \sigma_i$  ( $i=1 \dots k$ ) else  $s_i = 0$

then  $B$  is a best rank( $B$ )= $k$  approx. to  $A$

What do we mean by “best”:

- $B$  is a solution to  $\min_B \|A - B\|_F$  where  $\text{rank}(B) = k$

$$\begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & \dots \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & & & x_{mn} \end{pmatrix}_{m \times n} = \begin{pmatrix} U & & & \\ u_{11} & \dots & & \\ \vdots & \ddots & & \\ u_{m1} & & & \end{pmatrix}_{m \times r} \begin{pmatrix} \Sigma & & & \\ \sigma_{11} & 0 & \dots & \\ \vdots & & & \\ 0 & & & \\ \vdots & & & \end{pmatrix}_{r \times r} \begin{pmatrix} V^T & & & \\ v_{11} & \dots & & v_{1n} \\ \vdots & \ddots & & \vdots \\ & & & \end{pmatrix}_{r \times n}$$

$$\|A - B\|_F = \sqrt{\sum_{ij} (A_{ij} - B_{ij})^2}$$

# SVD - Interpretation #2

**Equivalent:**

**'spectral decomposition' of the matrix:**

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} | & | \\ u_1 & u_2 \\ | & | \end{bmatrix} \times \begin{bmatrix} \sigma_1 & \text{⊘} \\ \text{⊘} & \sigma_2 \end{bmatrix} \times \begin{bmatrix} \text{---} & v_1 & \text{---} \\ \text{---} & v_2 & \text{---} \end{bmatrix}$$

# SVD - Interpretation #2

**Equivalent:**

**'spectral decomposition' of the matrix**

$$\begin{array}{c} \leftarrow m \quad \rightarrow \\ \begin{array}{c} \uparrow \\ n \\ \downarrow \end{array} \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{array}{c} \leftarrow k \text{ terms} \quad \rightarrow \\ \sigma_1 \quad \mathbf{u}_1 \quad \mathbf{v}_1^T + \sigma_2 \quad \mathbf{u}_2 \quad \mathbf{v}_2^T + \dots \\ \begin{array}{c} \nearrow \\ n \times 1 \end{array} \quad \begin{array}{c} \nwarrow \\ 1 \times m \end{array} \end{array}$$

**Assume:  $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \dots \geq 0$**

**Why can we set small  $\sigma_i$  to 0?**

Vectors  $\mathbf{u}_i$  and  $\mathbf{v}_i$  are unit length, so  $\sigma_i$  scales them.

So, zeroing small  $\sigma_i$  introduces less error.

# SVD - Interpretation #2

**Q: How many  $\sigma_s$  to keep?**

**A: Rule-of-a thumb:**

**keep 80-90% of 'energy' =  $\sum_i \sigma_i^2$**

$$\begin{array}{c} \left. \begin{array}{c} \uparrow \\ \downarrow \end{array} \right\} n \\ \left[ \begin{array}{ccccc} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{array} \right] \end{array} \begin{array}{c} \longleftarrow m \qquad \longrightarrow \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{array} = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^T + \dots$$

**Assume:  $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \dots$**

# SVD - Complexity

- **To compute SVD:**
  - $O(nm^2)$  or  $O(n^2m)$  (whichever is less)
- **But:**
  - Less work, if we just want singular values
  - or if we want first  $k$  singular vectors
  - or if the matrix is sparse
- **Implemented in** linear algebra packages like
  - LINPACK, Matlab, SPlus, Mathematica, Sklearn ...



# SVD - Conclusions so far

- **SVD:  $A = U \Sigma V^T$ : unique**
  - **U**: user-to-concept similarities
  - **V**: movie-to-concept similarities
  - $\Sigma$  : strength of each concept
- **Dimensionality reduction:**
  - keep the few largest singular values (80-90% of 'energy')
  - SVD: picks up linear correlations

# SVD - for Document-Term Matrix

- **SVD:  $A = U \Sigma V^T$ : unique**
  - **U**: document-to-concept similarities
  - **V**: term-to-concept similarities
  - $\Sigma$  : strength of each concept
- **Dimensionality reduction:**
  - keep the few largest singular values (80-90% of 'energy')
  - SVD: picks up linear correlations

# Case study: How to query?

- **Q: Find users that like 'Matrix'**
- **A: Map query into a 'concept space' – how?**

↑ SciFi  
↓  
↑ Romance  
↓

Matrix	Alien	Serenity	Casablanca	Amelie						
1	1	1	0	0	=	0.13	0.02	-0.01		
3	3	3	0	0		0.41	0.07	-0.03		
4	4	4	0	0		0.55	0.09	-0.04		
5	5	5	0	0		0.68	0.11	-0.05	x	12.4
0	2	0	4	4		0.15	-0.59	0.65		0
0	0	0	5	5		0.07	-0.73	-0.67		9.5
0	1	0	2	2		0.07	-0.29	0.32		0
										1.3
						0.56	0.59	0.56	0.09	0.09
						0.12	-0.02	0.12	-0.69	-0.69
						0.40	-0.80	0.40	0.09	0.09

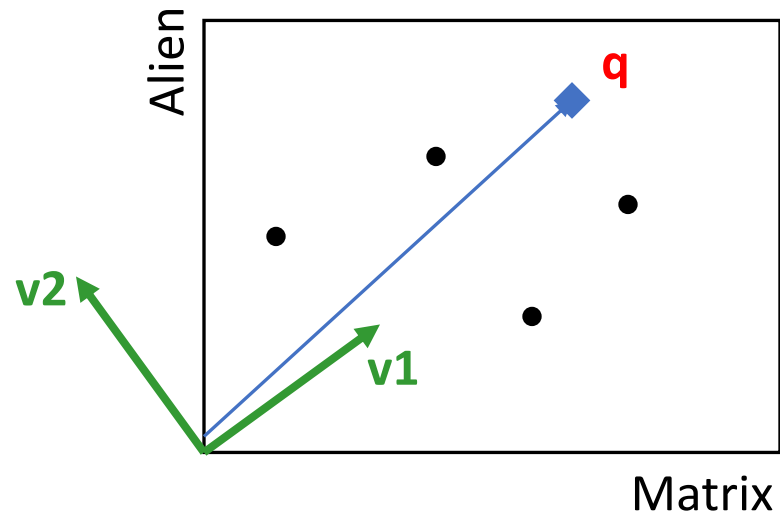
70

# Case study: How to query?

- **Q: Find users that like 'Matrix'**
- **A: Map query into a 'concept space' – how?**

$$q = \begin{bmatrix} \text{Matrix} \\ 5 \\ \text{Alien} \\ 0 \\ \text{Serenity} \\ 0 \\ \text{Casablanca} \\ 0 \\ \text{Amelie} \\ 0 \end{bmatrix}$$

**Project into concept space:**  
Inner product with each  
'concept' vector  $v_i$

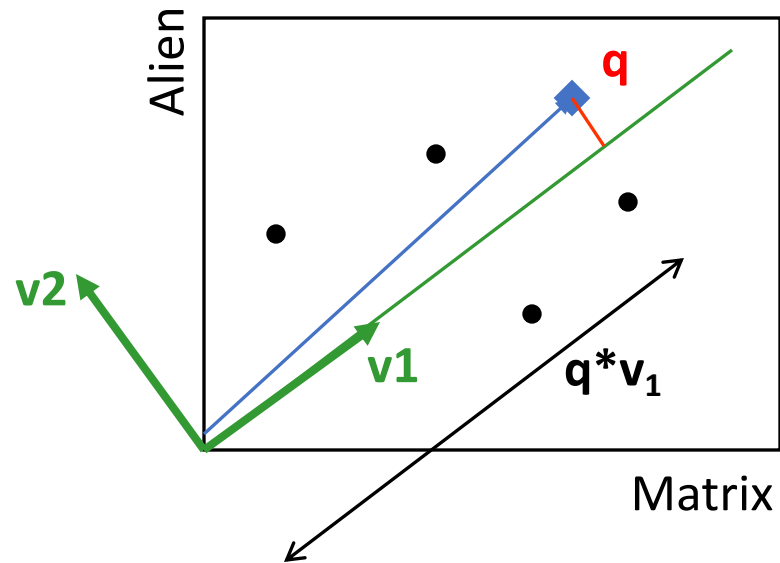


# Case study: How to query?

- **Q: Find users that like 'Matrix'**
- **A: Map query into a 'concept space' – how?**

$$q = \begin{bmatrix} \text{Matrix} \\ 5 \\ \text{Alien} \\ 0 \\ \text{Serenity} \\ 0 \\ \text{Casablanca} \\ 0 \\ \text{Amelie} \\ 0 \end{bmatrix}$$

**Project into concept space:**  
Inner product with each  
'concept' vector  $v_i$



# Case study: How to query?

Compactly, we have:

$$q_{\text{concept}} = q V$$

E.g.:

$$q = \begin{bmatrix} \text{Matrix} \\ 5 & 0 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 0.56 & 0.12 \\ 0.59 & -0.02 \\ 0.56 & 0.12 \\ 0.09 & -0.69 \\ 0.09 & -0.69 \end{bmatrix} = \begin{bmatrix} \text{SciFi-concept} \\ \downarrow \\ 2.8 & 0.6 \end{bmatrix}$$

movie-to-concept similarities (V)

# Case study: How to query?

- How would the user  $d$  that rated ('Alien', 'Serenity') be handled?

$$\mathbf{d}_{\text{concept}} = \mathbf{d} \mathbf{V}$$

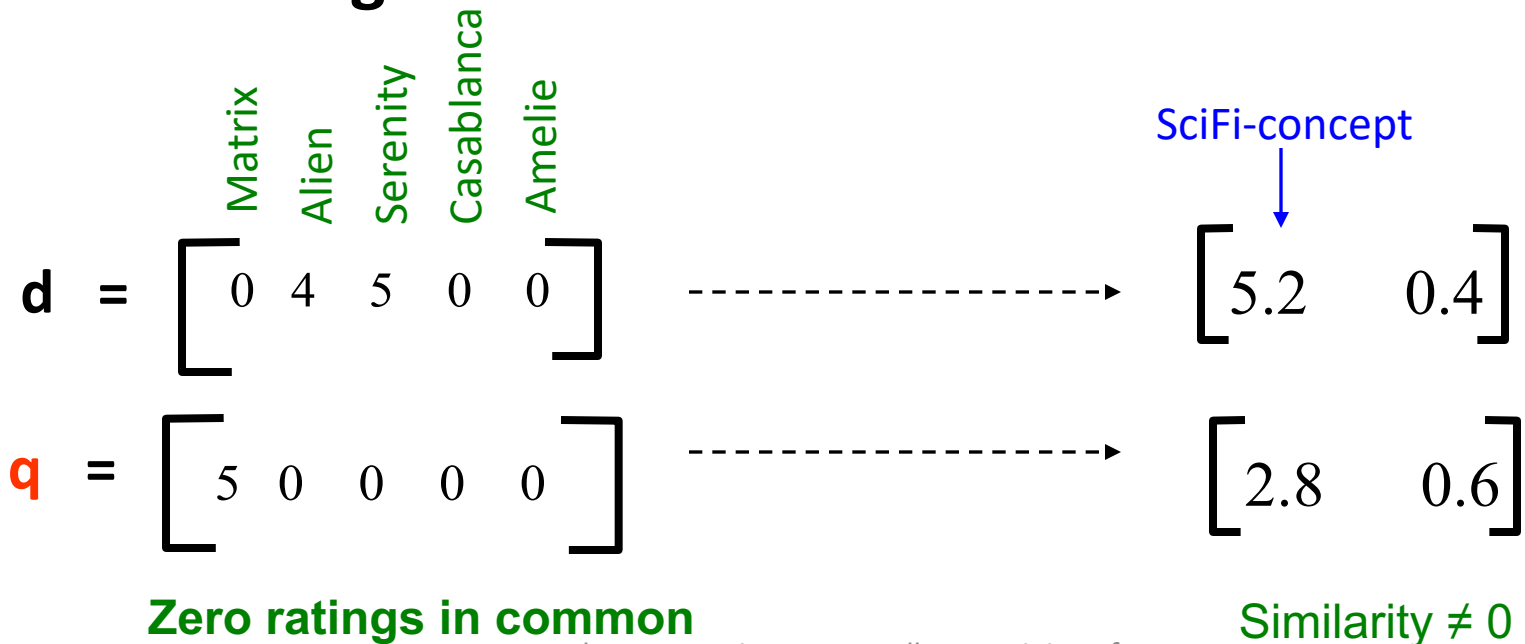
E.g.:

$$\mathbf{q} = \begin{bmatrix} \text{Matrix} \\ 0 \\ \text{Alien} \\ 4 \\ \text{Serenity} \\ 5 \\ \text{Casablanca} \\ 0 \\ \text{Amelie} \\ 0 \end{bmatrix} \mathbf{X} \begin{bmatrix} 0.56 & 0.12 \\ 0.59 & -0.02 \\ 0.56 & 0.12 \\ 0.09 & -0.69 \\ 0.09 & -0.69 \end{bmatrix} = \begin{bmatrix} \text{SciFi-concept} \\ 5.2 \\ 0.4 \end{bmatrix}$$

movie-to-concept similarities (V)

# Case study: How to query?

- **Observation:** User  $d$  that rated (*'Alien'*, *'Serenity'*) will be **similar** to user  $q$  that rated (*'Matrix'*), although  $d$  and  $q$  have **zero ratings in common!**





# SVD: Drawbacks

+ **Optimal low-rank approximation**

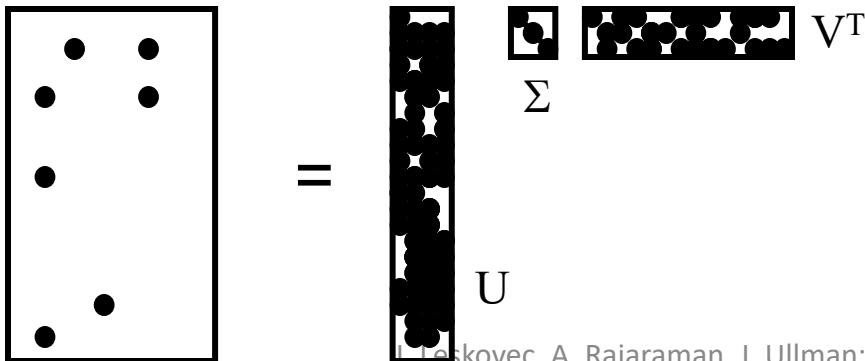
in terms of Frobenius norm

- **Interpretability problem:**

- A singular vector specifies a linear combination of all input columns or rows

- **Lack of sparsity:**

- Singular vectors are **dense!**



# Latent Semantic Analysis

Applying SVD to the Document Term Matrix