# Singular Value Decomposition

## Geometric Algorithms
## Lecture 26

CAS CS 132

# Introduction

# Recap Problem (+ Course Evaluations)

*Find an orthogonal diagonalization of* $\begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix}$.

https://www.bu.edu/courseeval

# Answer

# Objectives

1. Finish up our discussion of quadratic forms.
2. Introduce the singular value decomposition (probably the most important matrix decomposition for computer science).
3. Talk very briefly about what to do after this course if you want (or have to) to see more linear algebra.

# Quadratic Forms (Finishing Up)

# Quadratic Forms

**Definition.** A **quadratic form** is an function of variables $x_1, \ldots, x_n$ in which every term has degree two.

Examples:

# Quadratic Forms and Symmetric Matrices

**Fact.** Every quadratic form can be represented as

$$\mathbf{x}^T A\mathbf{x}$$

where $A$ is symmetric.

Example:

# Example: Computing the Quadratic Form for a Matrix

$$A = \begin{bmatrix} 3 & -2 \\ -2 & 7 \end{bmatrix}$$

This means, given a symmetric matrix $A$, we can compute its corresponding quadratic form:

# Quadratic forms and Symmetric Matrices (Again)

Furthermore, we can generally say

$$\mathbf{x}^T A \mathbf{x} = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} x_i x_j = \sum_{i=1}^{n} A_{ii} x_i^2 + \sum_{i \neq j} (A_{ij} + A_{ji}) x_i x_j$$

Verify:

# A Slightly more Complicated Example

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

Let's expand $\mathbf{x}^T A \mathbf{x}$:

# Matrices from Quadratic Forms

$$Q(\mathbf{x}) = 5x_1^2 + 3x_2^2 + 2x_3^2 - x_1x_2 + 8x_2x_3$$

We can also go in the other direction. Let's express this as $\mathbf{x}^T A\mathbf{x}$:

# How To: Matrices of Quadratic Forms

**Problem.** Given a quadratic form $Q(\mathbf{x})$, find the symmetric matrix $A$ such that $Q(\mathbf{x}) = \mathbf{x}^T A \mathbf{x}$.

**Solution.**

» if $Q(\mathbf{x})$ has the term $\alpha x_i^2$ then $A_{ii} = \alpha$

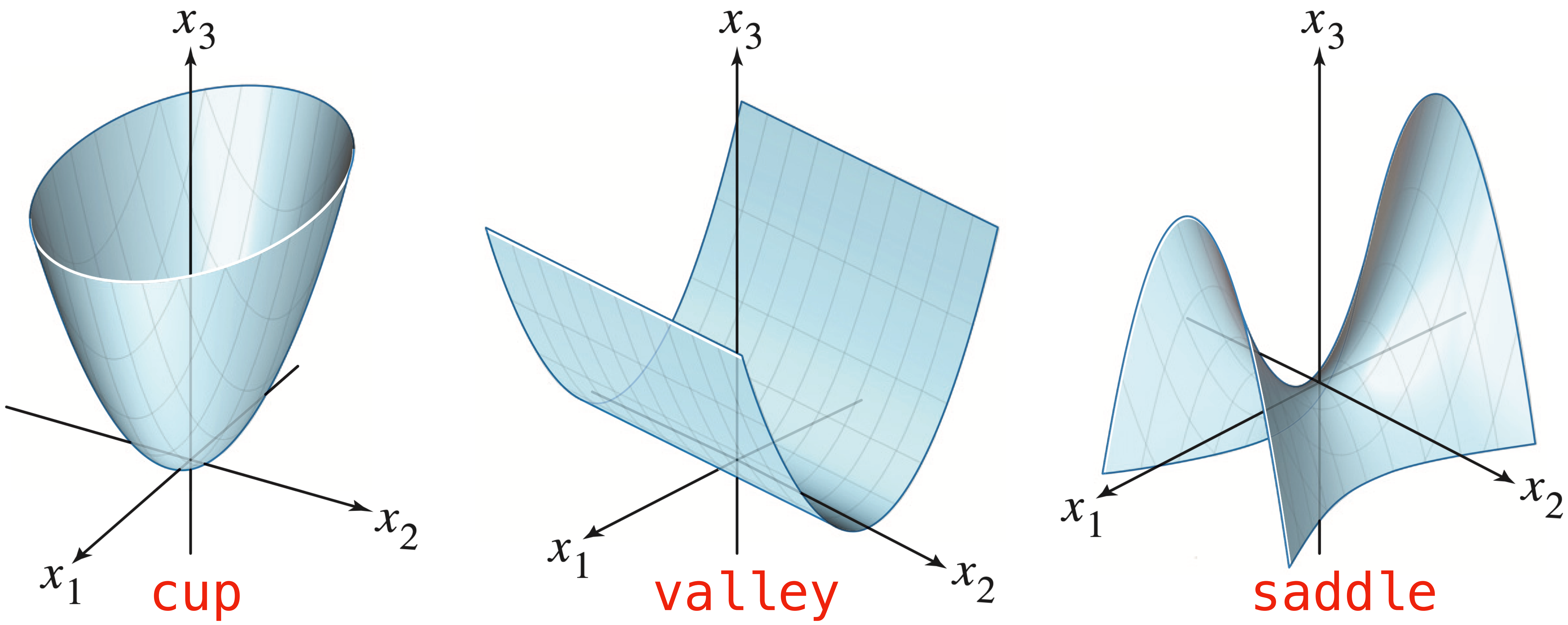
» if $Q(\mathbf{x})$ has the term $\alpha x_i x_j$, then $A_{ij} = A_{ji} = \dfrac{\alpha}{2}$

# Example

$$Q(x_1, x_2, x_3, x_4) = x_1^2 + 3x_2^2 - 2x_3x_4 - 6x_4^2 + 7x_1x_3$$

*Find the symmetric matrix $A$ such that $Q(\mathbf{x}) = \mathbf{x}^T A\mathbf{x}$.*

# Shapes of of Quadratic Forms

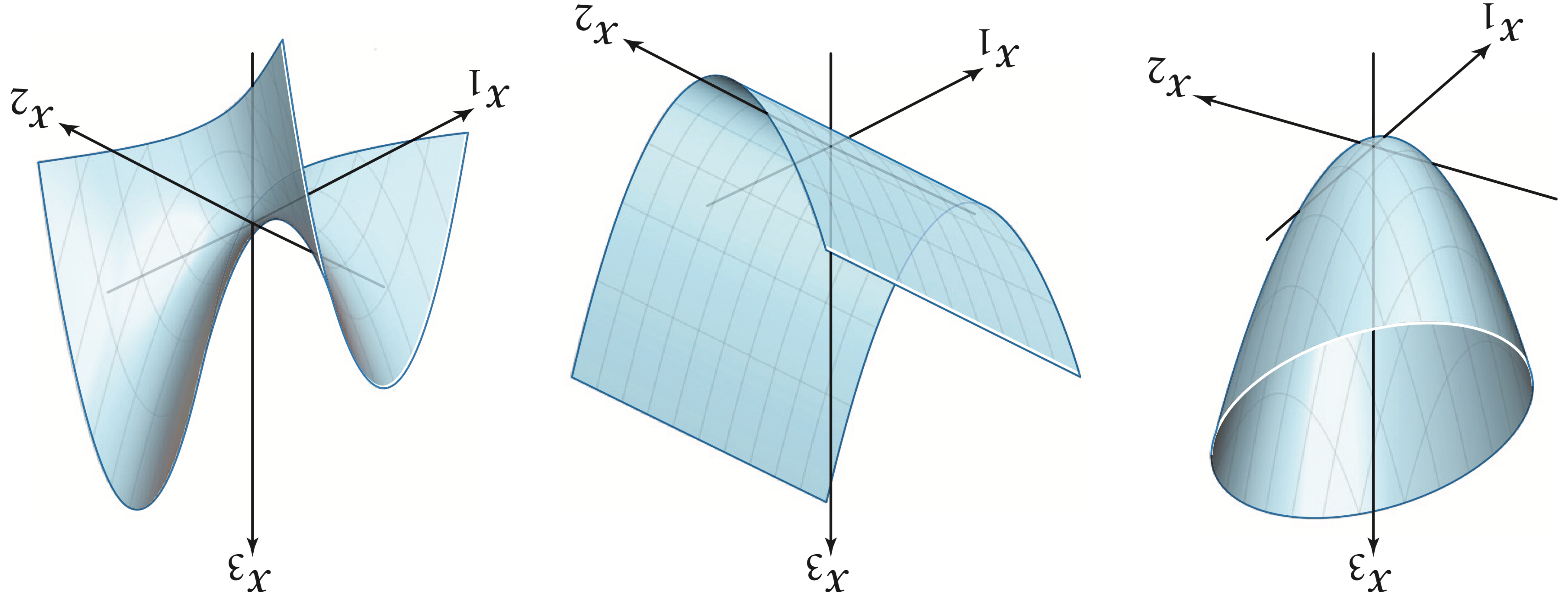



There are essentially three possible shapes (six if you include the negations).

*Can we determine what shape it will be mathematically?*

Linear Algebra and its Applications, Lay, Lay, McDonald

# Shapes of of Quadratic Forms

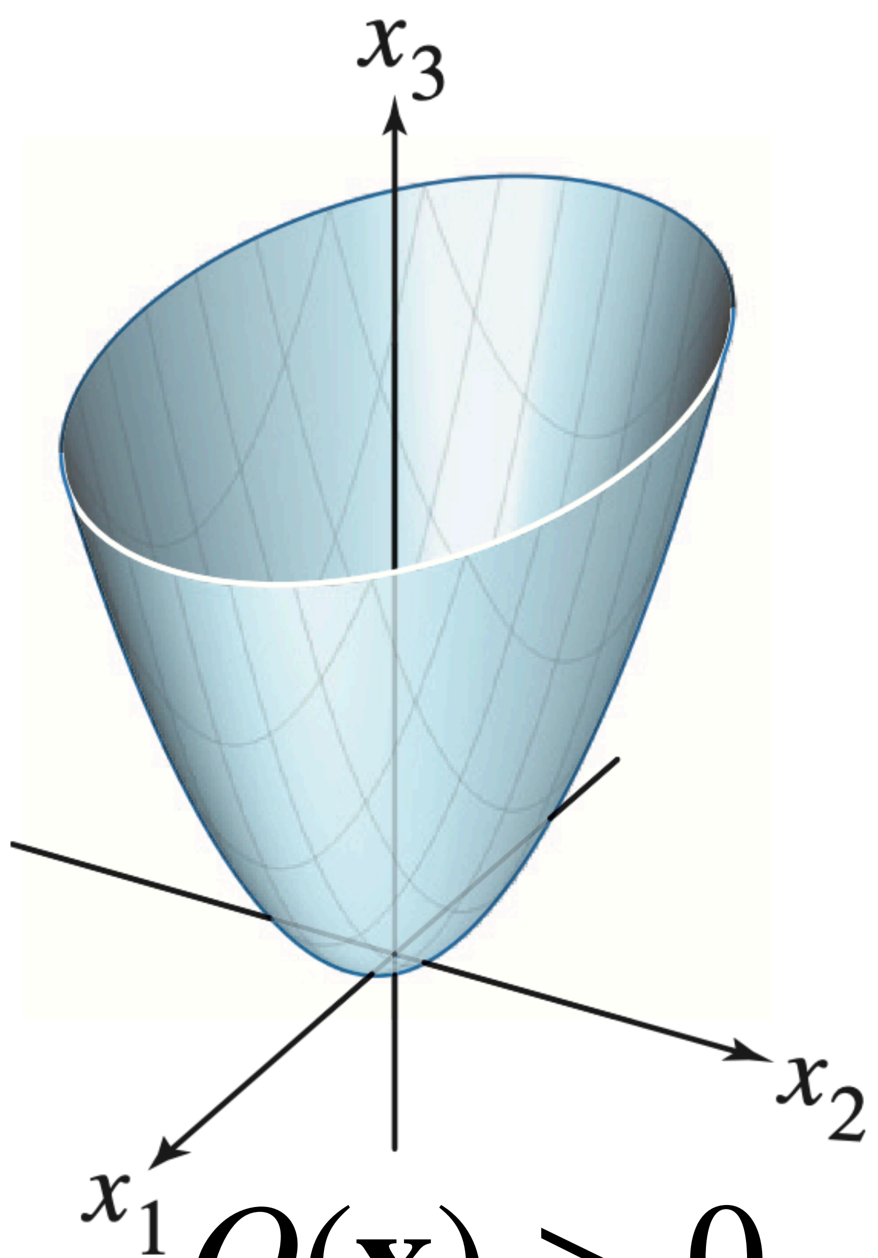


There are essentially three possible shapes (six if you include the negations).

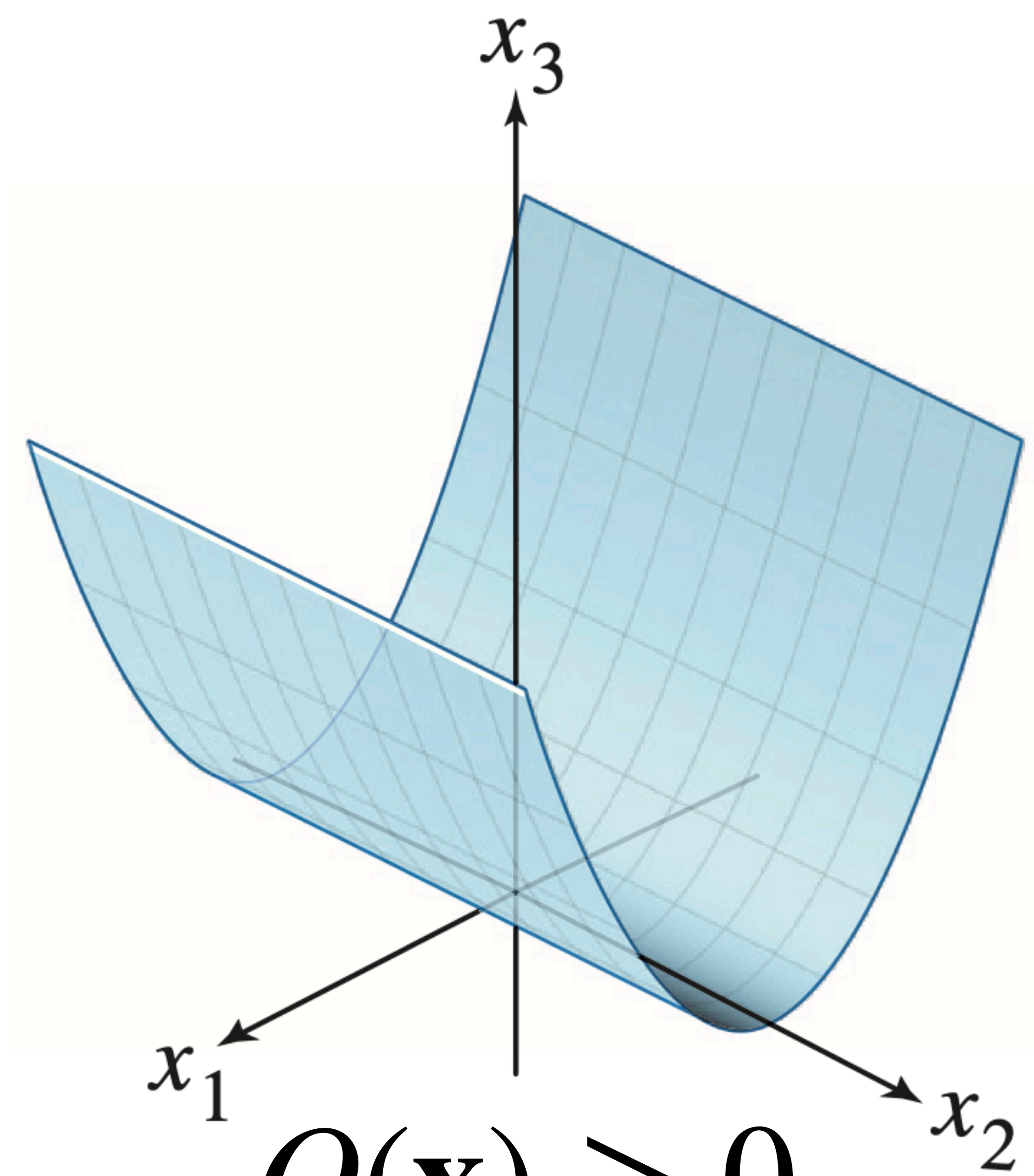
*Can we determine what shape it will be mathematically?*

Linear Algebra and its Applications, Lay, Lay, McDonald

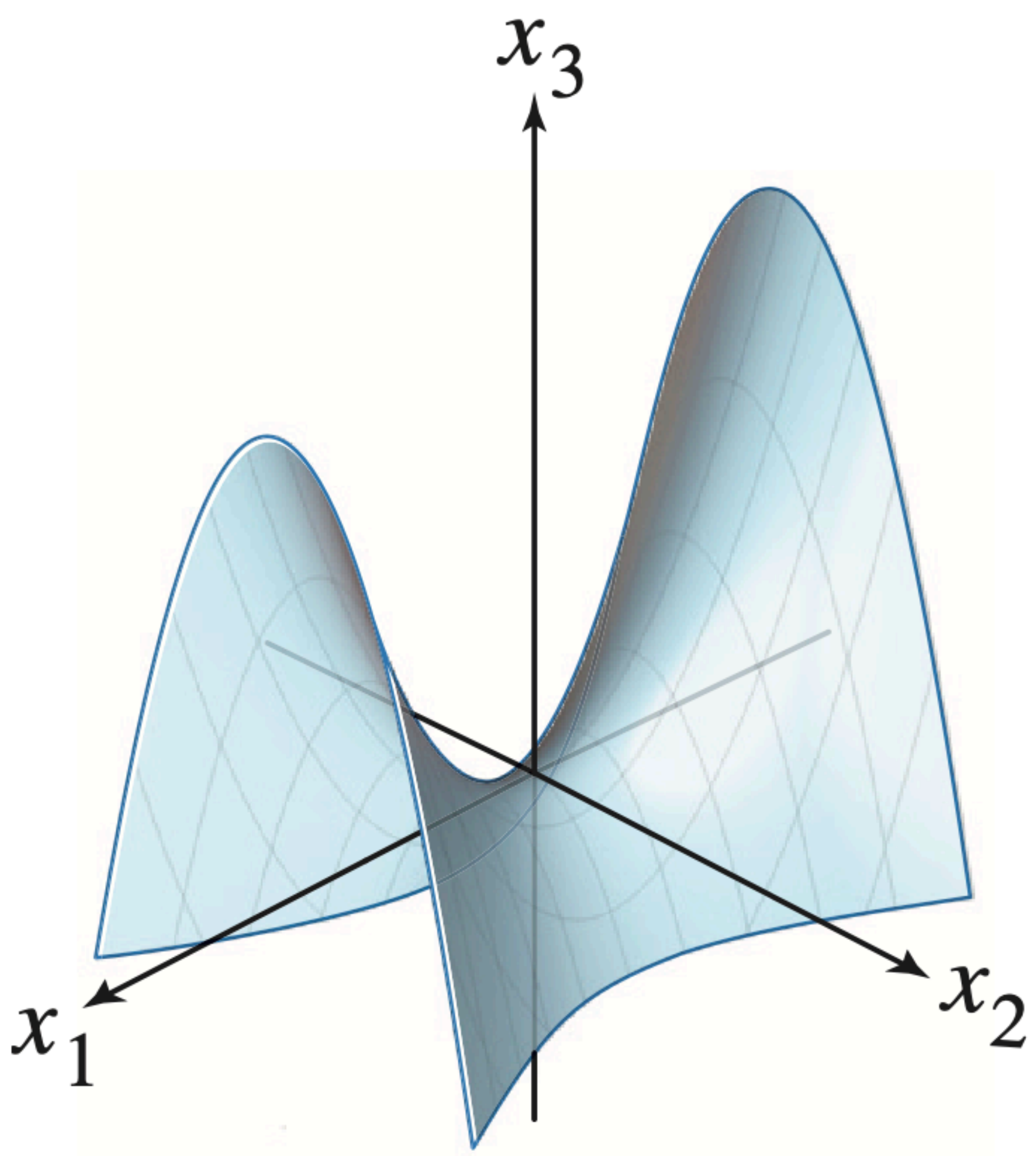
# Definiteness



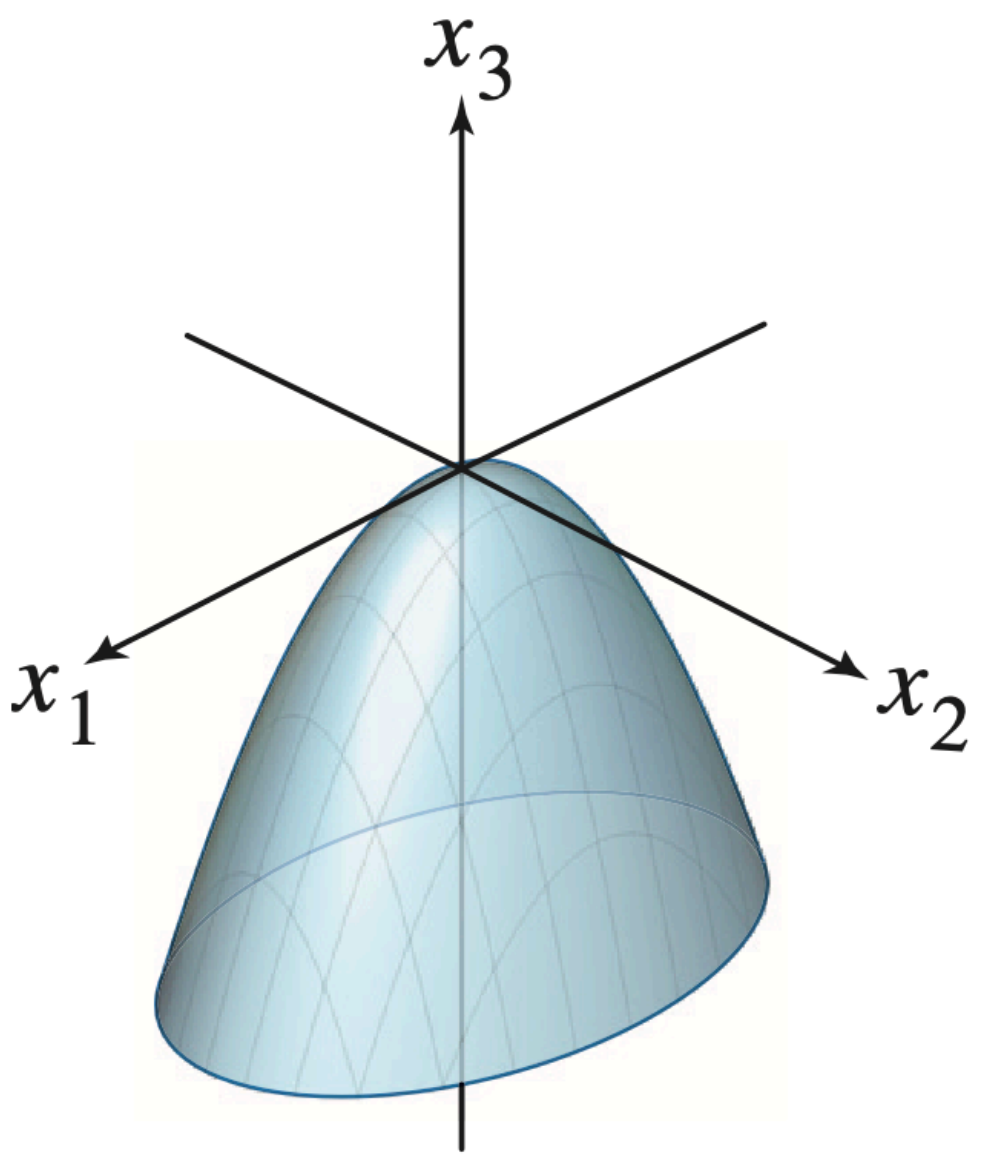
$Q(\mathbf{x}) > 0$



$Q(\mathbf{x}) \geq 0$



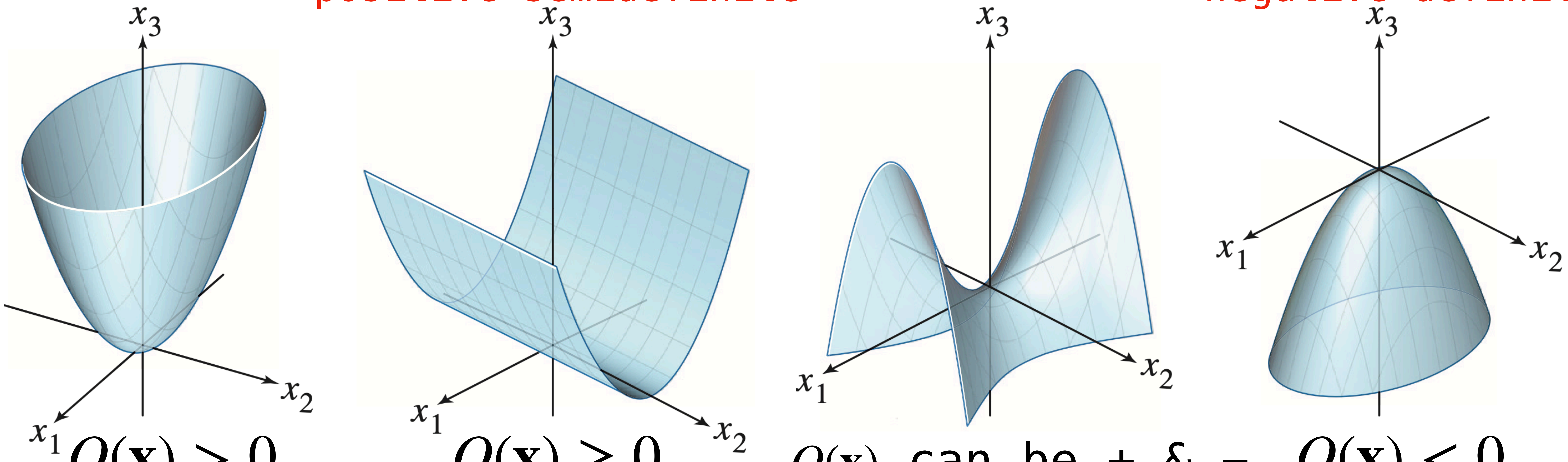
$Q(\mathbf{x})$ can be + & –



$Q(\mathbf{x}) < 0$

For $\mathbf{x} \neq \mathbf{0}$, each of the above graphs satisfy the associated properties.

# Definiteness




positive semidefinite — negative definite

$Q(\mathbf{x}) > 0$ — $Q(\mathbf{x}) \geq 0$ — $Q(\mathbf{x})$ can be + & − — $Q(\mathbf{x}) < 0$

positive definite — indefinite

For $\mathbf{x} \neq \mathbf{0}$, each of the above graphs satisfy the associated properties.

# Definiteness and Eigenvectors

**Theorem.** For a symmetric matrix $A$, the quadratic form $\mathbf{x}^T A \mathbf{x}$

» **positive definite** ≡ all <u>positive</u> eigenvalues

» **positive semidefinite** ≡ all <u>nonnegative</u> eigenvalues

» **indefinite** ≡ positive and negative eigenvalues

» **negative definite** ≡ all <u>negative</u> eigenvalues

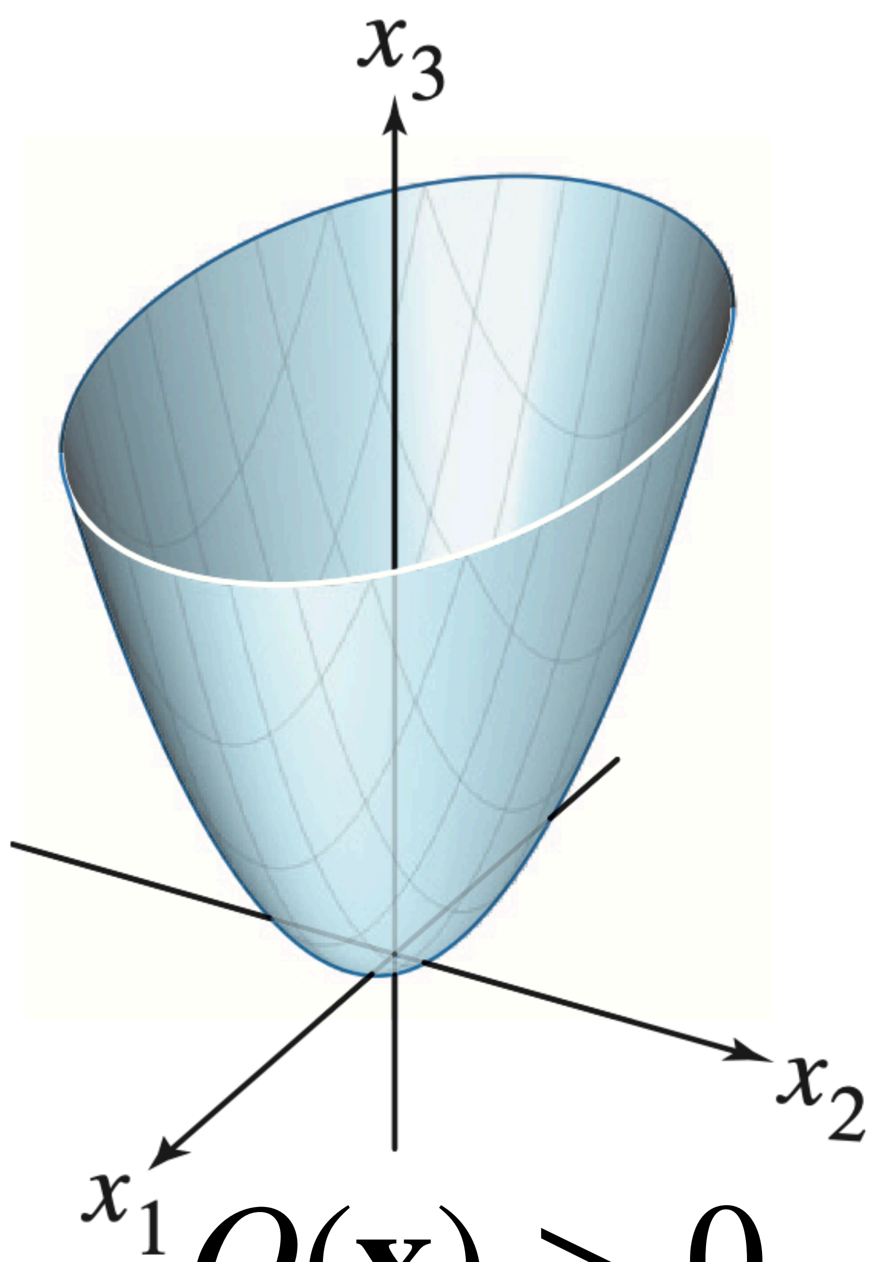
# Definiteness

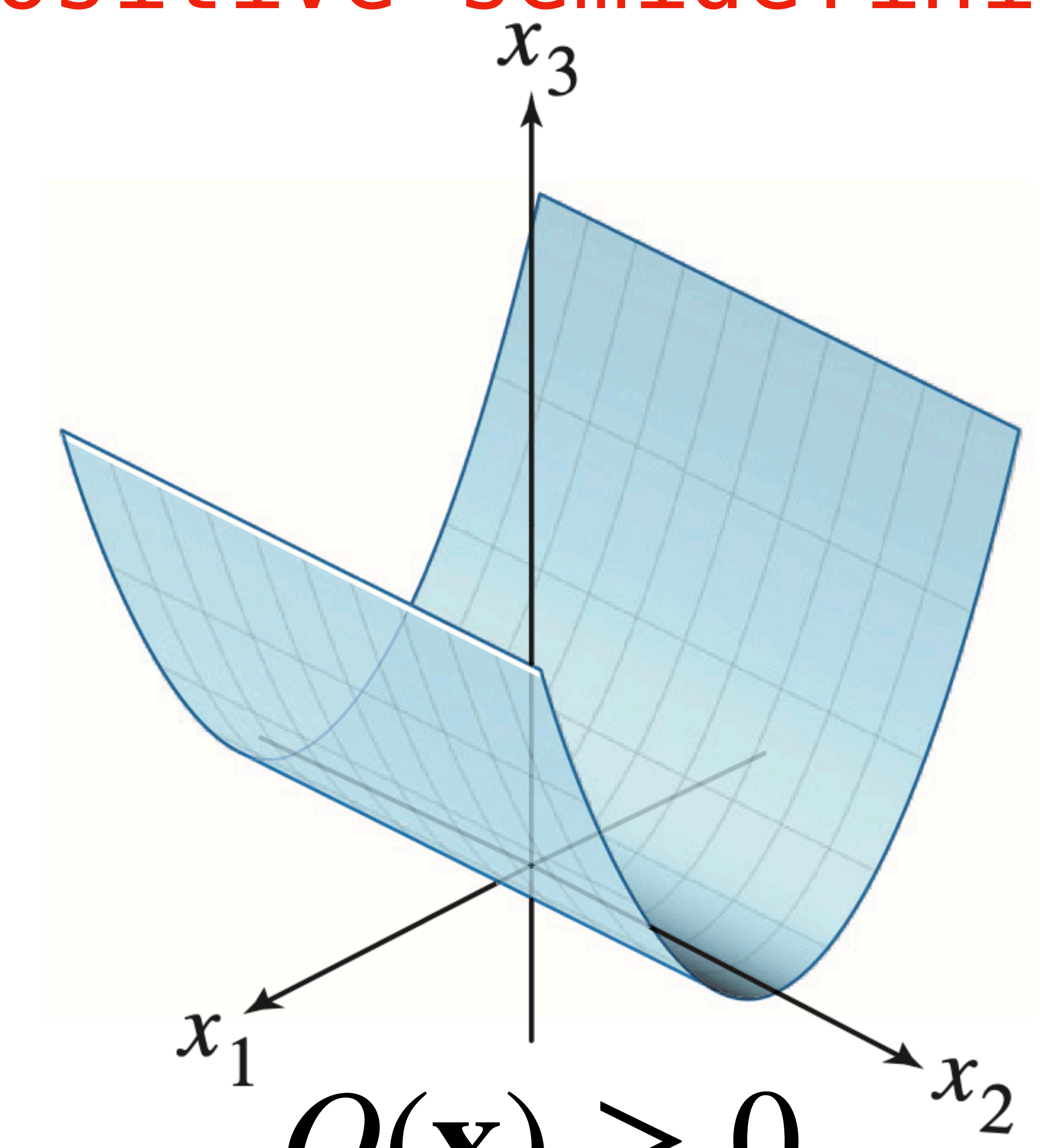
all nonneg. eigenvals

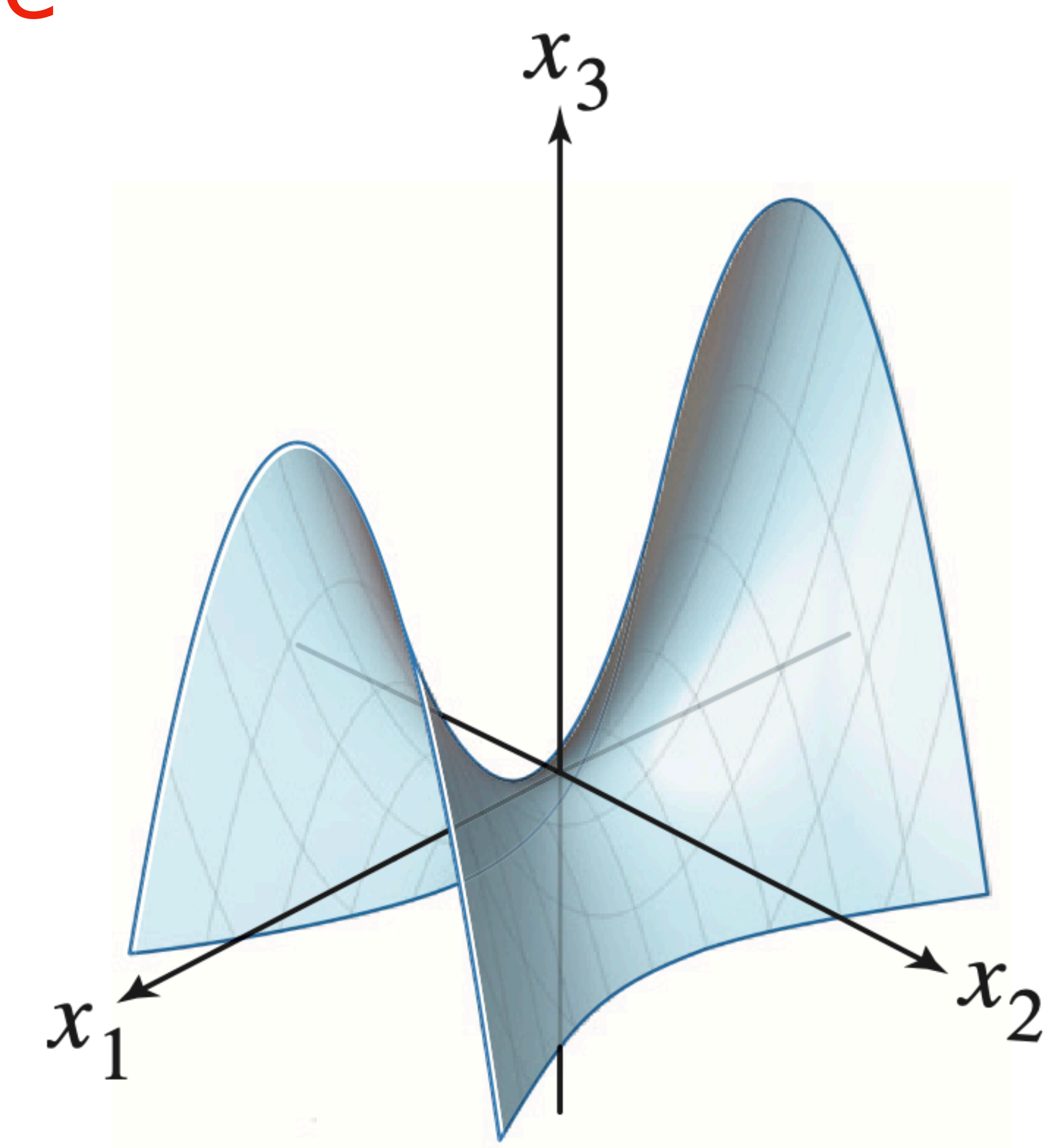
positive semidefinite

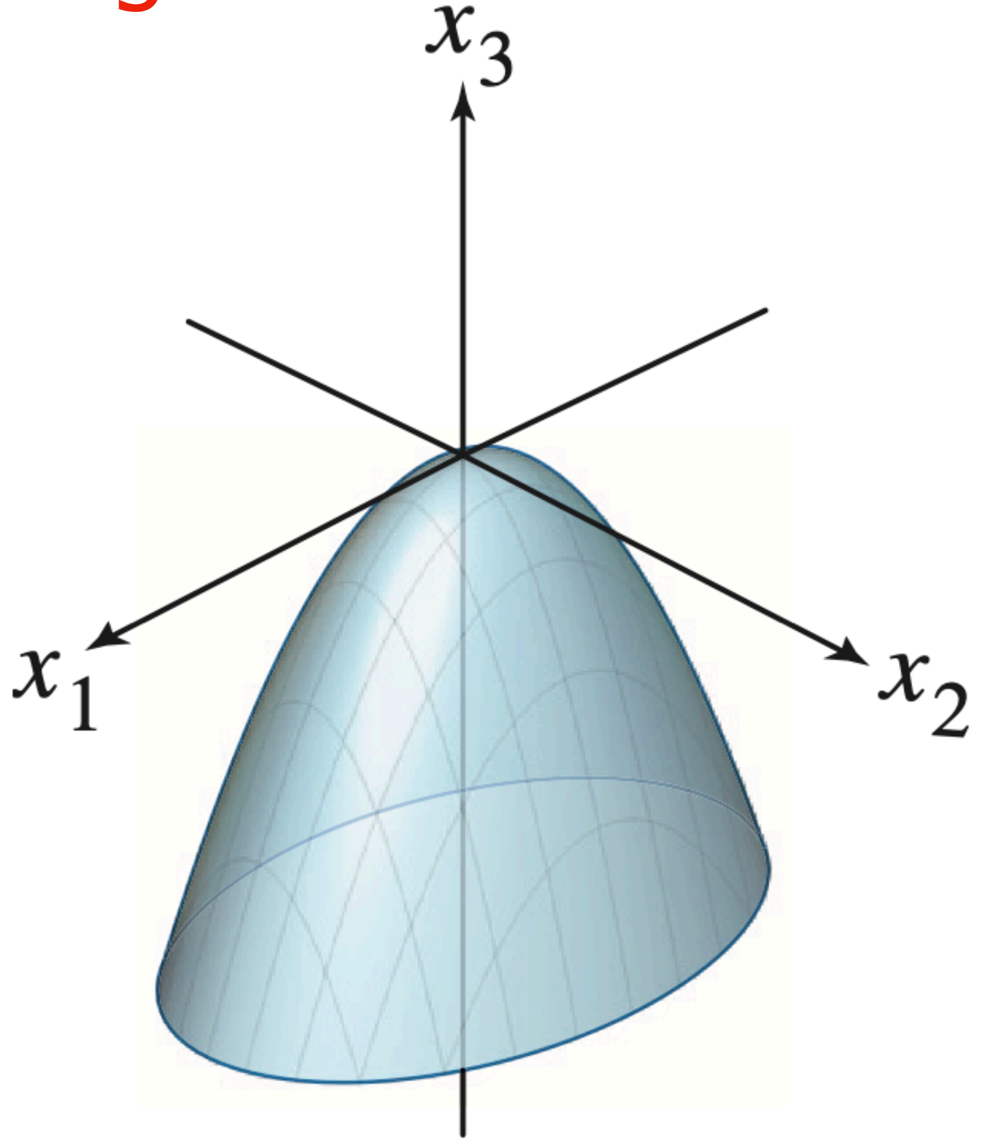
all neg. eigenvals

negative definite




$Q(\mathbf{x}) > 0$ $\quad Q(\mathbf{x}) \geq 0$ $\quad Q(\mathbf{x})$ can be + & – $\quad Q(\mathbf{x}) < 0$

positive definite

all pos. eigenvals

indefinite

pos. and neg. eigenvals

# Example

$$Q(x_1, x_2, x_3) = 3x_1^2 + x_2^2 + 4x_2x_3 + x_3^2$$

```
Let's determine which case this is:
```

# Constrained Optimization

# In General

# In General

Given a function $f: \mathbb{R}^n \to \mathbb{R}$ and a set of vectors $X$ from $\mathbb{R}^n$ the **constrained minimization problem** for $f$ over $X$ is the problem of determining

$$\min_{\mathbf{v} \in X} f(\mathbf{v})$$

# In General

Given a function $f : \mathbb{R}^n \to \mathbb{R}$ and a set of vectors $X$ from $\mathbb{R}^n$ the **constrained minimization problem** for $f$ over $X$ is the problem of determining

$$\min_{\mathbf{v} \in X} f(\mathbf{v})$$

(analogously for maximization)

# In General

Given a function $f : \mathbb{R}^n \to \mathbb{R}$ and a set of vectors $X$ from $\mathbb{R}^n$ the **constrained minimization problem** for $f$ over $X$ is the problem of determining

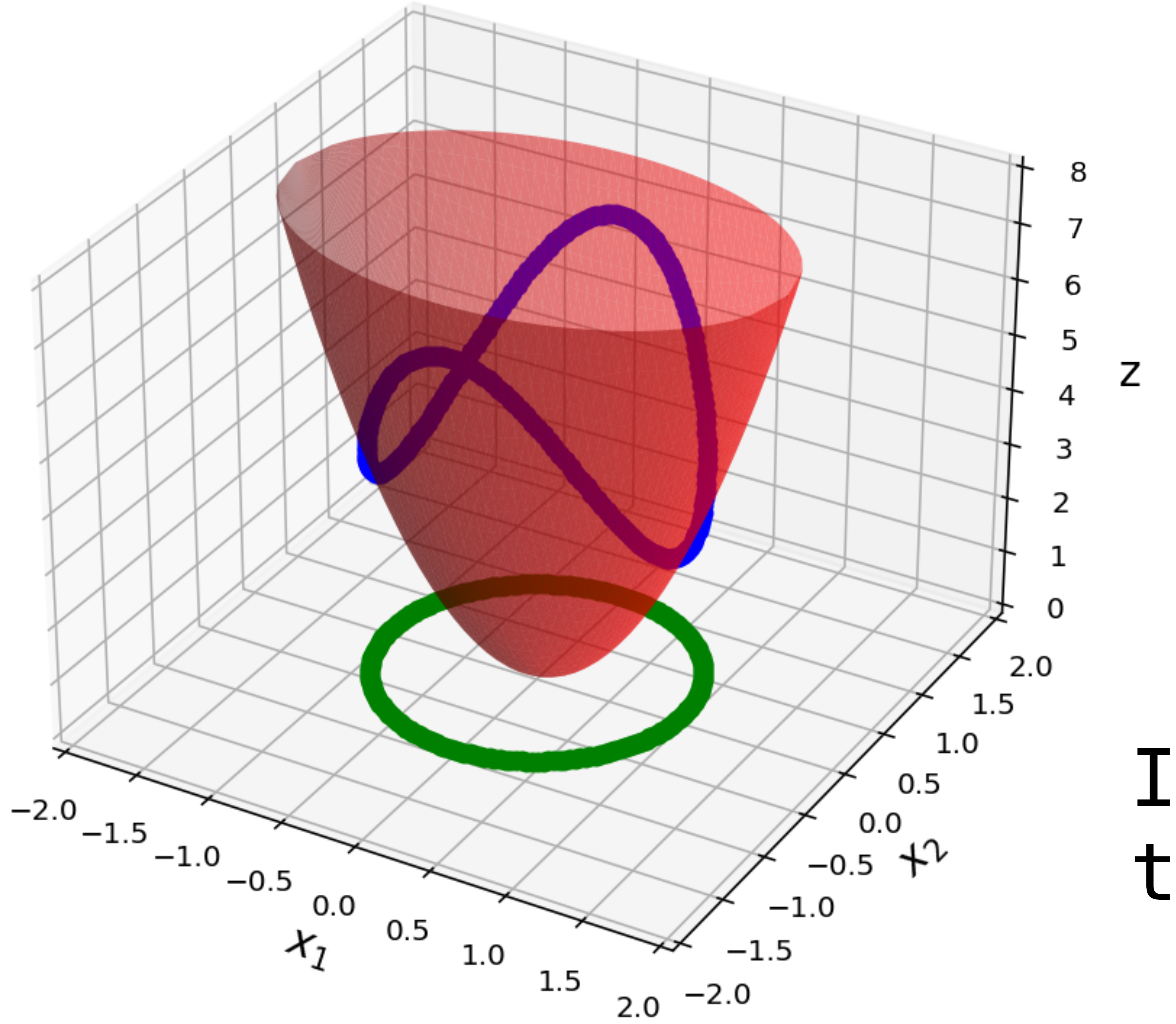
$$\min_{\mathbf{v} \in X} f(\mathbf{v})$$

(analogously for maximization)

*Find the smallest value of $f(\mathbf{v})$ subject to choosing a vector in $X$*

# Constrained Optimization for Quadratic Forms and Unit Vectors

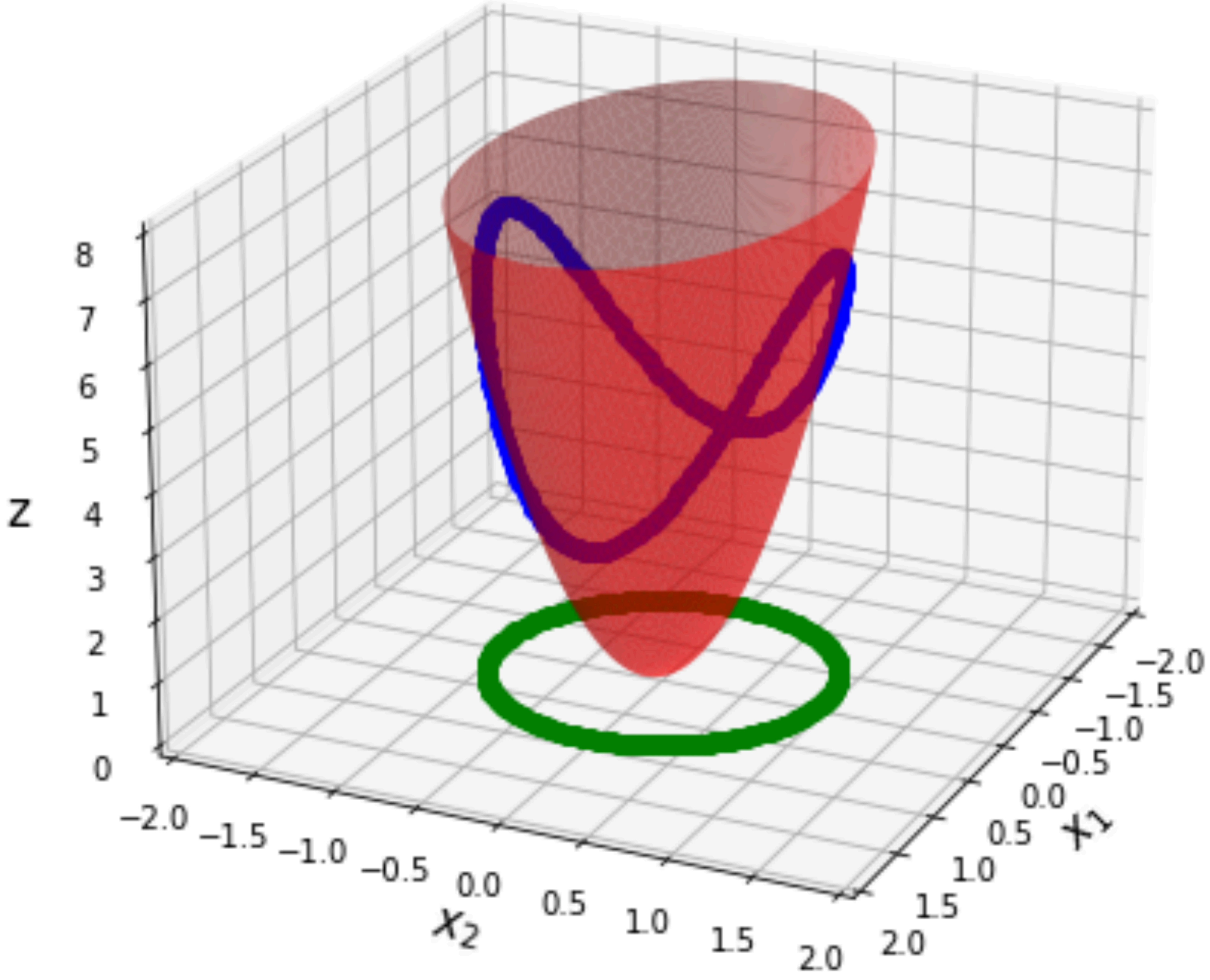
mini/maximize $\mathbf{x}^T A \mathbf{x}$ subject to $\|\mathbf{x}\| = 1$




It's common to constraint to unit vectors.

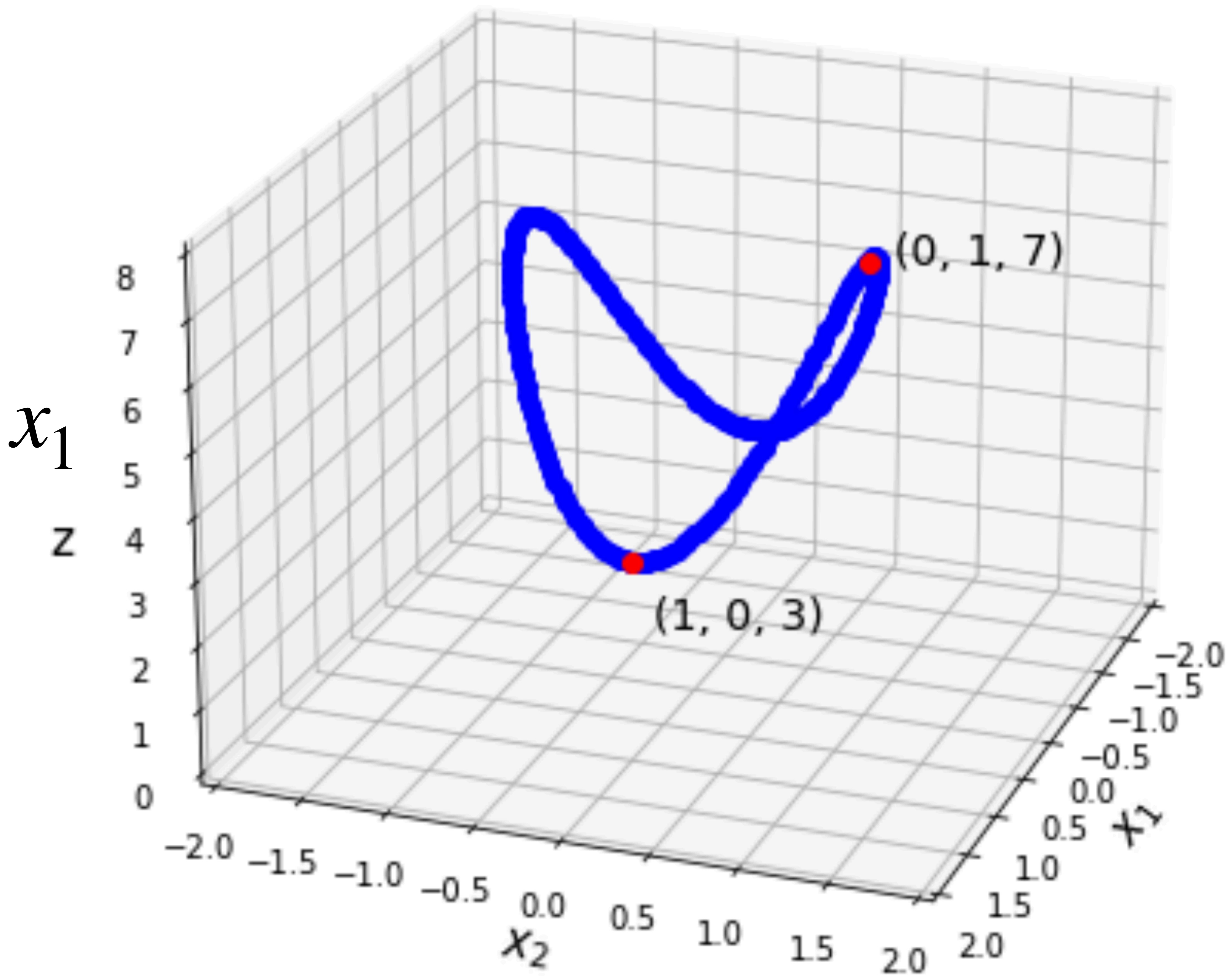
# **Example:** $3x_1^2 + 7x_2^2$

```
What are the min/max
values?:
```



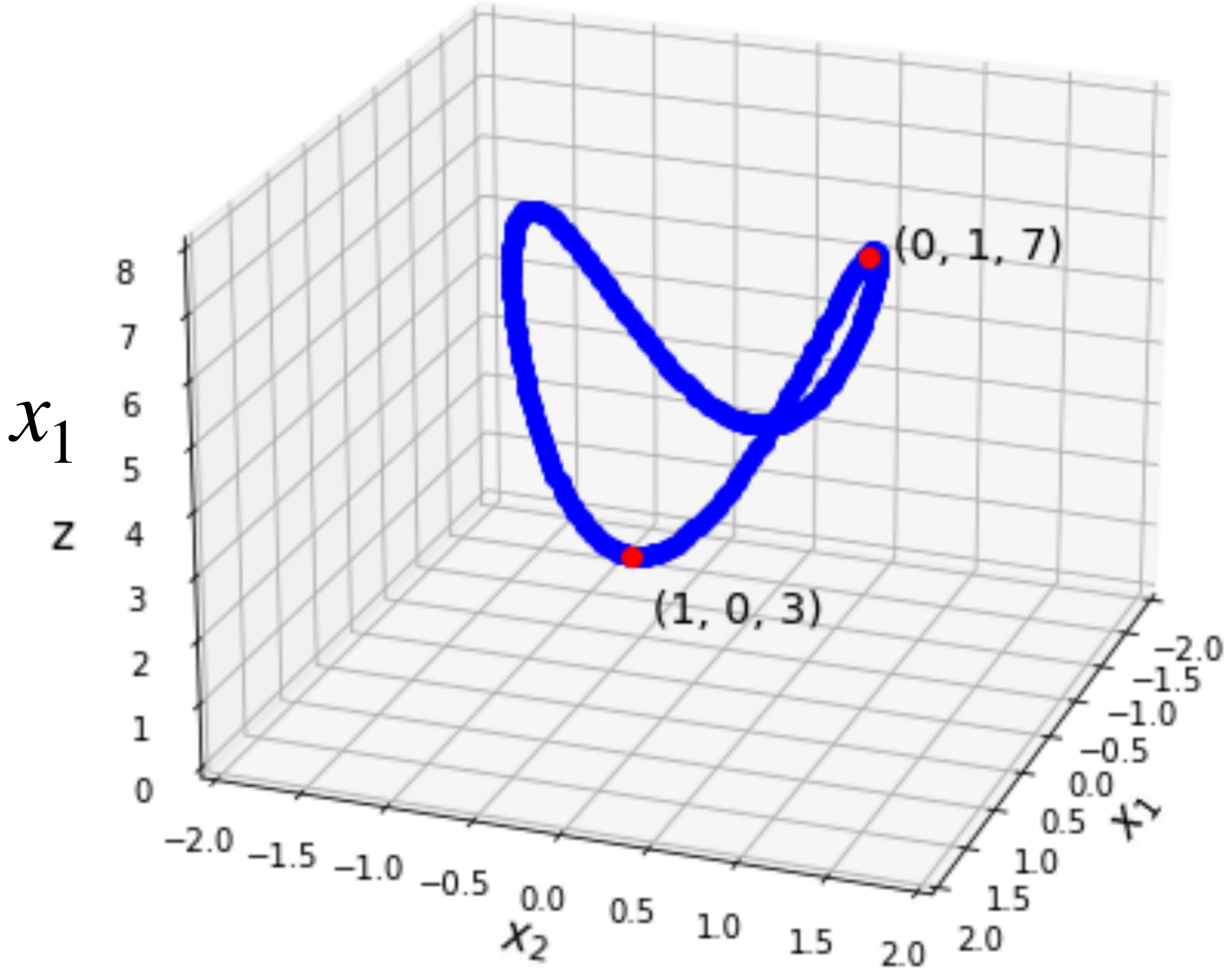
# Example: $3x_1^2 + 7x_2^2$

The minimum and maximum values are attained when the "weight" of the vector is distributed all on $x_1$ or $x_2$.



# **Example:** $3x_1^2 + 7x_2^2$

```
What is the matrix?:
```



# Constrained Optimization and Eigenvalues

**Theorem.** For a symmetric matrix $A$, with *largest* eigenvalue $\lambda_1$ and *smallest* eigenvalue $\lambda_n$

$$\max_{\|\mathbf{x}\|=1} \mathbf{x}^T A \mathbf{x} = \lambda_1 \qquad \min_{\|\mathbf{x}\|=1} \mathbf{x}^T A \mathbf{x} = \lambda_n$$

*No matter the shape of $A$, this will hold.*

# How To: Constrained Optimization

# How To: Constrained Optimization

**Problem.** Find the maximum value of $\mathbf{x}^T A\mathbf{x}$ subject to $\|\mathbf{x}\| = 1$.

# How To: Constrained Optimization

**Problem.** Find the maximum value of $\mathbf{x}^T A \mathbf{x}$ subject to $\|\mathbf{x}\| = 1$.

**Solution.** Find the largest eigenvalue of $A$, this will be the maximum value.

# How To: Constrained Optimization

**Problem.** Find the maximum value of $\mathbf{x}^T A \mathbf{x}$ subject to $\|\mathbf{x}\| = 1$.

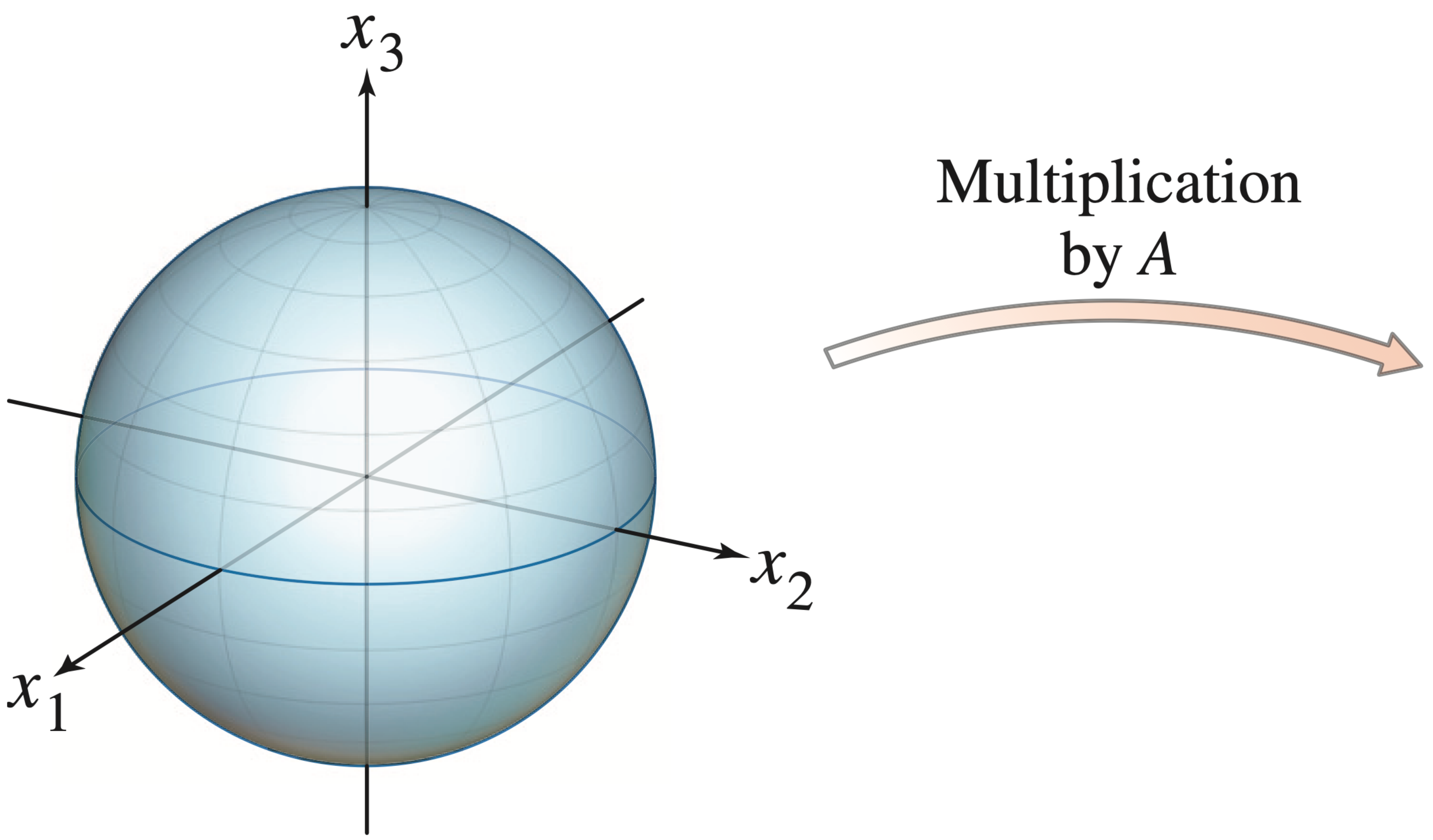
**Solution.** Find the largest eigenvalue of $A$, this will be the maximum value.

*(Use NumPy)*

# Singular Value Decomposition

# Question

*What shape is a the unit sphere after a linear transformation?*

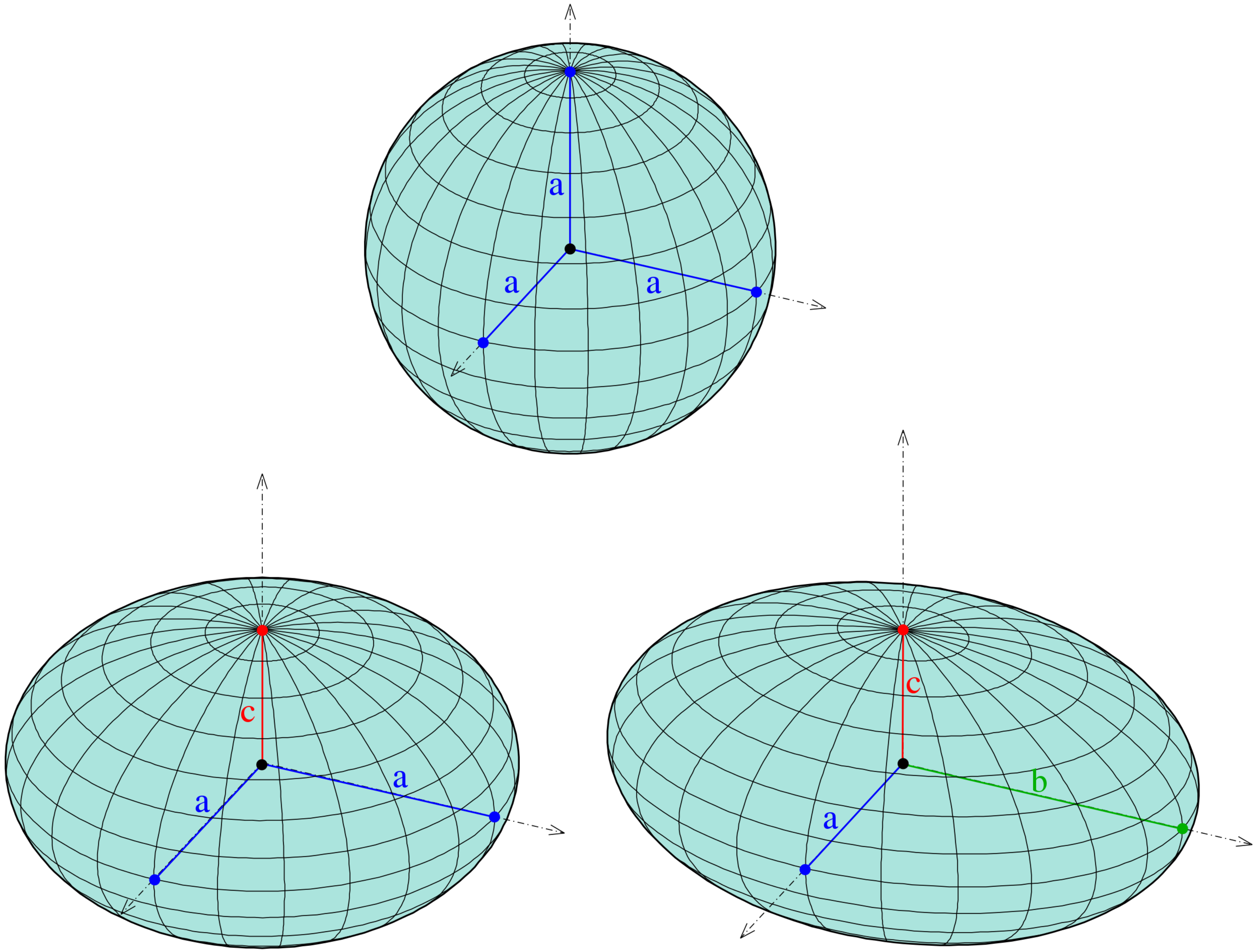



**???**

# Ellipsoids

Ellipsoids are spheres "stretched" in orthogonal directions called the **axes of symmetry** or the **principle axes.**

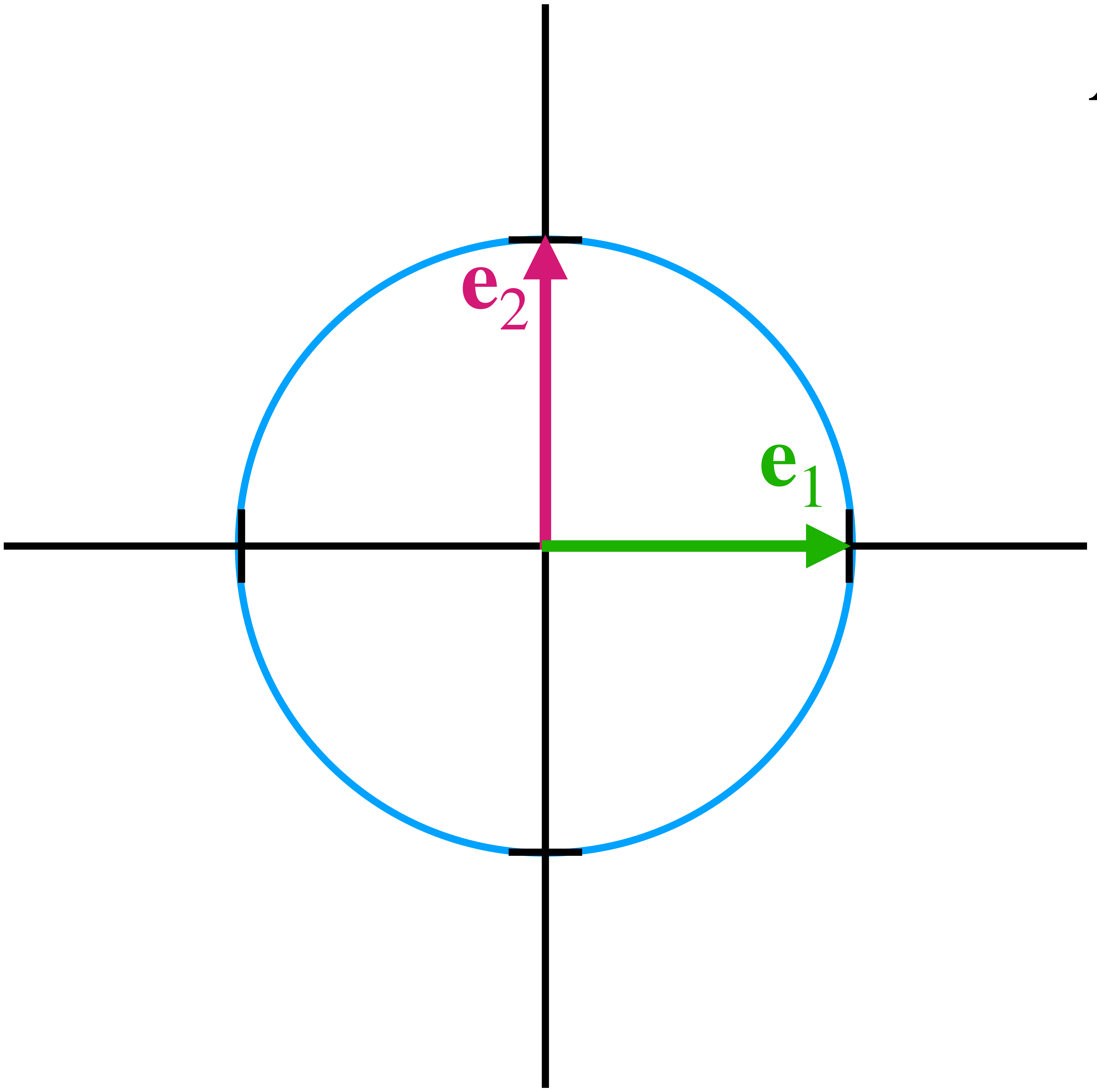
**Linear transformations maps <u>spheres</u> to <u>ellipsoids</u>.**

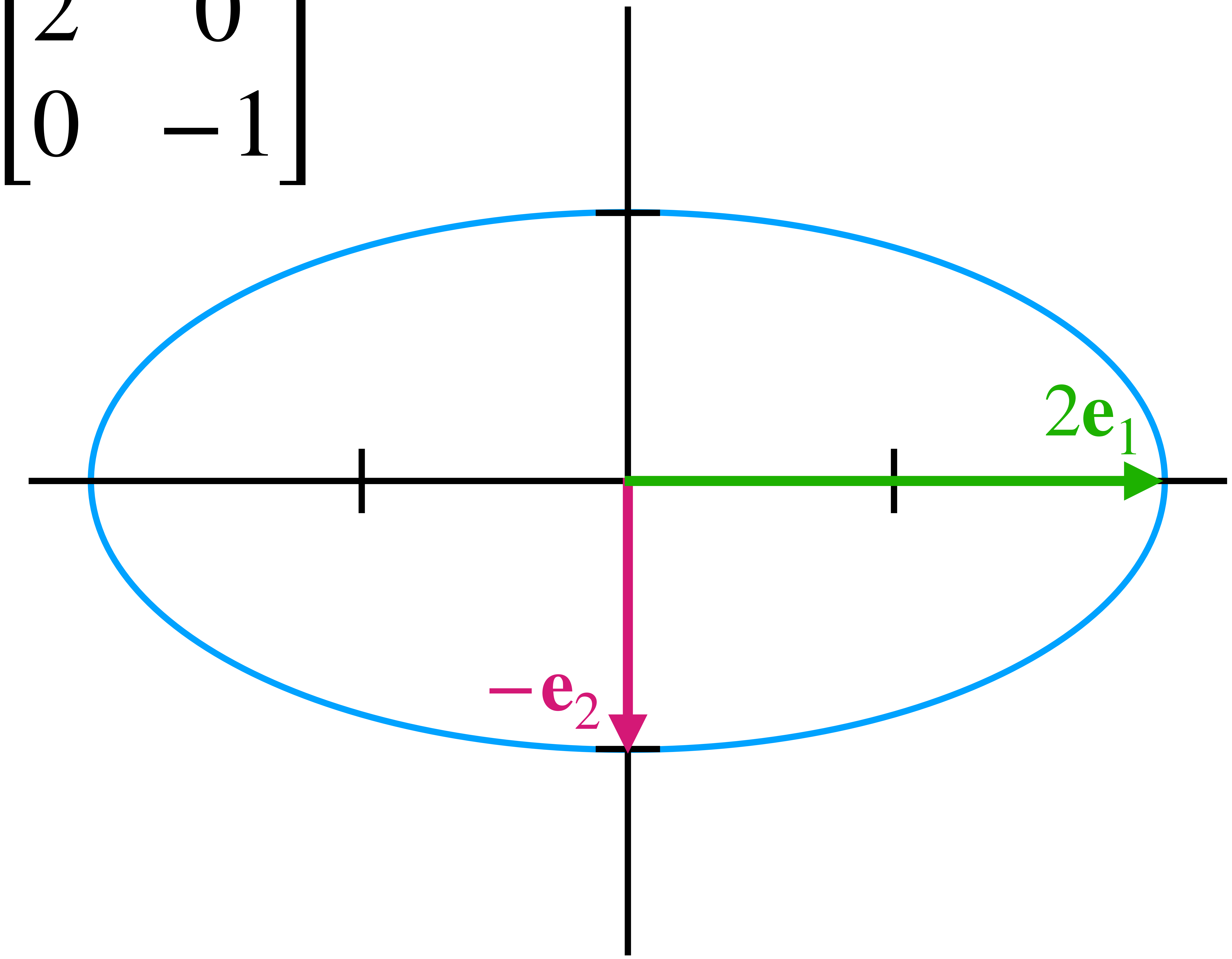



https://commons.wikimedia.org/wiki/File:Ellipsoide.svg

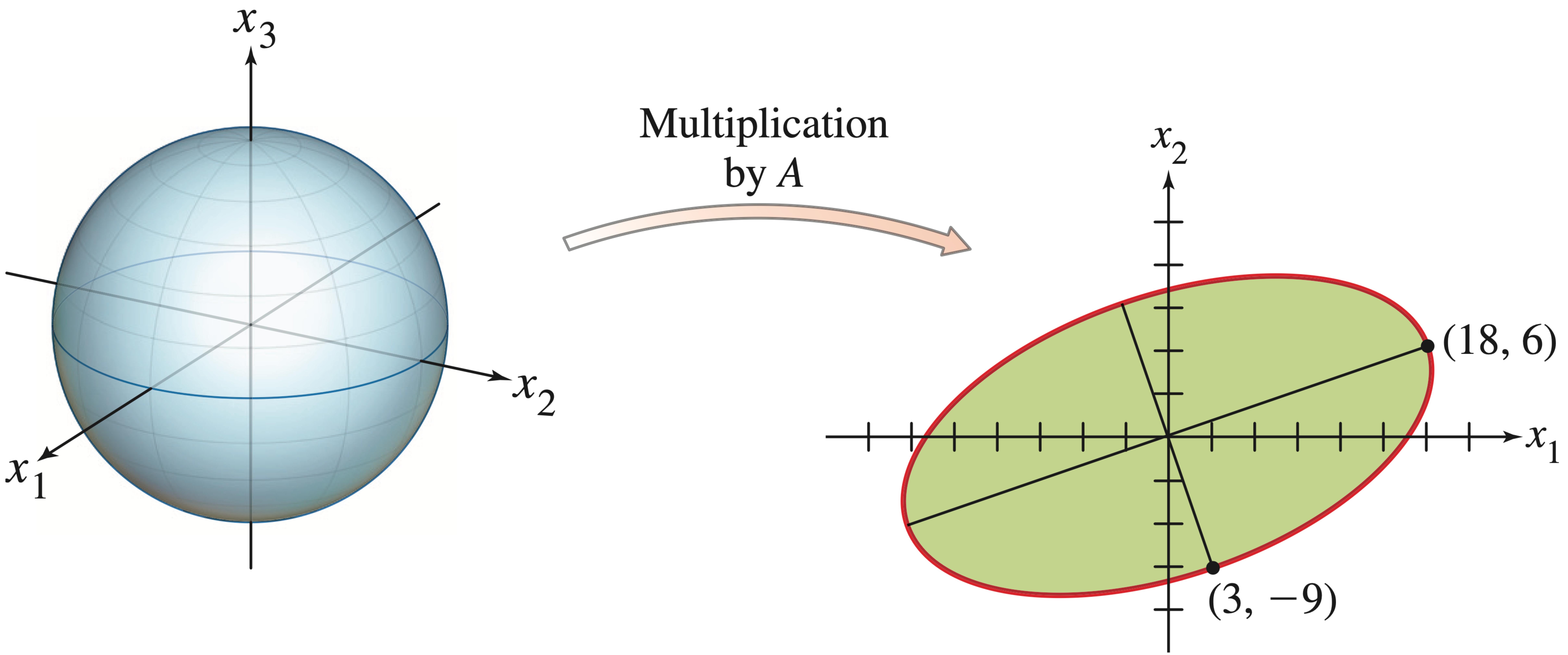
# Simple Example : Scaling Matrices

$$A = \begin{bmatrix} 2 & 0 \\ 0 & -1 \end{bmatrix}$$

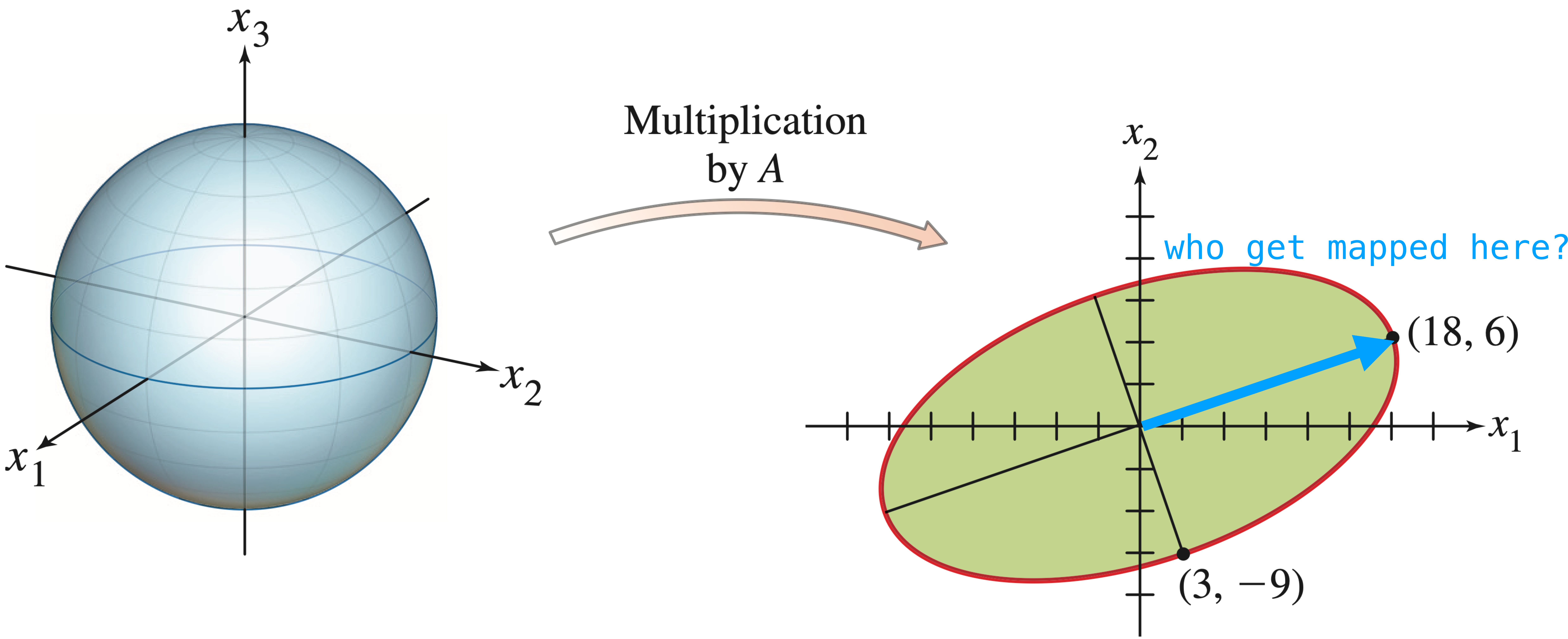


# The Picture

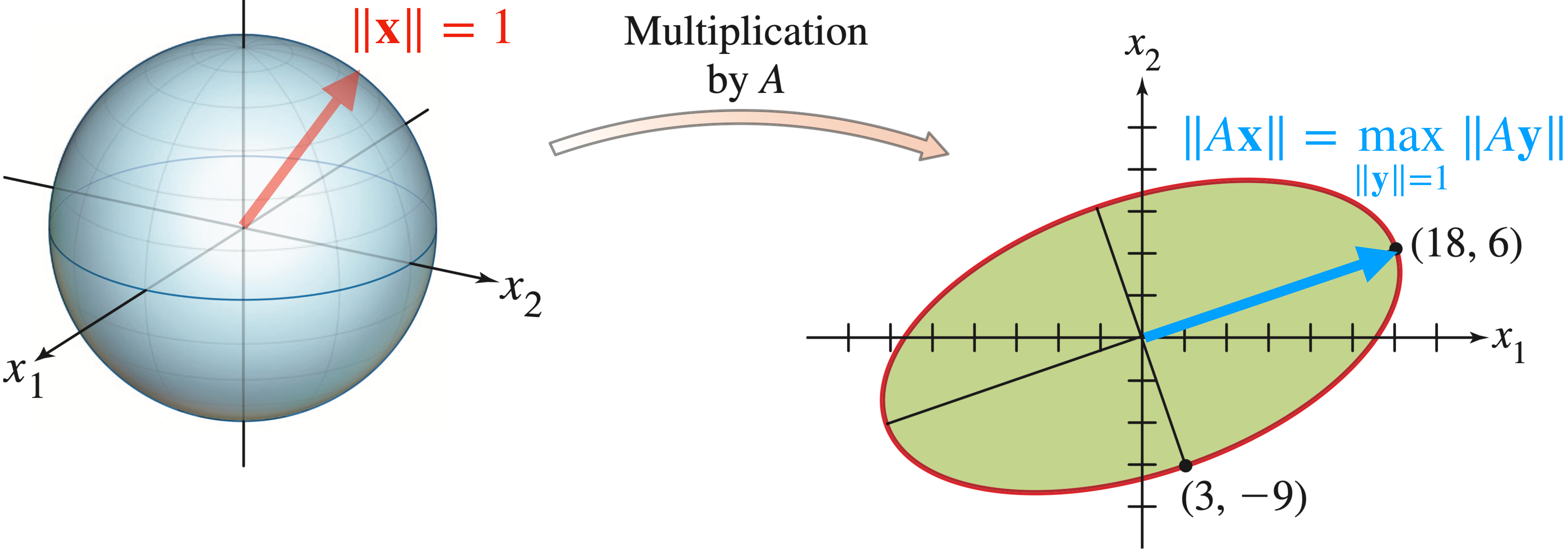


# The Picture

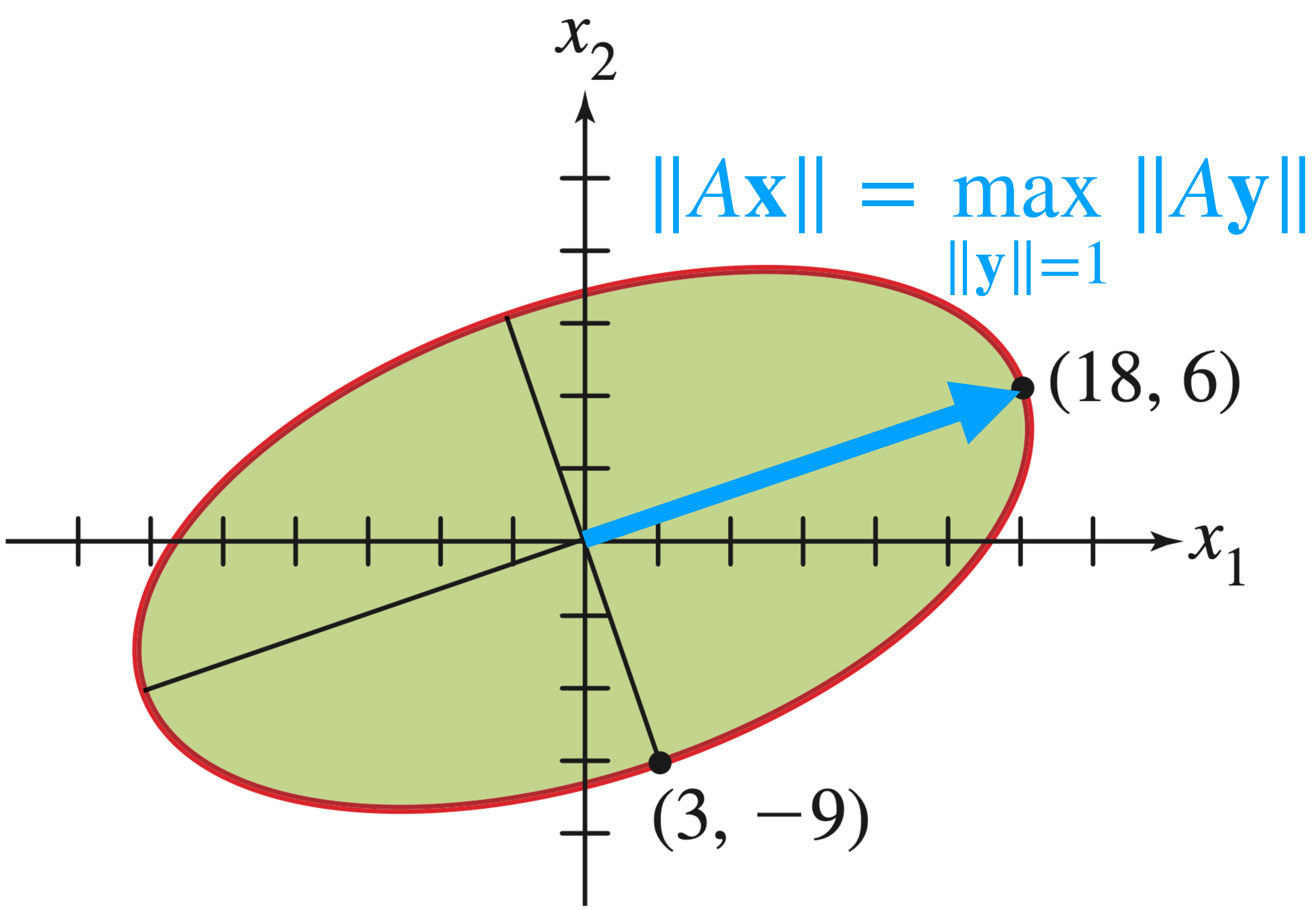


# The Picture

The longest end of the ellipse is the solution to a constrained optimization problem

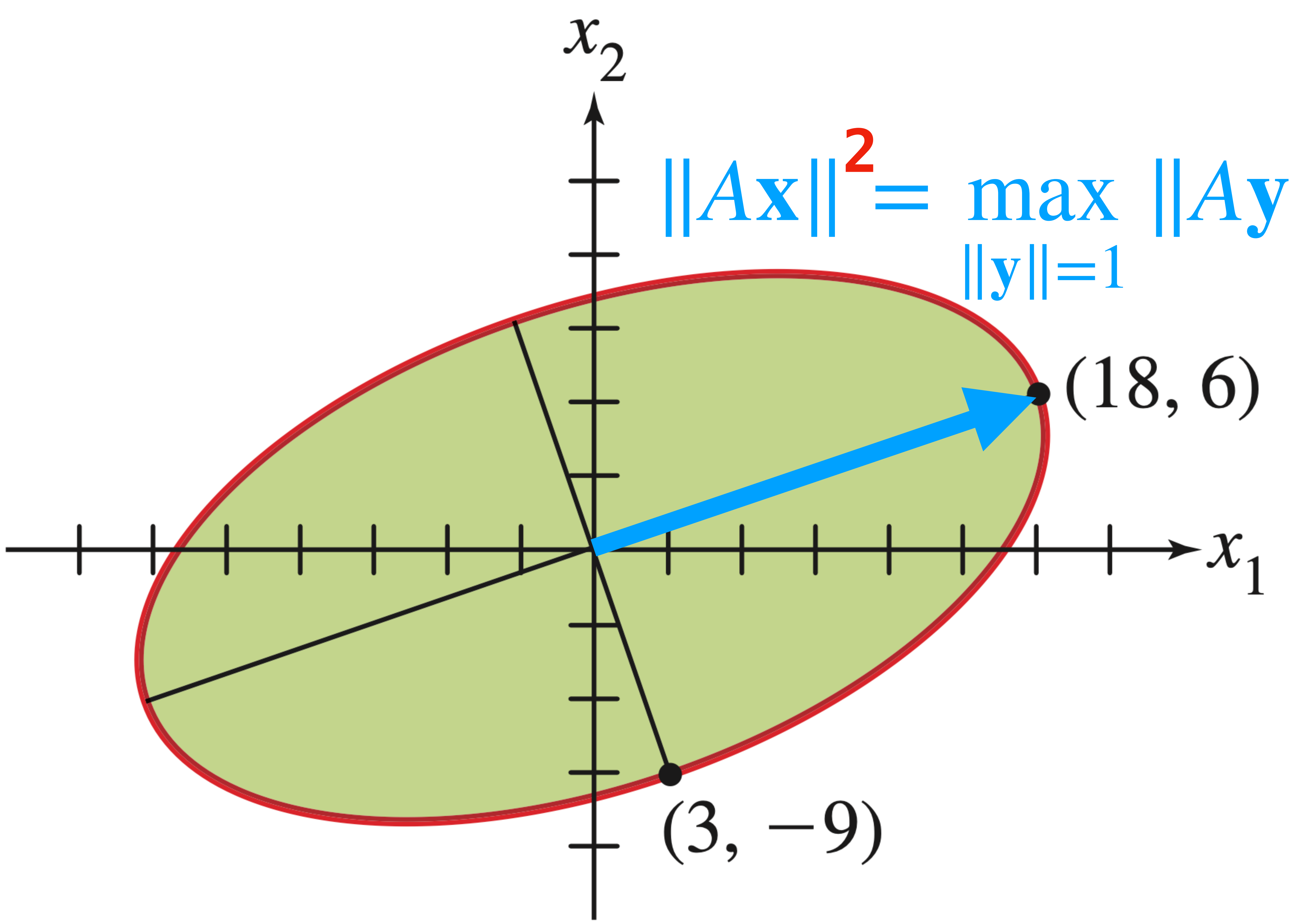


# The Picture




This is not a quadratic form...

# The Picture




$$\|A\mathbf{x}\|^2 = \max_{\|\mathbf{y}\|=1} \|A\mathbf{y}\|^2$$

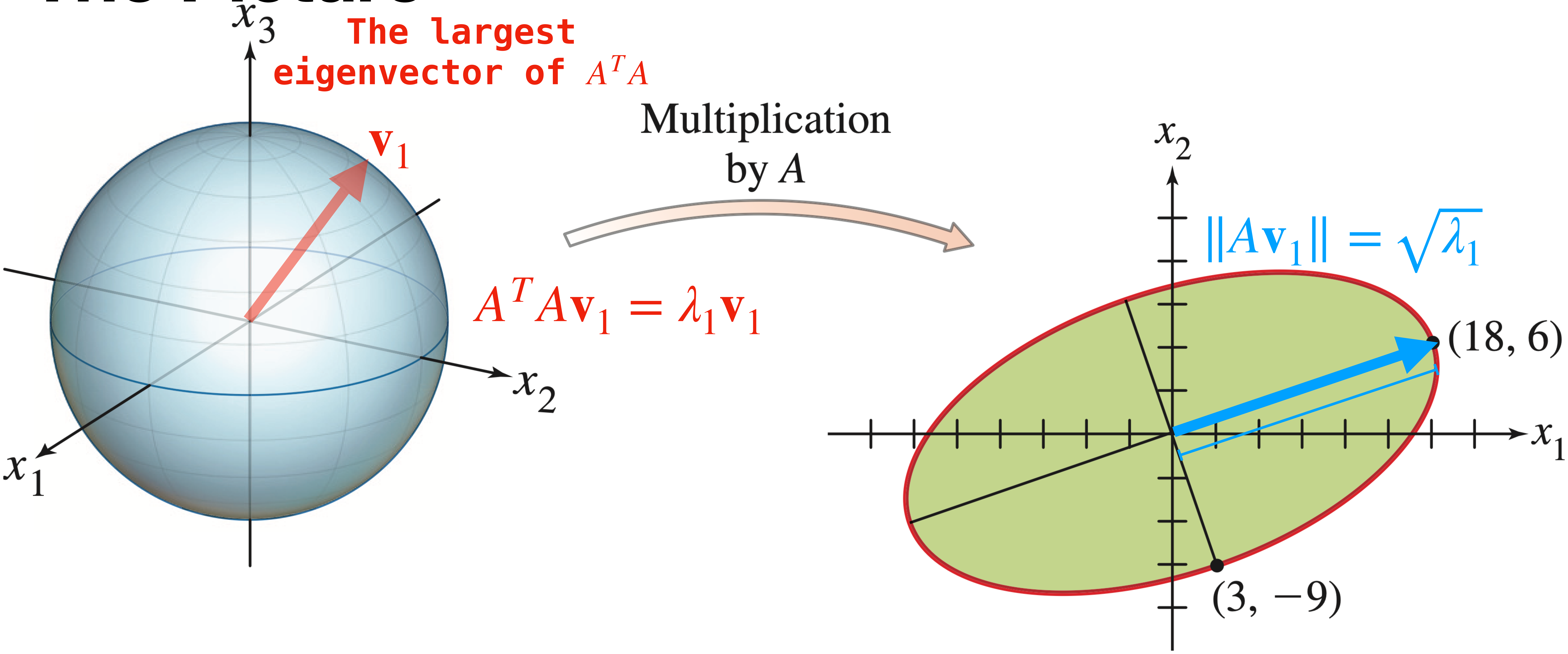
But this is.

This is not a quadratic form...

# A Quadratic Form

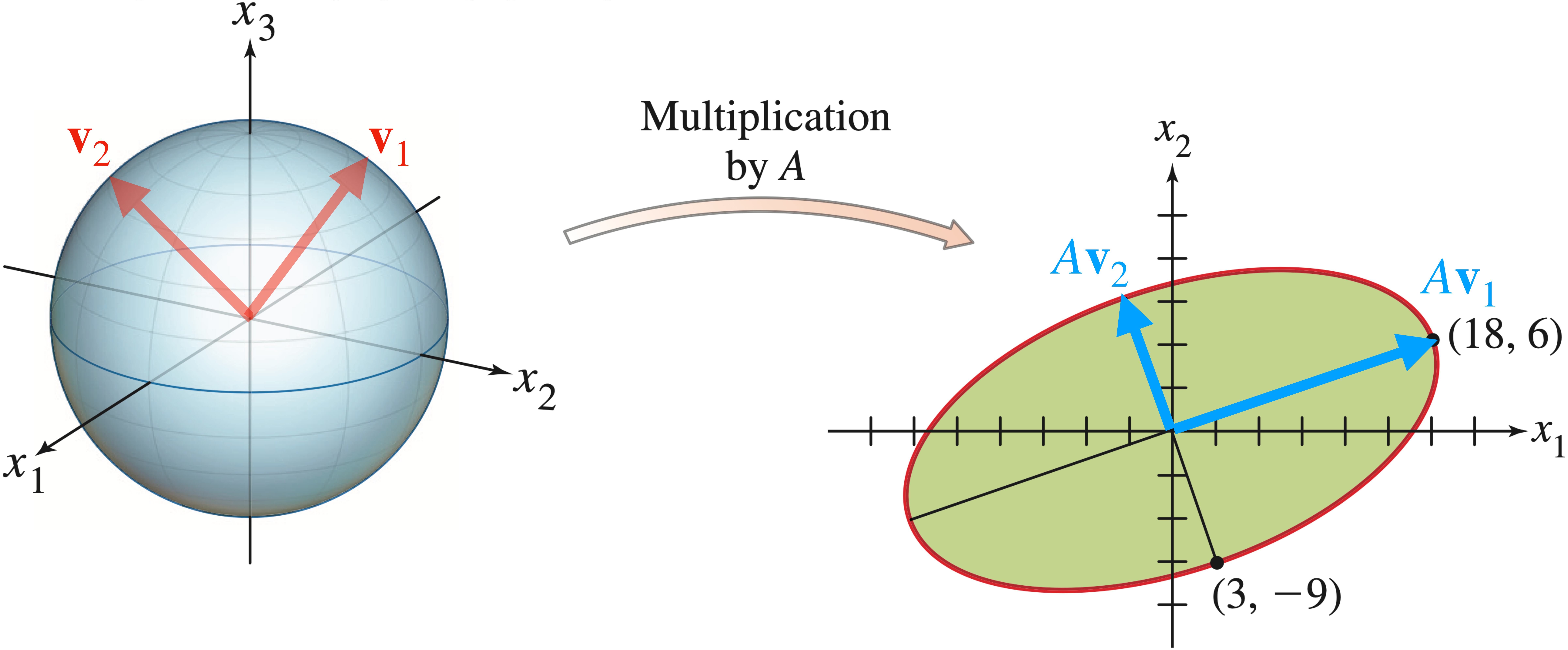
What does $\|A\mathbf{x}\|^2$ look like?:

# The Picture




$\mathbf{v}_1$ solves the constrained optimization problem.

# The "Influence" of $A$




$\mathbf{v}_1$ is "most affected" by $A$ and $\mathbf{v}_2$ is "least affected"

# Properties of $A^T A$

# Properties of $A^T A$

» It's symmetric.

# Properties of $A^T A$

» It's symmetric.

» So its orthogonally diagonalizable.

# Properties of $A^T A$

- It's symmetric.
- So its orthogonally diagonalizable.
- **There is an orthogonal basis of eigenvectors.**

# Properties of $A^T A$

- It's symmetric.
- So its orthogonally diagonalizable.
- **There is an orthogonal basis of eigenvectors.**
- It's eigenvalues are nonnegative.

# Properties of $A^T A$

- It's symmetric.
- So its orthogonally diagonalizable.
- **There is an orthogonal basis of eigenvectors.**
- It's eigenvalues are nonnegative.
- **It's positive semidefinite.**

# Singular Values

**Definition.** For an $m \times n$ matrix $A$, the **singular values** of $A$ are the $n$ values

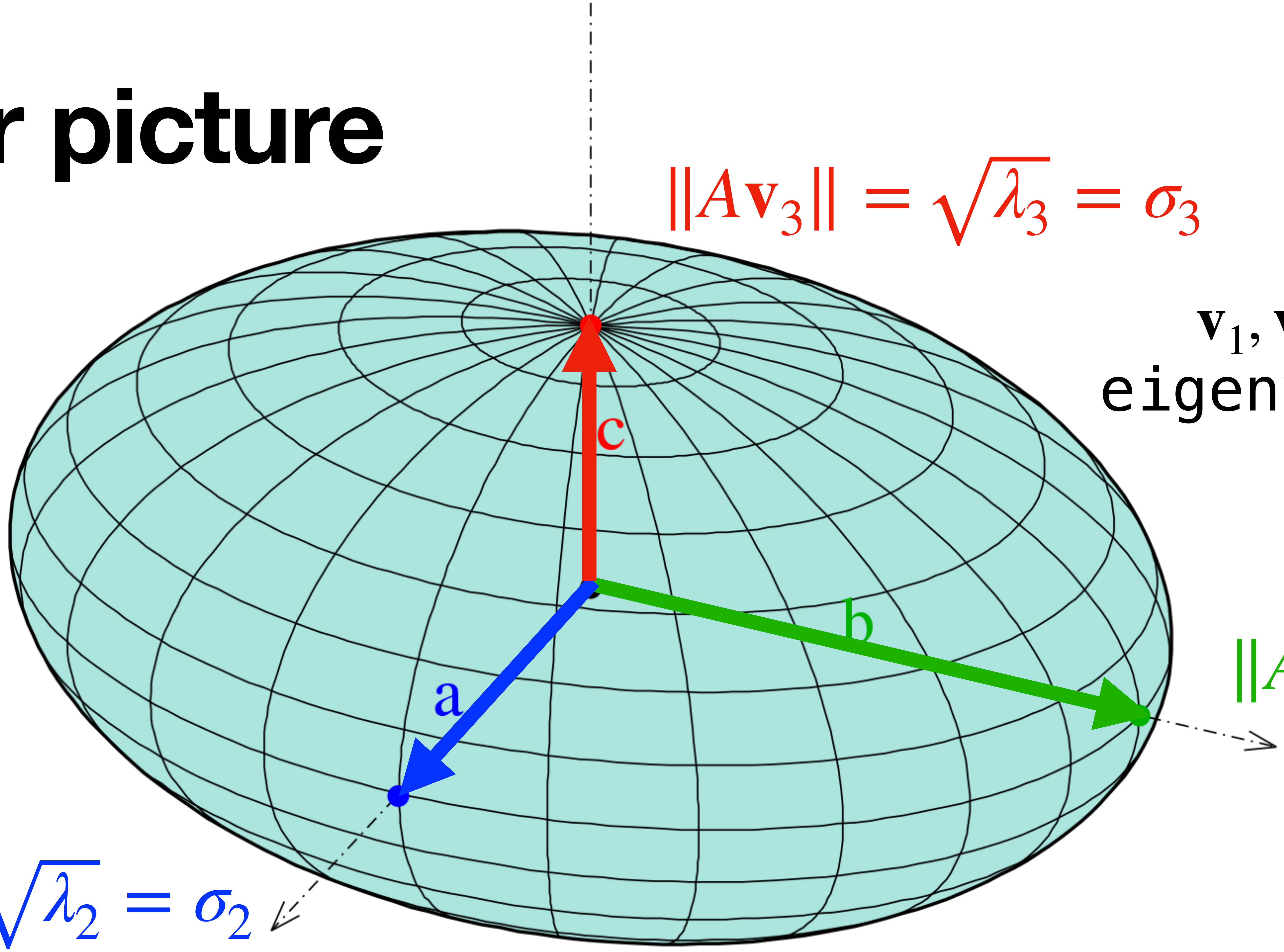
$$\sigma_1 \geq \sigma_2 \ldots \geq \sigma_n \geq 0$$

where $\sigma_i = \sqrt{\lambda_i}$ and $\lambda_i$ is an eigenvalue of $A^T A$.

# Another picture

$\|A\mathbf{v}_3\| = \sqrt{\lambda_3} = \sigma_3$

$\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ are the eigenvectors of $A^TA$




$\|A\mathbf{v}_1\| = \sqrt{\lambda_1} = \sigma_1$

$\|A\mathbf{v}_2\| = \sqrt{\lambda_2} = \sigma_2$

The **singular values** are the lengths of the *axes of symmetry* of the ellipsoid after transforming the unit sphere.

https://commons.wikimedia.org/wiki/File:Ellipsoide.svg

Every $m \times n$ matrix transforms the unit $m$–sphere into an $n$–ellipsoid.

So <u>every</u> $m \times n$ matrix has $n$ singular values.

# What else can we say?

Let $\mathbf{v}_1, \ldots, \mathbf{v}_n$ be an **orthogonal** eigenbasis of $\mathbb{R}^n$ for $A^TA$ and suppose $A$ has $r$ nonzero singular values.

**Theorem.** $A\mathbf{v}_1, \ldots, A\mathbf{v}_r$ is an orthogonal basis of $\text{Col}(A)$.

# What else can we say?

Let $\mathbf{v}_1, \ldots, \mathbf{v}_n$ be an **orthogonal** eigenbasis of $\mathbb{R}^n$ for $A^TA$ and suppose $A$ has $r$ nonzero singular values.

**Theorem.** $A\mathbf{v}_1, \ldots, A\mathbf{v}_r$ is an orthogonal basis of $\text{Col}(A)$.

**This is the most important theorem for SVD.**

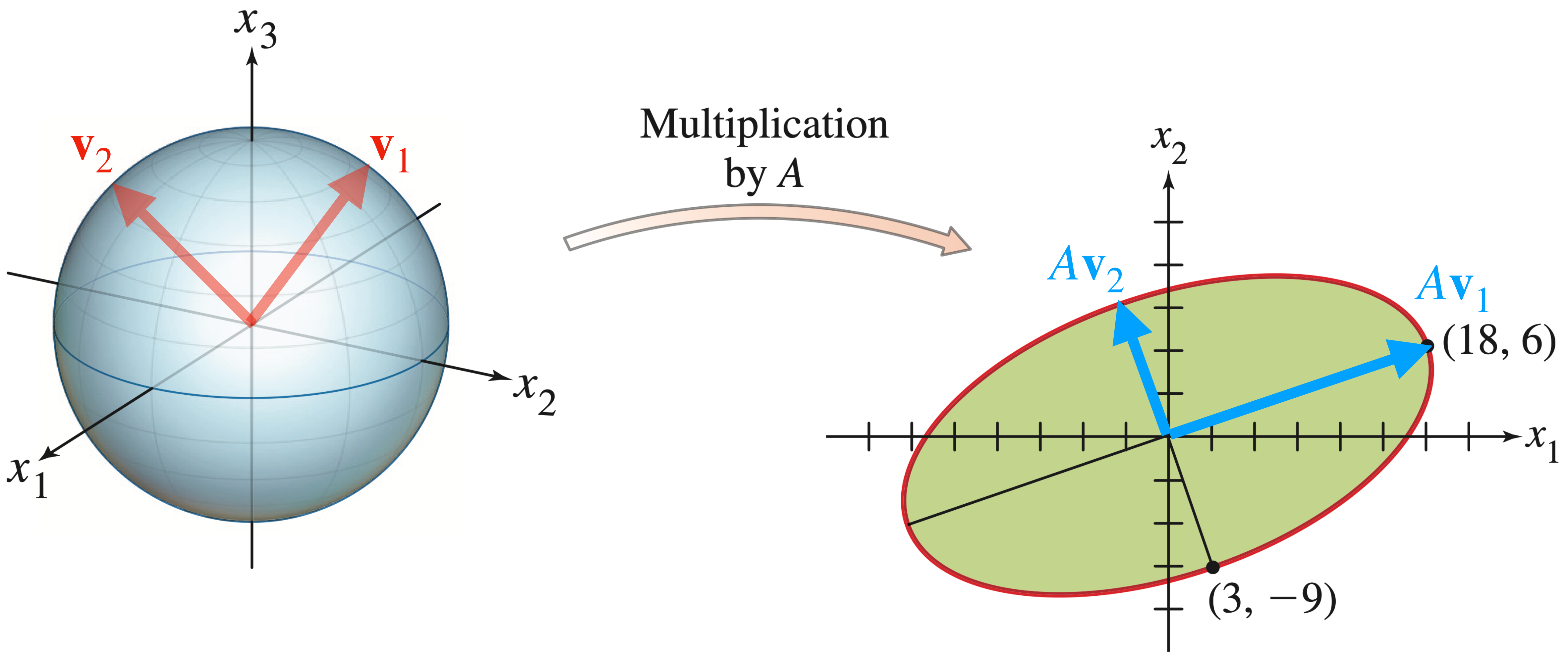
# Verifying it

Let's show $A\mathbf{v}_1, \ldots, A\mathbf{v}_r$ are linearly independent:

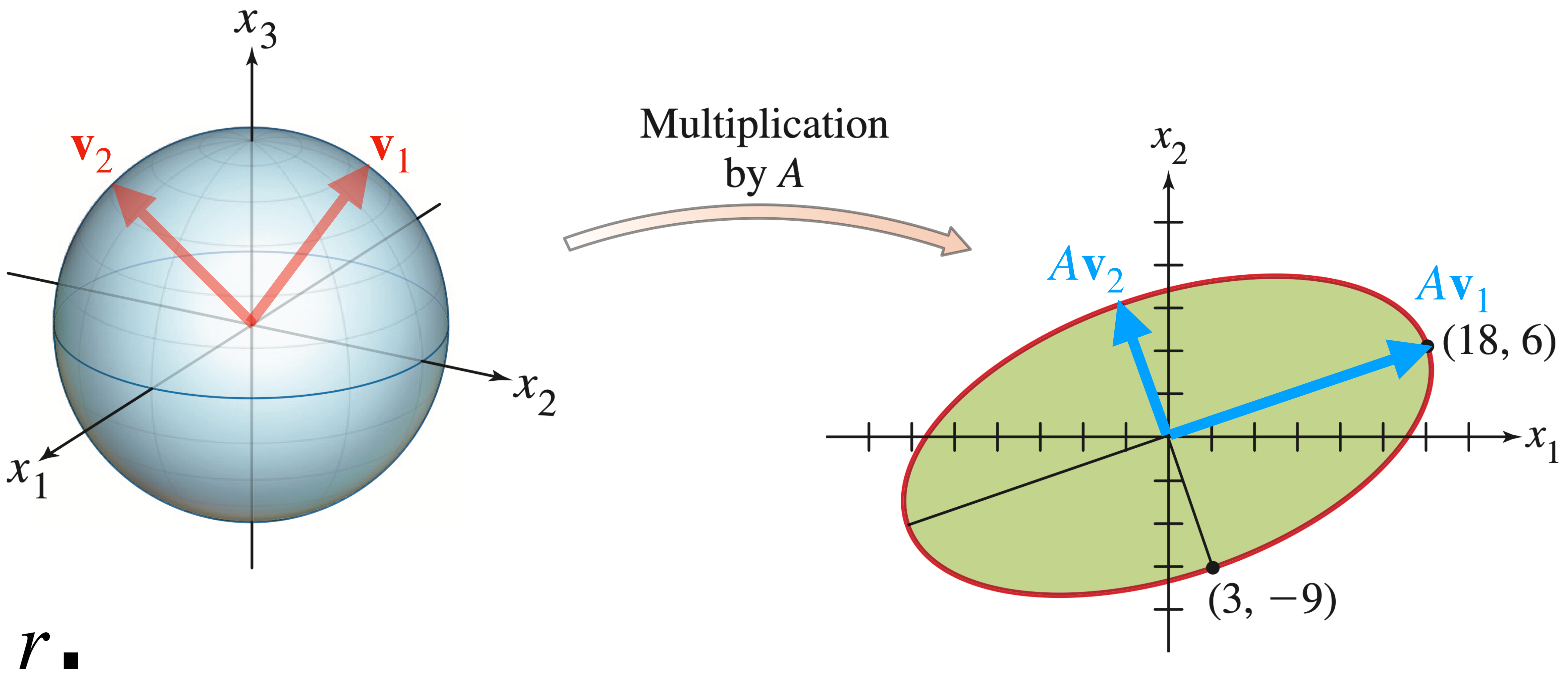
# Verifying it

Let's show $A\mathbf{v}_1, \ldots, A\mathbf{v}_r$ span $\text{Col}(A)$:

# Putting it all together

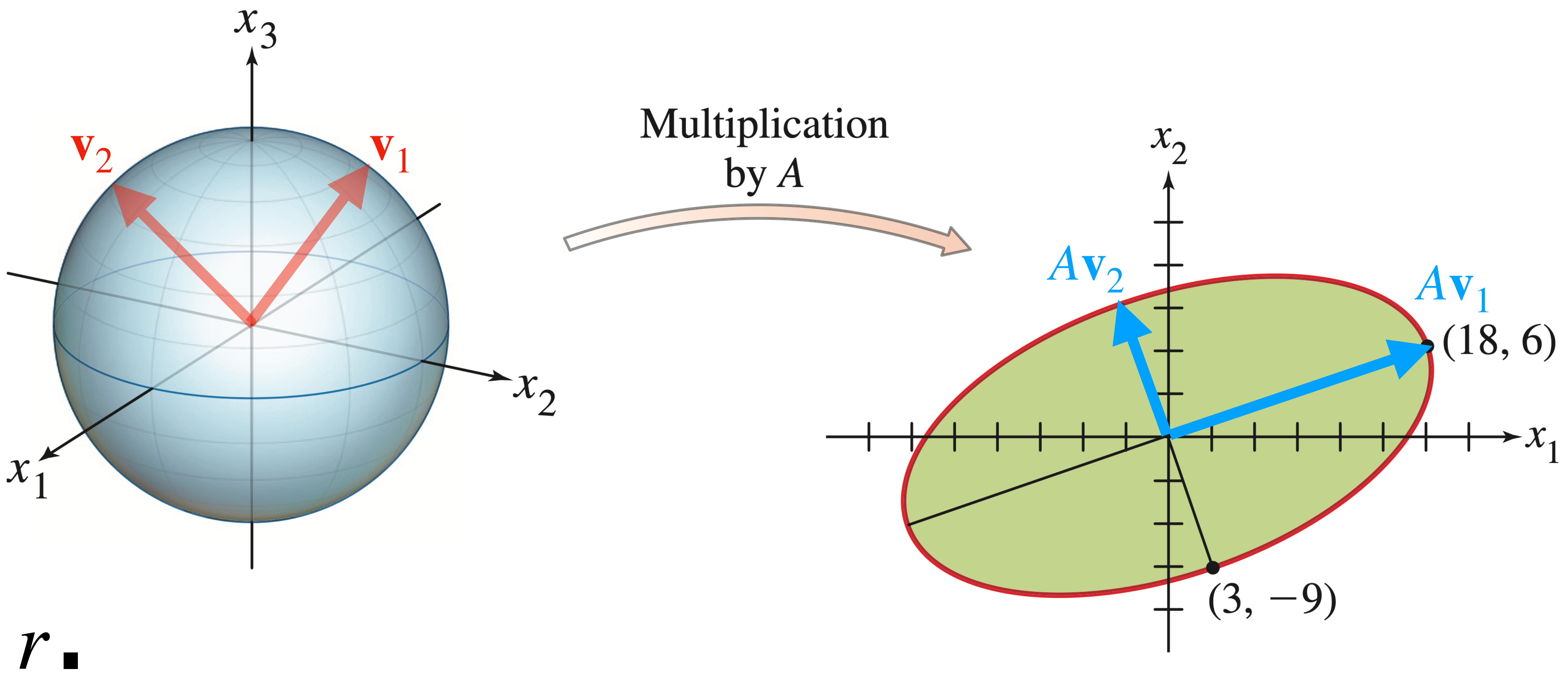


# Putting it all together




Let $A$ be an $m \times n$ matrix of rank $r$.

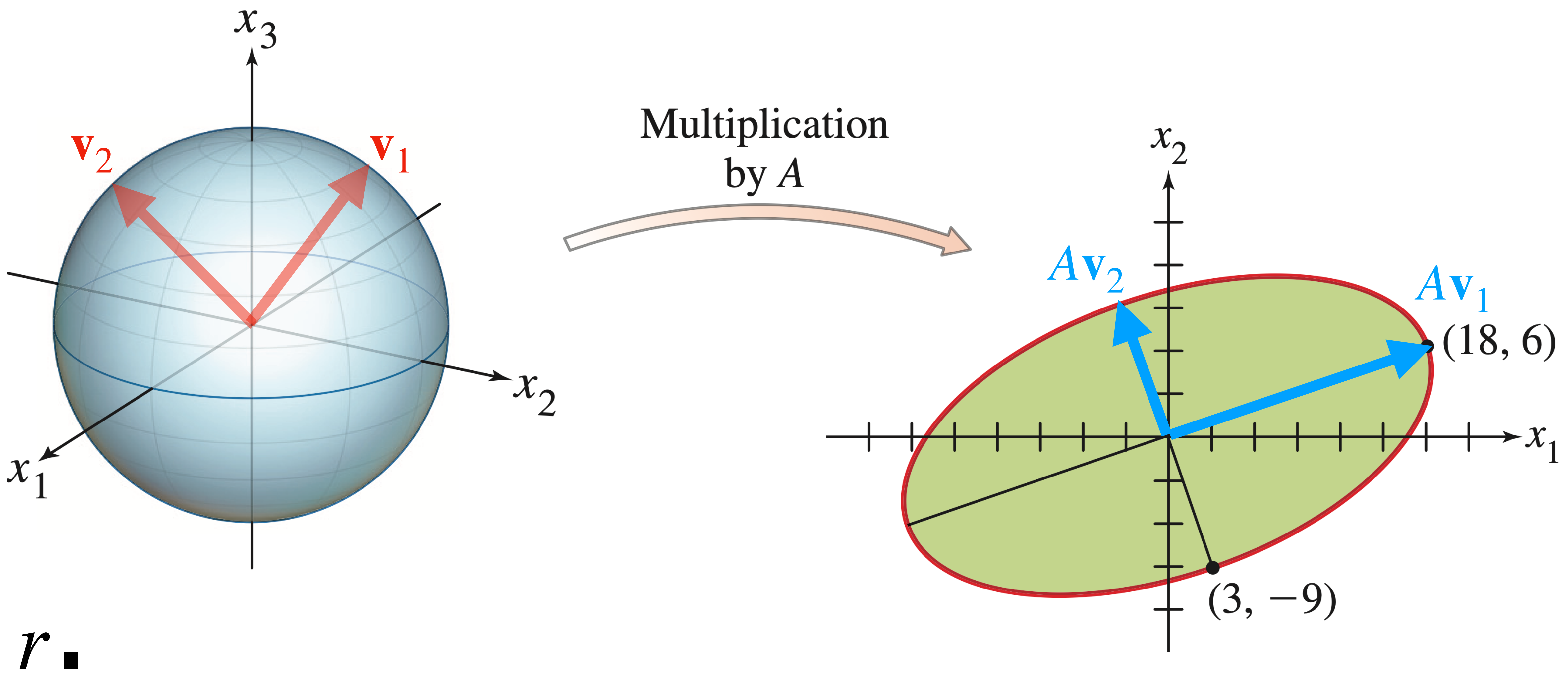
# Putting it all together




Let $A$ be an $m \times n$ matrix of rank $r$.

<u>What we know:</u>

# Putting it all together

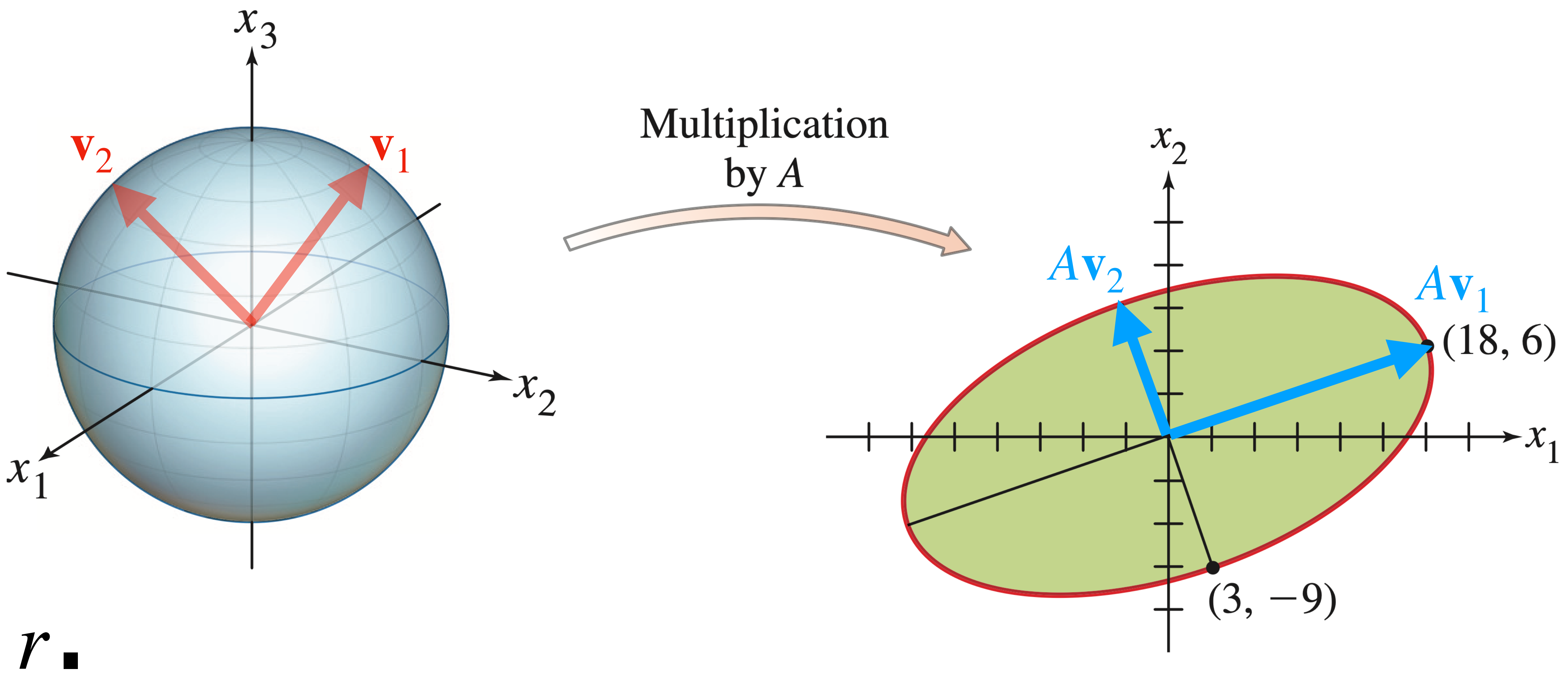



Let $A$ be an $m \times n$ matrix of rank $r$.

<u>What we know:</u>

» We can find orthonormal vectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$ in $\mathbb{R}^n$ such that $A\mathbf{v}_1, \ldots, A\mathbf{v}_r$ in $\mathbb{R}^m$ form an orthogonal basis for Col$(A)$.

# Putting it all together



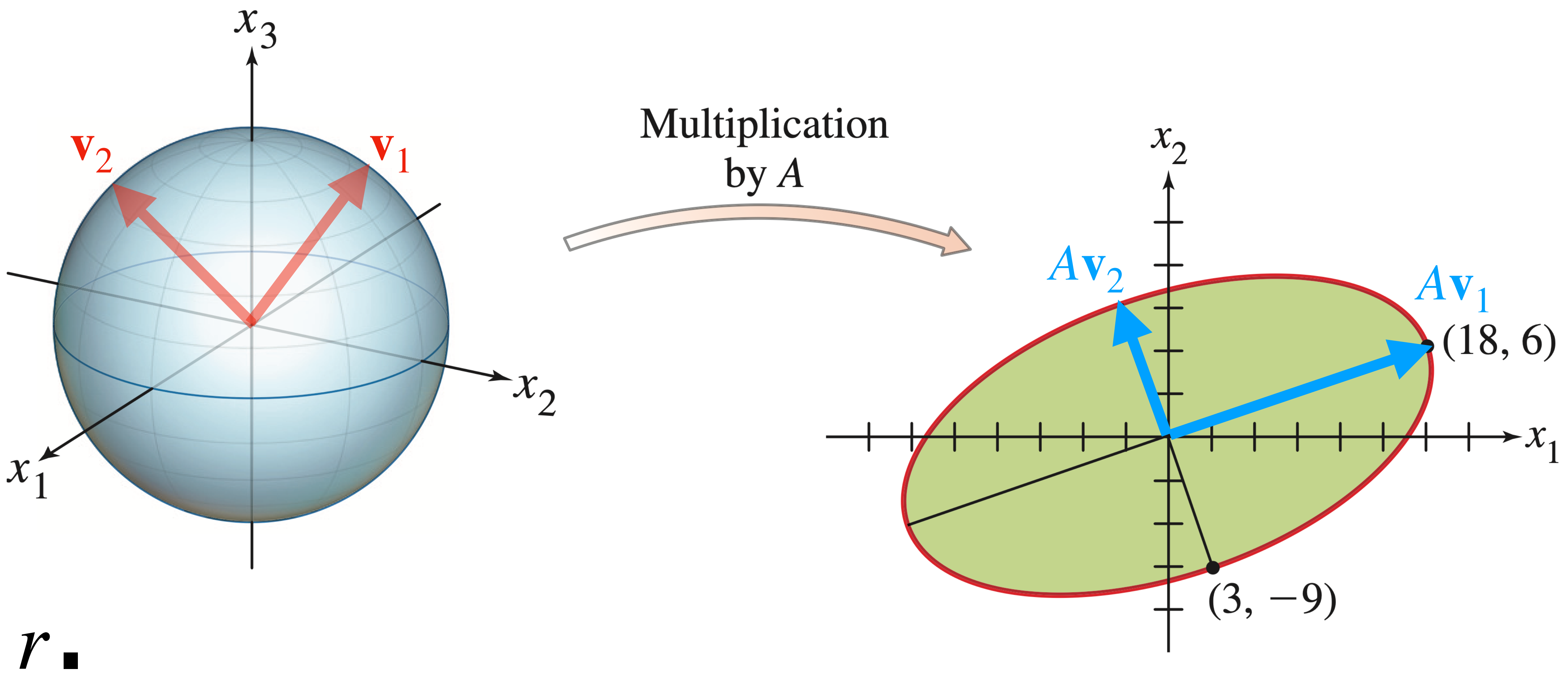

Let $A$ be an $m \times n$ matrix of rank $r$.

<u>What we know:</u>

» We can find orthonormal vectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$ in $\mathbb{R}^n$ such that $A\mathbf{v}_1, \ldots, A\mathbf{v}_r$ in $\mathbb{R}^m$ form an orthogonal basis for $\mathrm{Col}(A)$.

» So if we take $\mathbf{u}_i = \dfrac{A\mathbf{v}_i}{\|A\mathbf{v}_i\|}$, we get an **orthonormal** basis of $\mathrm{Col}(A)$

# Putting it all together




Let $A$ be an $m \times n$ matrix of rank $r$.

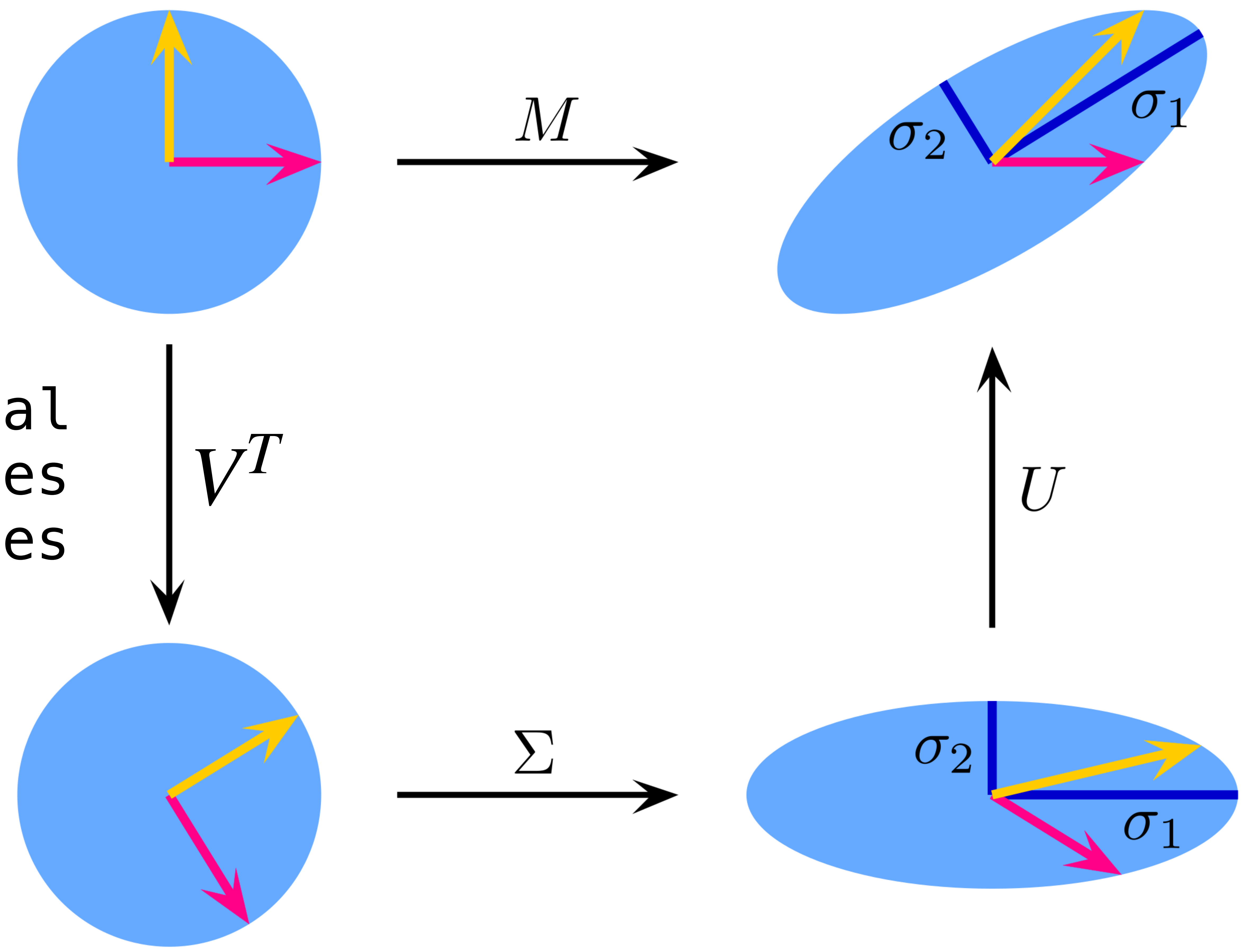
<u>What we know:</u>

» We can find orthonormal vectors $\mathbf{v}_1, \ldots, \mathbf{v}_r$ in $\mathbb{R}^n$ such that $A\mathbf{v}_1, \ldots, A\mathbf{v}_r$ in $\mathbb{R}^m$ form an orthogonal basis for Col$(A)$.

» So if we take $\mathbf{u}_i = \dfrac{A\mathbf{v}_i}{\|A\mathbf{v}_i\|}$, we get an **orthonormal** basis of Col$(A)$

» And we can extend this to $\mathbf{u}_1, \ldots, \mathbf{u}_m$ an orthonormal basis of $\mathbb{R}^m$ (via Gram-Schmidt).

# High Level View of the Decomposition




<u>recall</u>: Orthogonal matrices preserves lengths and angles

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

https://commons.wikimedia.org/wiki/File:Singular-Value-Decomposition.svg

# The Important Equality

$$\mathbf{u}_i = \frac{A\mathbf{v}_i}{\|A\mathbf{v}_i\|}$$

$$A\mathbf{v}_i = \|A\mathbf{v}_i\|\mathbf{u}_i = \sigma_i\mathbf{u}_i$$

# The Important Equality

$$\mathbf{u}_i = \frac{A\mathbf{v}_i}{\|A\mathbf{v}_i\|}$$

$$A\mathbf{v}_i = \|A\mathbf{v}_i\|\mathbf{u}_i = \sigma_i\mathbf{u}_i$$

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

# The Important Equality

$$\mathbf{u}_i = \frac{A\mathbf{v}_i}{\|A\mathbf{v}_i\|}$$

$$A\mathbf{v}_i = \|A\mathbf{v}_i\|\mathbf{u}_i = \sigma_i\mathbf{u}_i$$

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

What happens when we write this in matrix form?

# The Important Equality

$$A[\mathbf{v}_1 \ \ \dots \ \ \mathbf{v}_n] = [\sigma_1\mathbf{u}_1 \ \ \dots\sigma_n\mathbf{u}_n]$$

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

# The Important Equality

$$A[\mathbf{v}_1 \ \ \dots \ \ \mathbf{v}_n] = [\sigma_1 \mathbf{u}_1 \ \ \dots \sigma_n \mathbf{u}_n]$$

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

Let's take $V = [\mathbf{v}_1 \ \ \dots \ \ \mathbf{v}_n]$ and $U = [\mathbf{u}_1 \ \ \dots \ \ \mathbf{u}_m]$ and

# The Important Equality

$$A[\mathbf{v}_1 \ \dots \ \mathbf{v}_n] = [\sigma_1\mathbf{u}_1 \ \dots \sigma_n\mathbf{u}_n]$$

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

Let's take $V = [\mathbf{v}_1 \ \dots \ \mathbf{v}_n]$ and $U = [\mathbf{u}_1 \ \dots \ \mathbf{u}_m]$ and

$m > n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}$$

or

$m < n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_m & 0 & \dots & 0 \end{bmatrix}$$

or

$m = n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \end{bmatrix}$$

# The Important Equality

$$A[\mathbf{v}_1 \ \ \dots \ \ \mathbf{v}_n] = [\sigma_1\mathbf{u}_1 \ \ \dots\sigma_n\mathbf{u}_n]$$

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

Let's take $V = [\mathbf{v}_1 \ \dots \ \mathbf{v}_n]$ and $U = [\mathbf{u}_1 \ \dots \ \mathbf{u}_m]$ and

**remember: $U$ is orthonormal**

$m > n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}$$

or

$m < n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_m & 0 & \dots & 0 \end{bmatrix}$$

or

$m = n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \end{bmatrix}$$

# The Important Equality

$$AV = U\Sigma$$

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

Let's take $V = [\mathbf{v}_1 \ \dots \ \mathbf{v}_n]$ and $U = [\mathbf{u}_1 \ \dots \ \mathbf{u}_m]$ and

**remember: $U$ is orthonormal**

$m > n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}$$

$m < n$

or $$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_m & 0 & \dots & 0 \end{bmatrix}$$

$m = n$

or $$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \end{bmatrix}$$

# The Important Equality

$$\underset{m\times n}{A}\,\underset{n\times n}{V} = \underset{m\times m}{U}\,\underset{m\times n}{\Sigma}$$

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

Let's take $V = [\mathbf{v}_1 \ \dots \ \mathbf{v}_n]$ and $U = [\mathbf{u}_1 \ \dots \ \mathbf{u}_m]$ and

**remember: $U$ is orthonormal**

$m > n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}$$

or ($m < n$)

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_m & 0 & \dots & 0 \end{bmatrix}$$

or ($m = n$)

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \end{bmatrix}$$

# The Important Equality

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

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

Let's take $V = [\mathbf{v}_1 \ \dots \ \mathbf{v}_n]$ and $U = [\mathbf{u}_1 \ \dots \ \mathbf{u}_m]$ and

**remember: $U$ is orthonormal**

$m > n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}$$

or

$m < n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_m & 0 & \dots & 0 \end{bmatrix}$$

or

$m = n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \end{bmatrix}$$

# The Important Equality

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

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

Let's take $V = [\mathbf{v}_1 \ \dots \ \mathbf{v}_n]$ and $U = [\mathbf{u}_1 \ \dots \ \mathbf{u}_m]$ and

**remember: $U$ is orthonormal**

$m > n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}$$

or

$m < n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_m & 0 & \dots & 0 \end{bmatrix}$$

or

$m = n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \end{bmatrix}$$

# The Important Equality

singular value decomposition

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

Remember that $\sigma_i = \sqrt{\lambda_i}$ is the singular value, which is the length $\|A\mathbf{v}_i\|$.

Let's take $V = [\mathbf{v}_1 \ \dots \ \mathbf{v}_n]$ and $U = [\mathbf{u}_1 \ \dots \ \mathbf{u}_m]$ and

remember: $U$ is orthonormal

$m > n$

$$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \\ 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{bmatrix}$$

$m < n$

or $$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_m & 0 & \dots & 0 \end{bmatrix}$$

$m = n$

or $$\Sigma = \begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_n \end{bmatrix}$$

# Singular Value Decomposition

**Theorem.** For a $m \times n$ matrix $A$, there are *orthogonal* matrices $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ such that

$$A = \underbrace{U}_{m \times m} \underbrace{\Sigma}_{m \times n} \underbrace{V^T}_{n \times n}$$

where diagonal entries* of $\Sigma$ are $\sigma_1, \ldots, \sigma_n$ the singular values of $A$.

* these are diagonal entries in a non-square matrix.

# Singular Value Decomposition

**Theorem.** For a $m \times n$ matrix $A$, there are *orthogonal* matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that

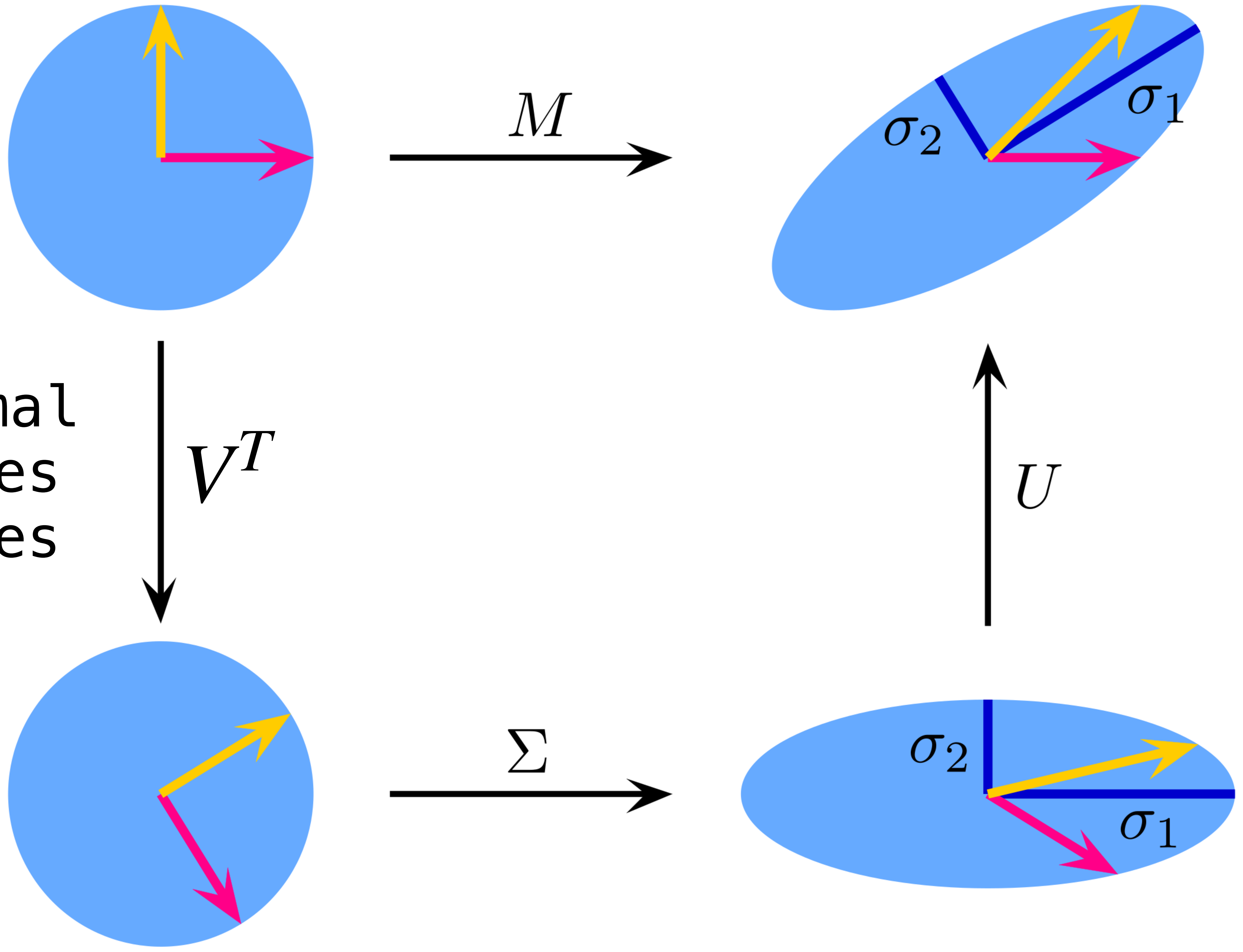
left singular vectors — right singular vectors

$$\underbrace{A = \overset{m\times m}{U} \underset{m\times n}{\Sigma} \overset{n\times n}{V^T}}$$

where diagonal entries* of $\Sigma$ are $\sigma_1, \ldots, \sigma_n$ the singular values of $A$.

* these are diagonal entries in a non-square matrix.

# The Picture (Again)




recall: Orthonormal matrices preserves lengths and angles

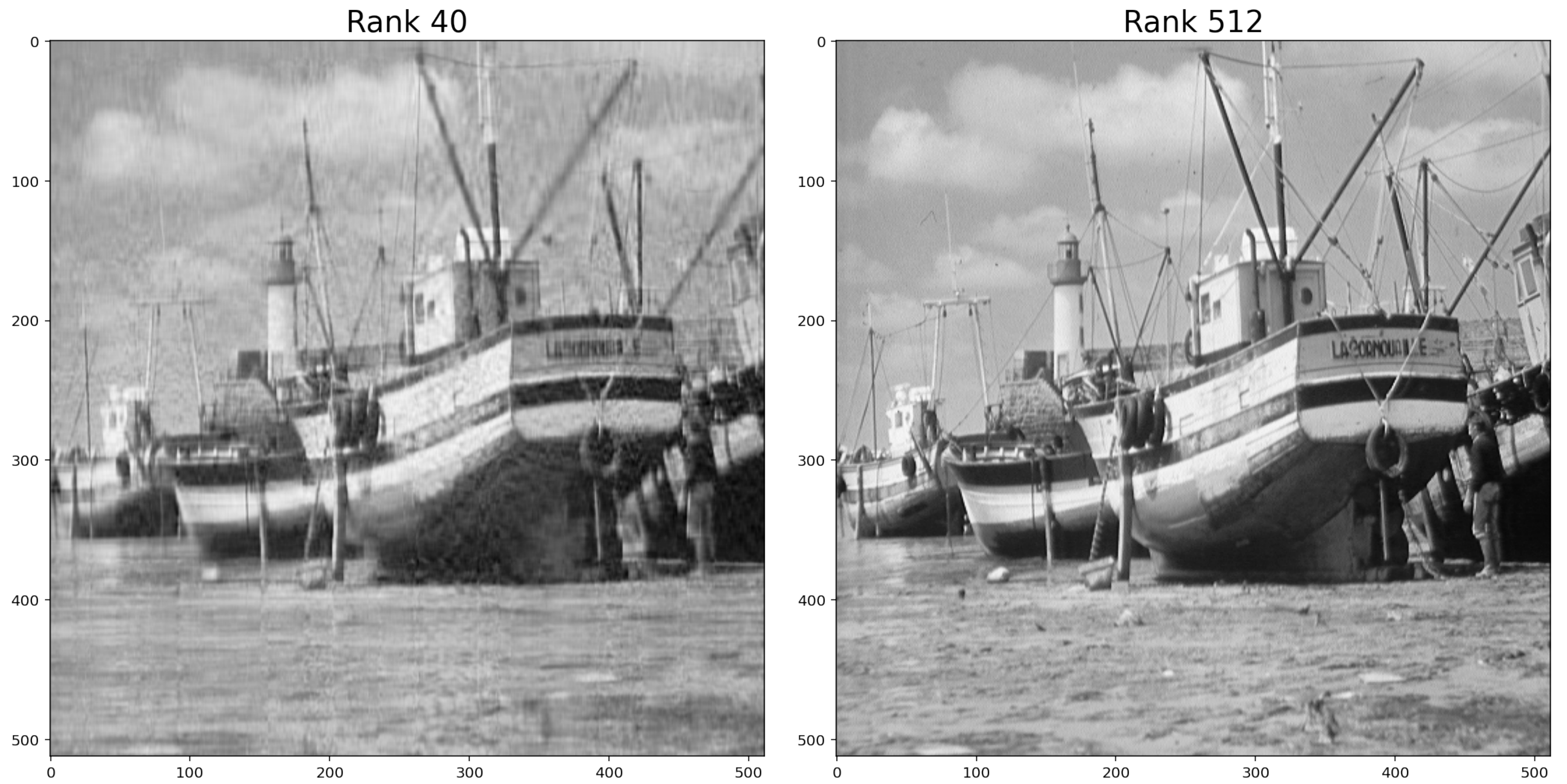
$$M = U \cdot \Sigma \cdot V^T$$

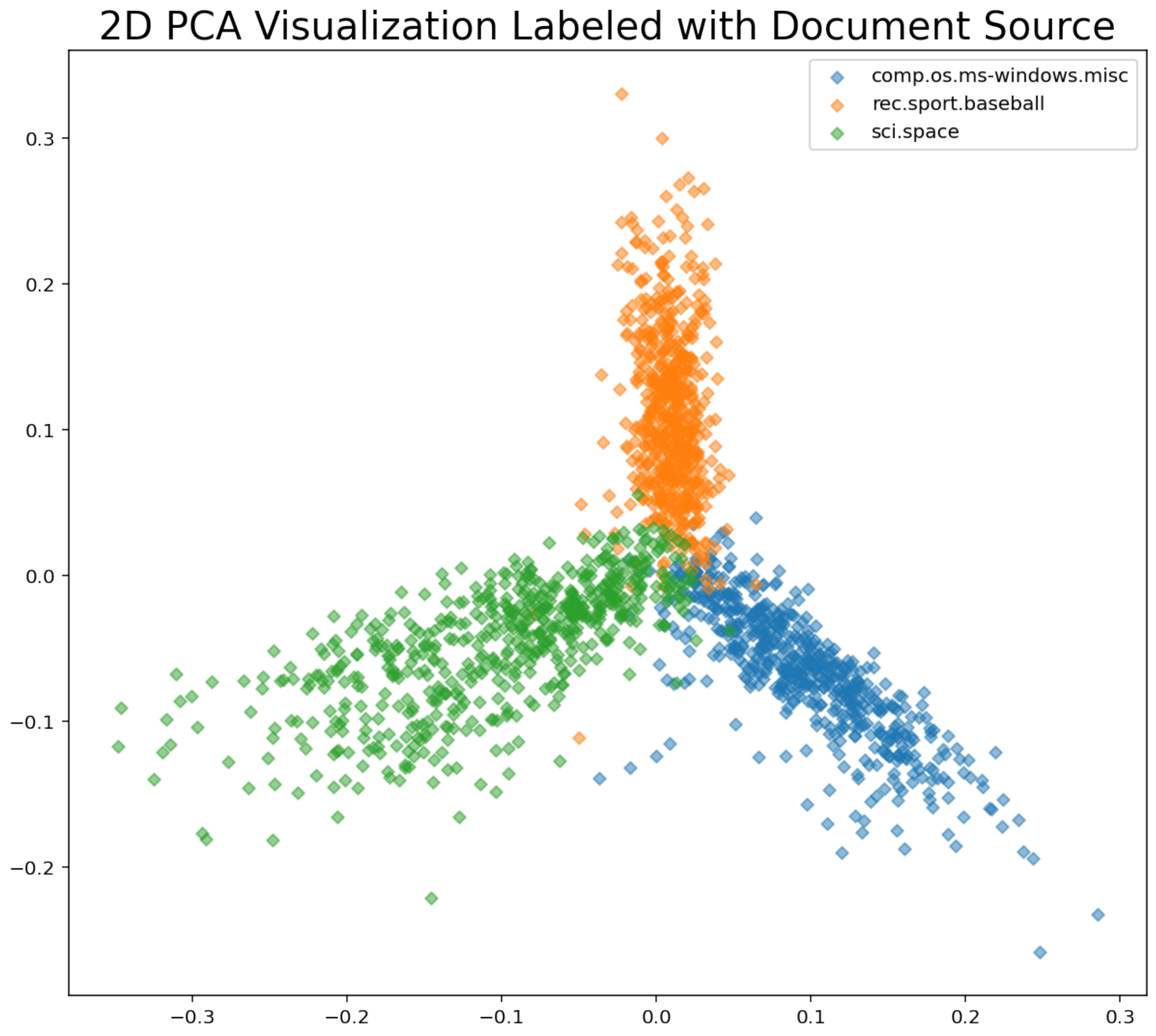
https://commons.wikimedia.org/wiki/File:Singular-Value-Decomposition.svg

# What's next?
# A couple final thoughts

# Applications of SVD

image compression



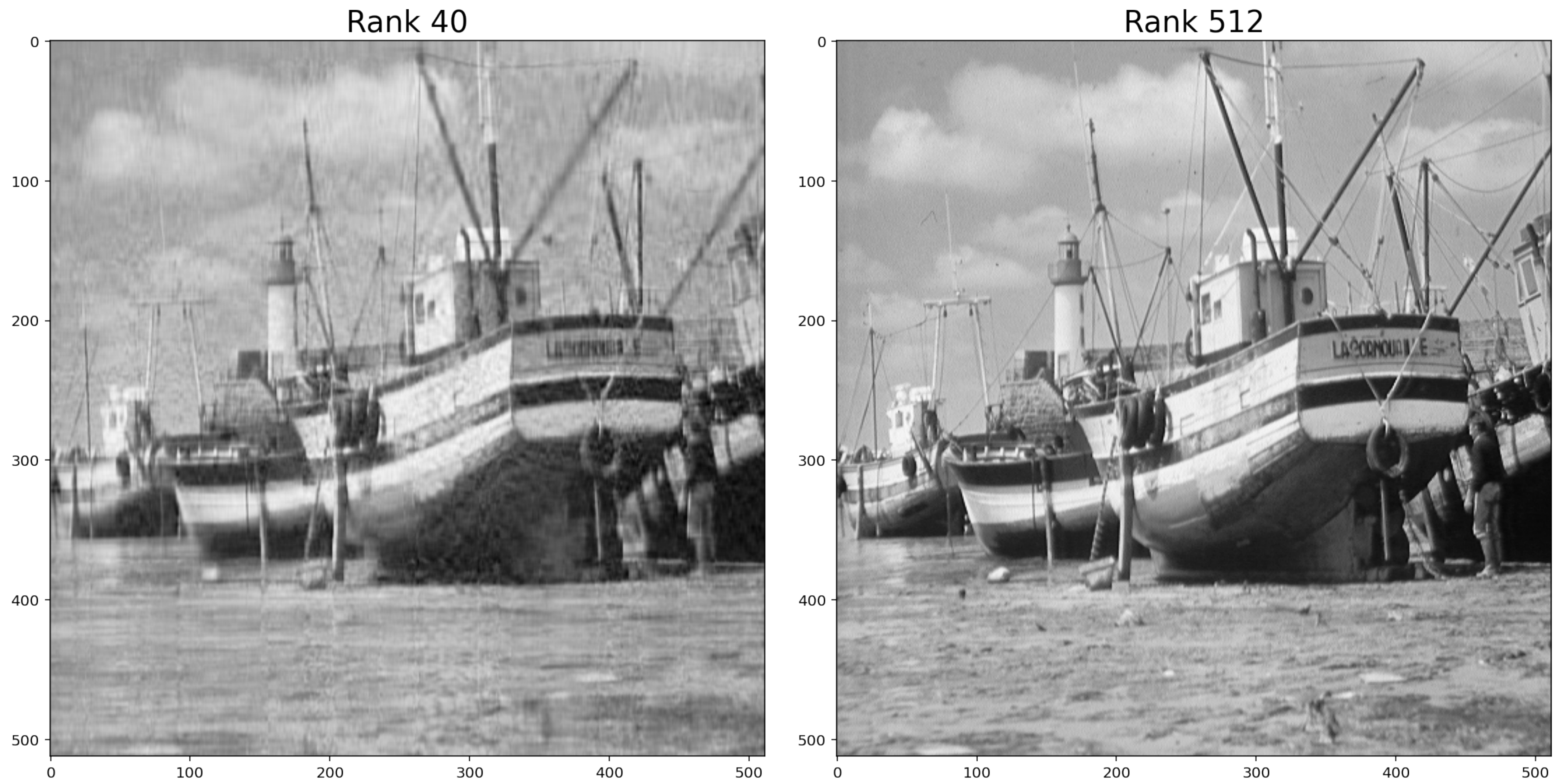



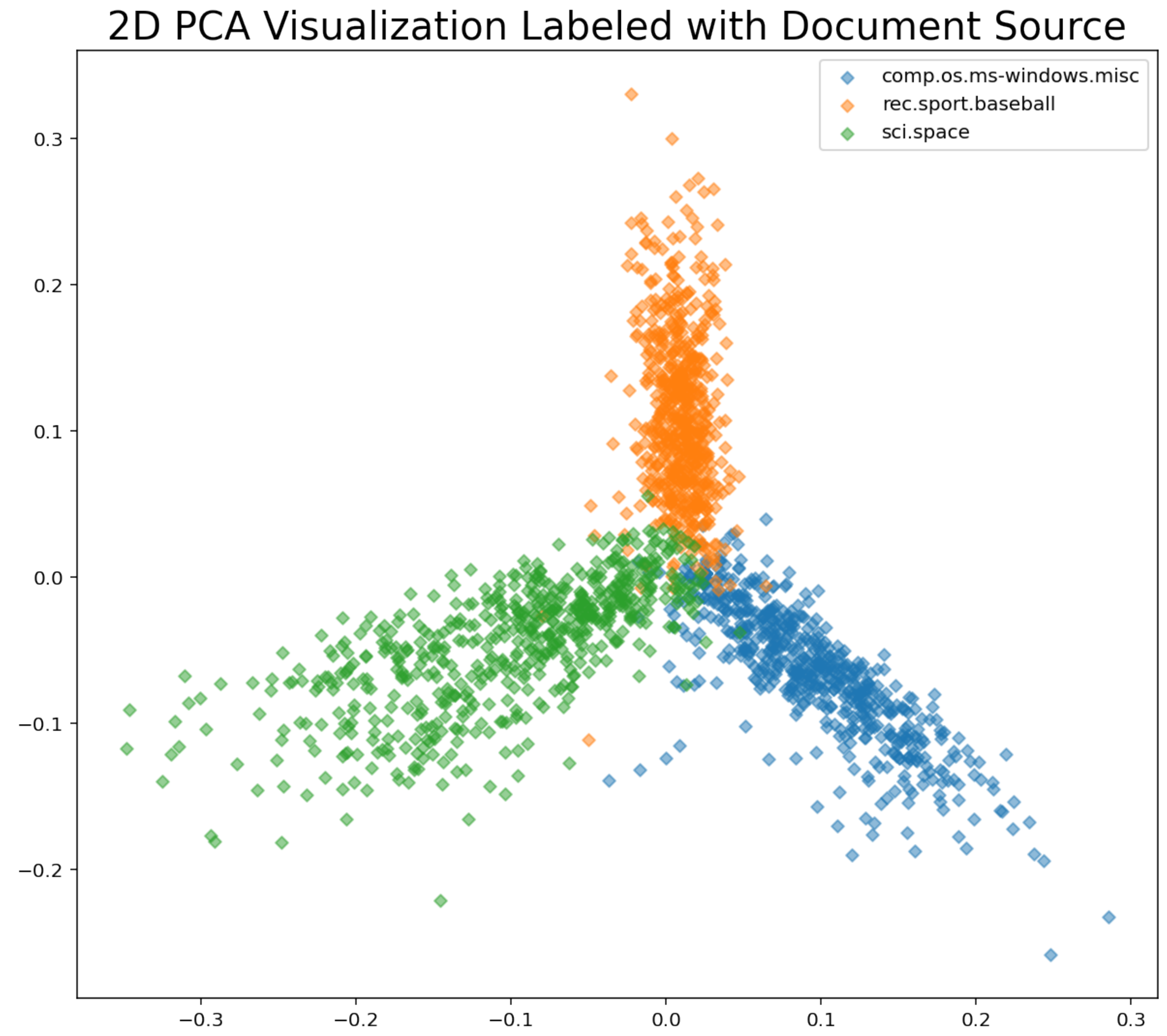

document classification

# Applications of SVD

- Reduced SVD, pseudoinverses and least squares

image compression



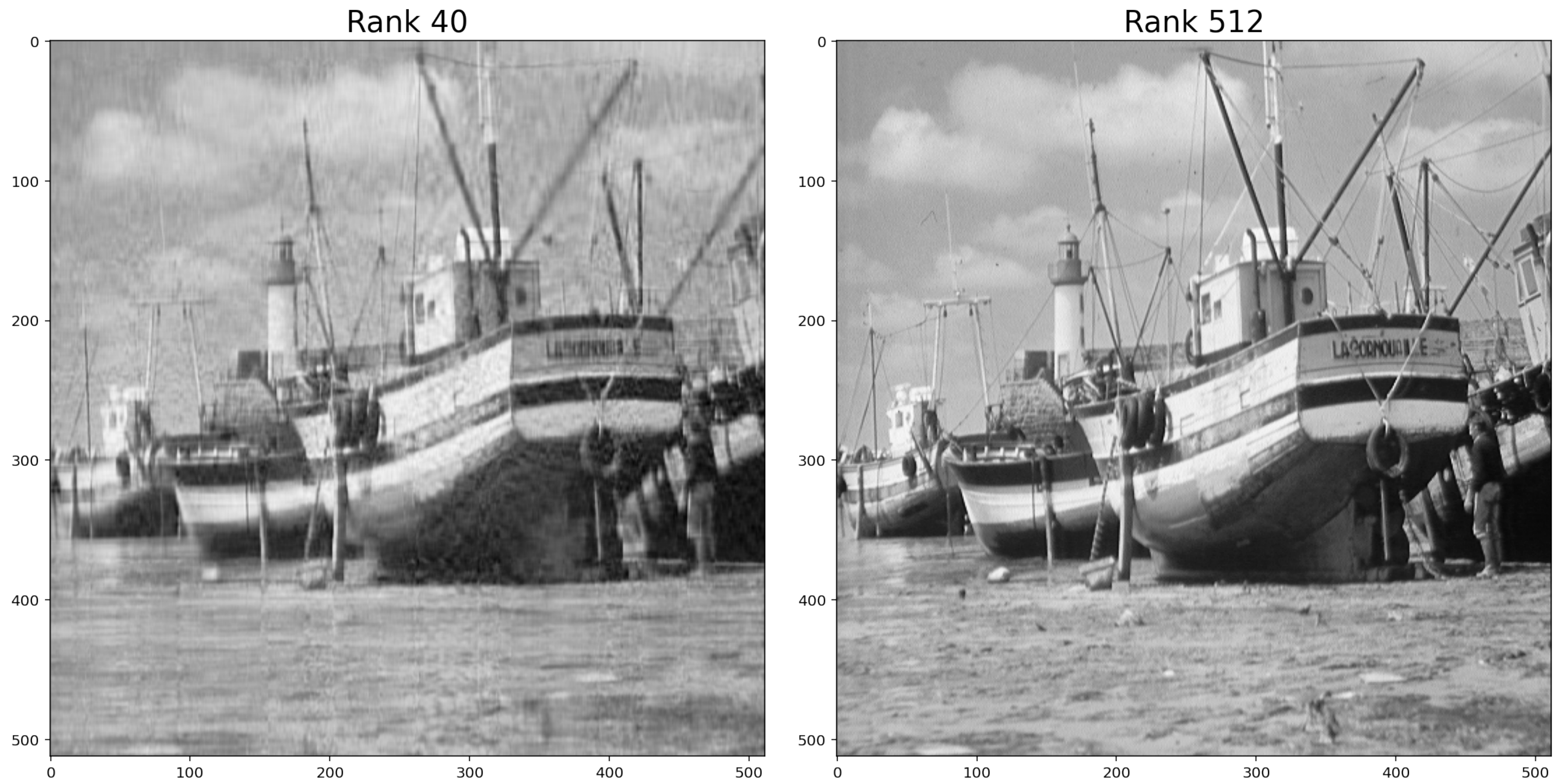



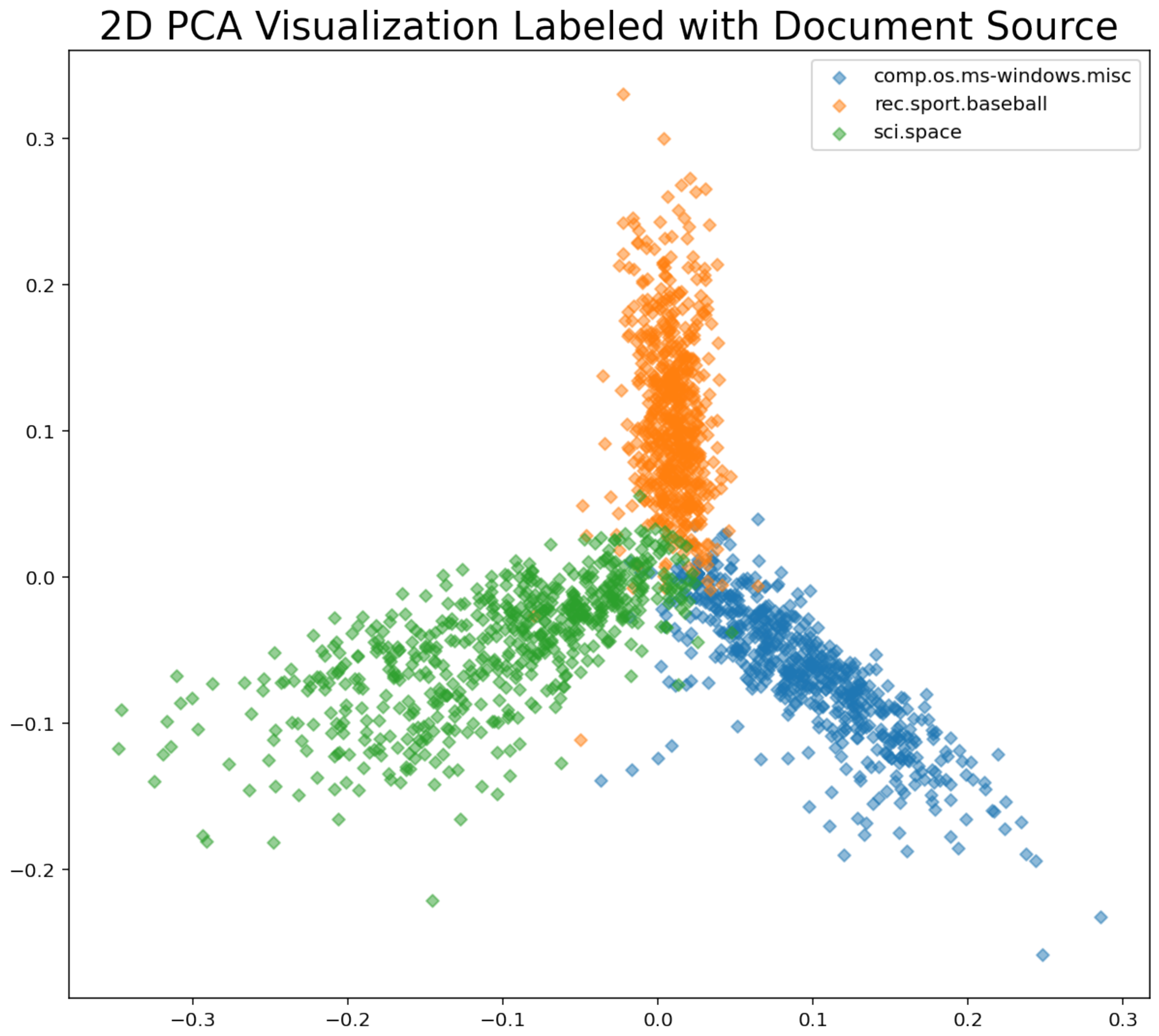

document classification

# Applications of SVD

- Reduced SVD, pseudoinverses and least squares
  - If $A^+ = V\Sigma^{-1}U^T$, then $A^+\mathbf{b}$ is a least squares solution of minimum length

image compression



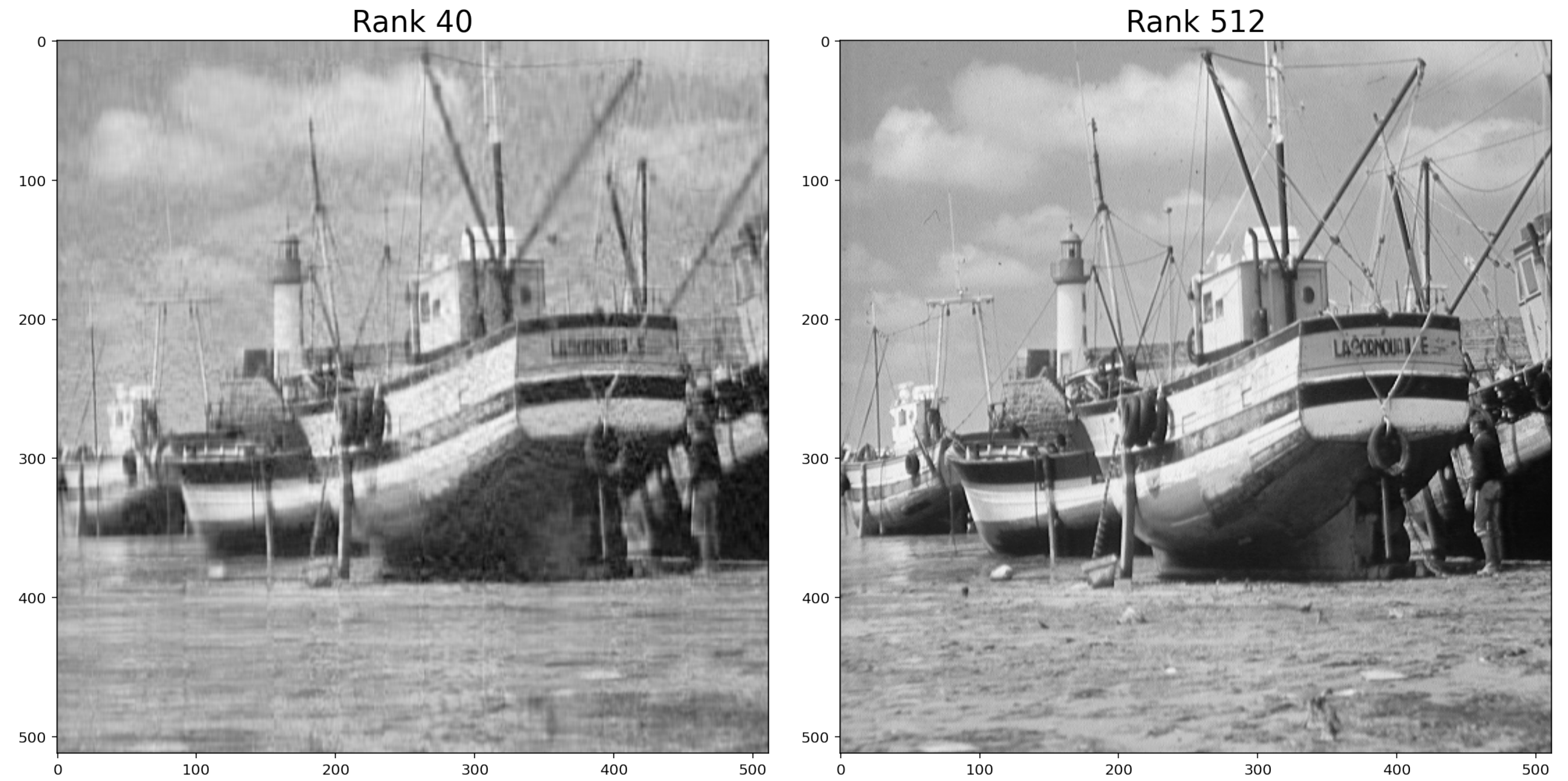



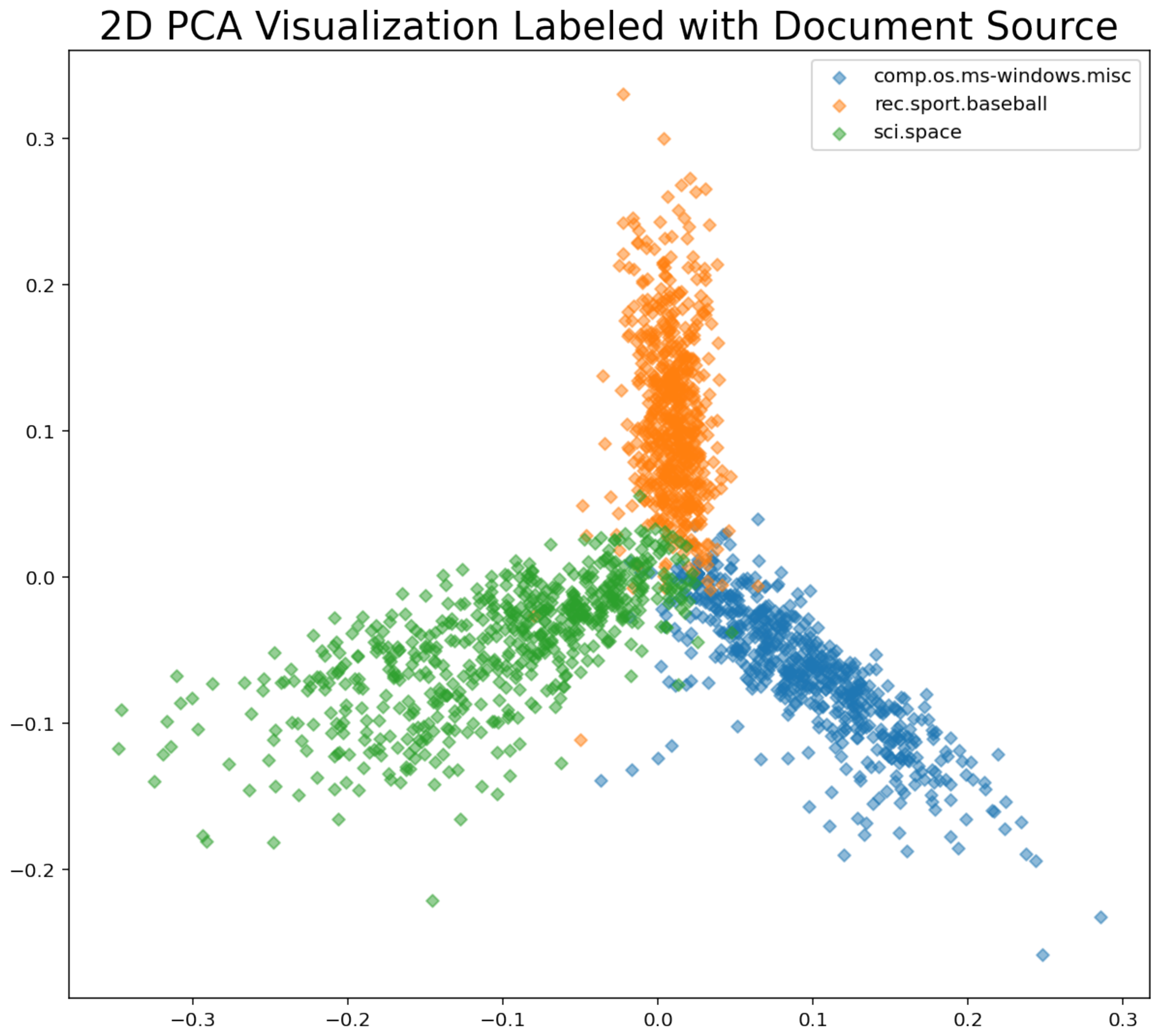

document classification

# Applications of SVD

- Reduced SVD, pseudoinverses and least squares
  - If $A^+ = V\Sigma^{-1}U^T$, then $A^+\mathbf{b}$ is a least squares solution of minimum length
- Low Rank Approximation and Data Compression

image compression



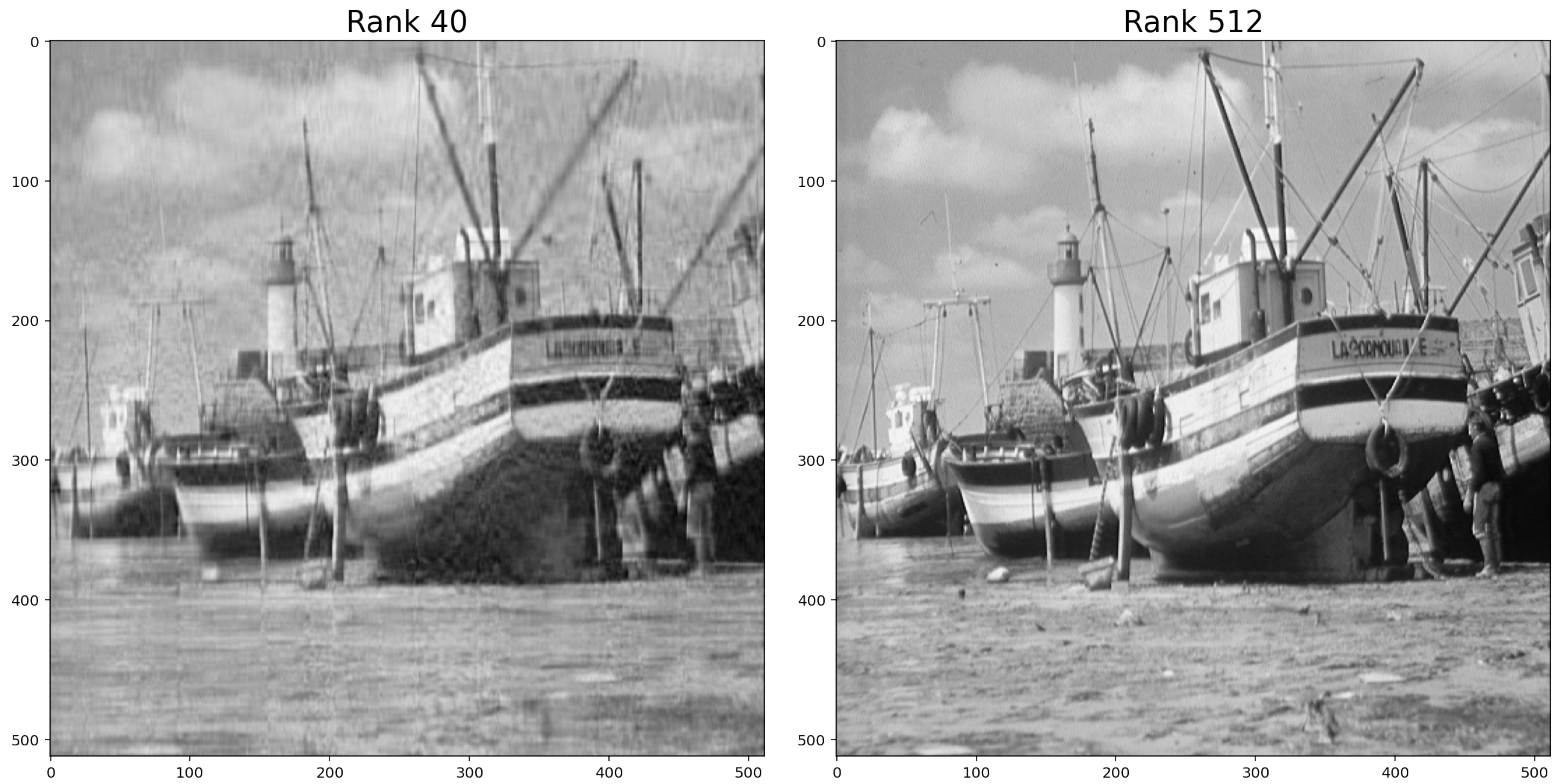



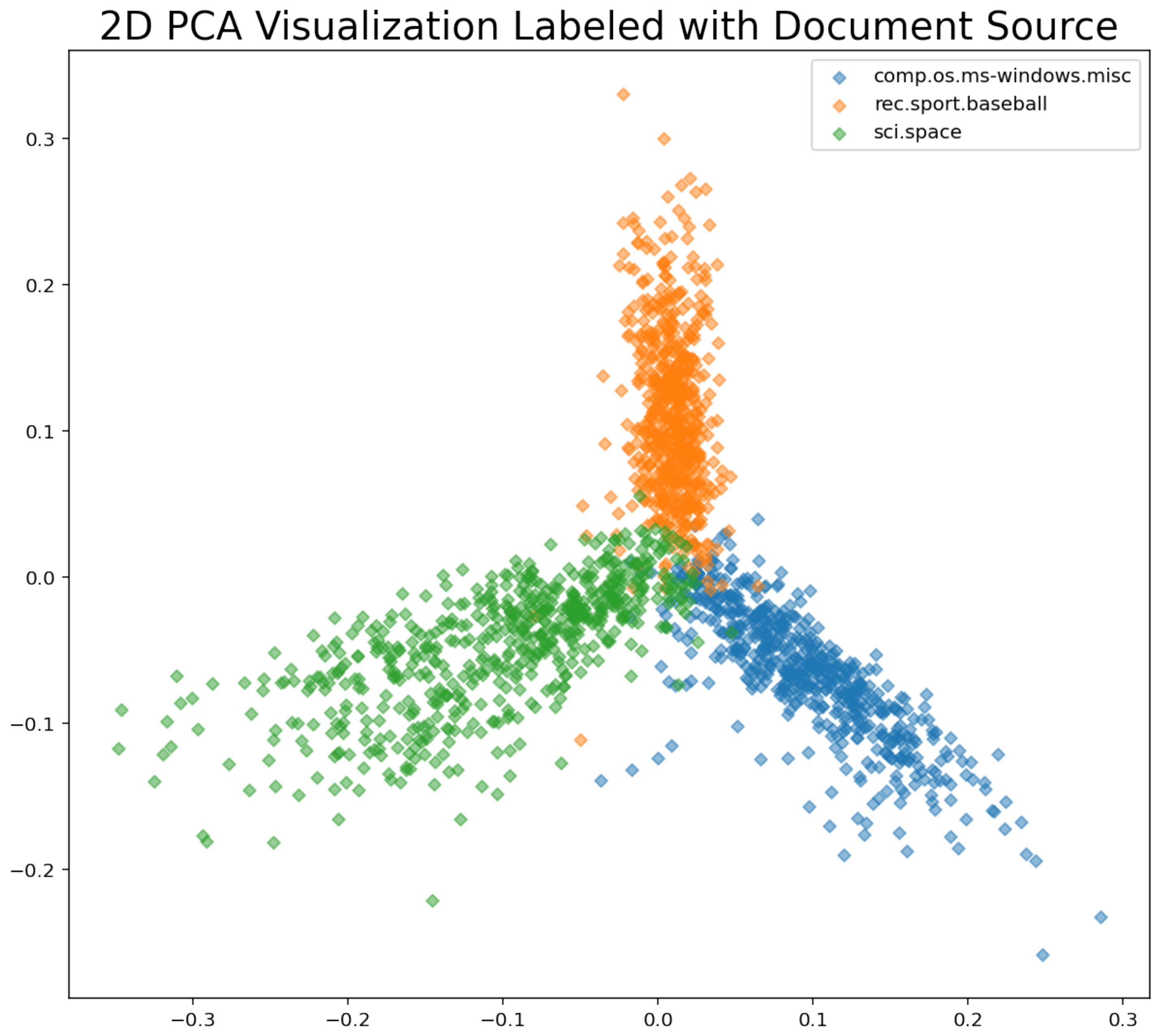

document classification

# Applications of SVD

- Reduced SVD, pseudoinverses and least squares
  - If $A^+ = V\Sigma^{-1}U^T$, then $A^+\mathbf{b}$ is a least squares solution of minimum length
- Low Rank Approximation and Data Compression
  - Replacing small singular values with zero in $\Sigma$ gives a good approximation to $A$.

image compression



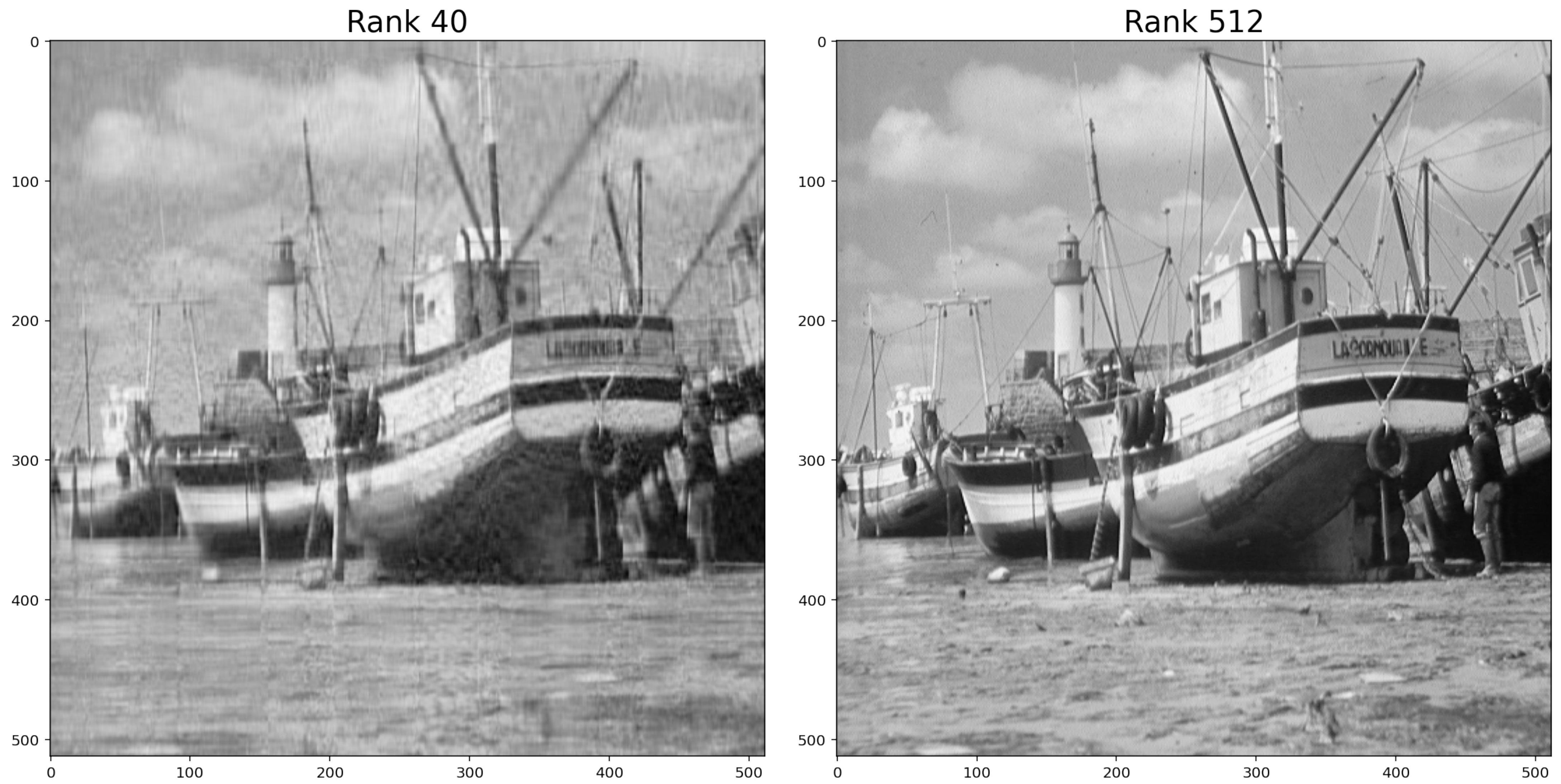



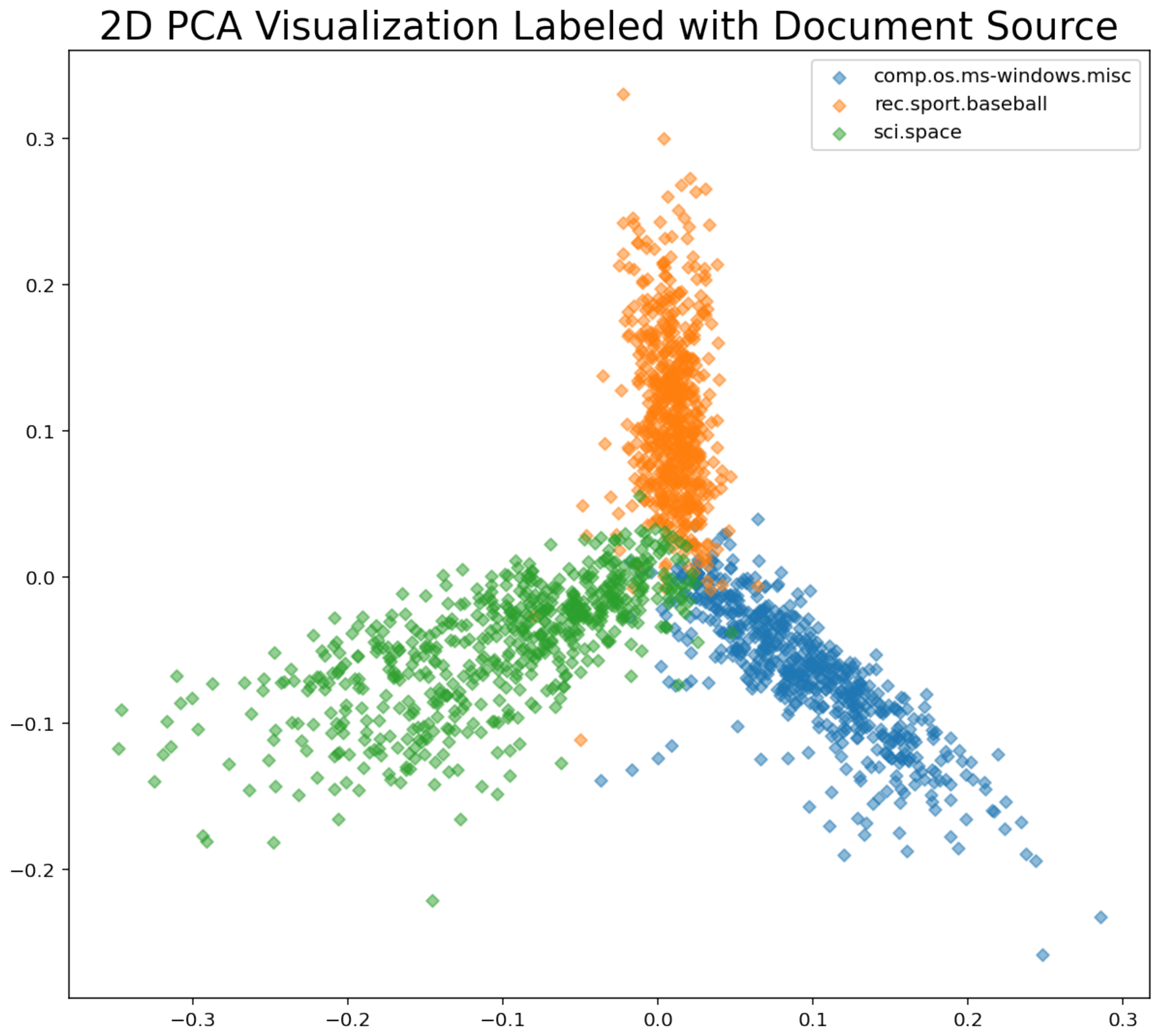

document classification

# Applications of SVD

- Reduced SVD, pseudoinverses and least squares
  - If $A^+ = V\Sigma^{-1}U^T$, then $A^+\mathbf{b}$ is a least squares solution of minimum length
- Low Rank Approximation and Data Compression
  - Replacing small singular values with zero in $\Sigma$ gives a good approximation to $A$.
  - This is used for image compression

image compression



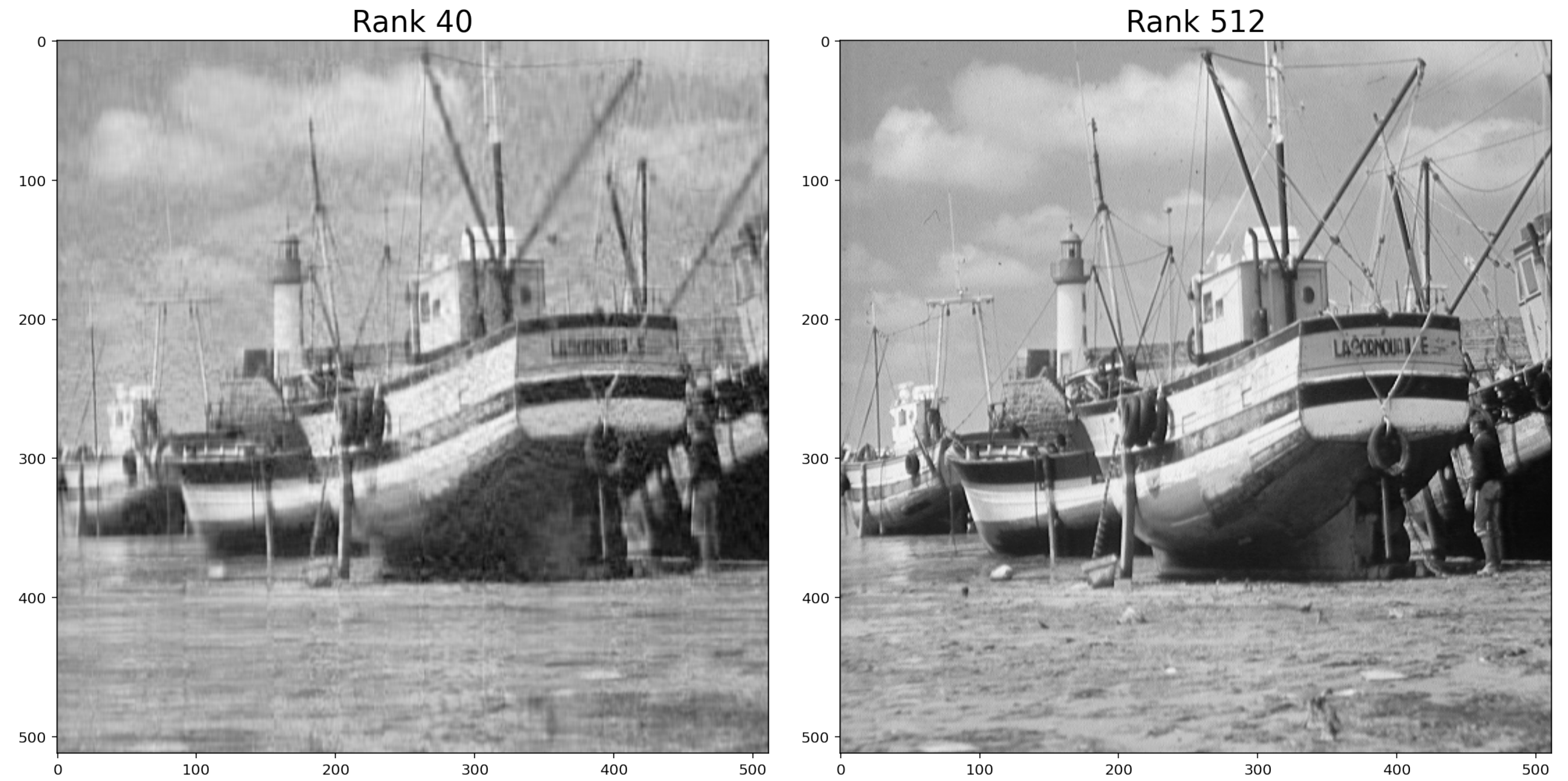



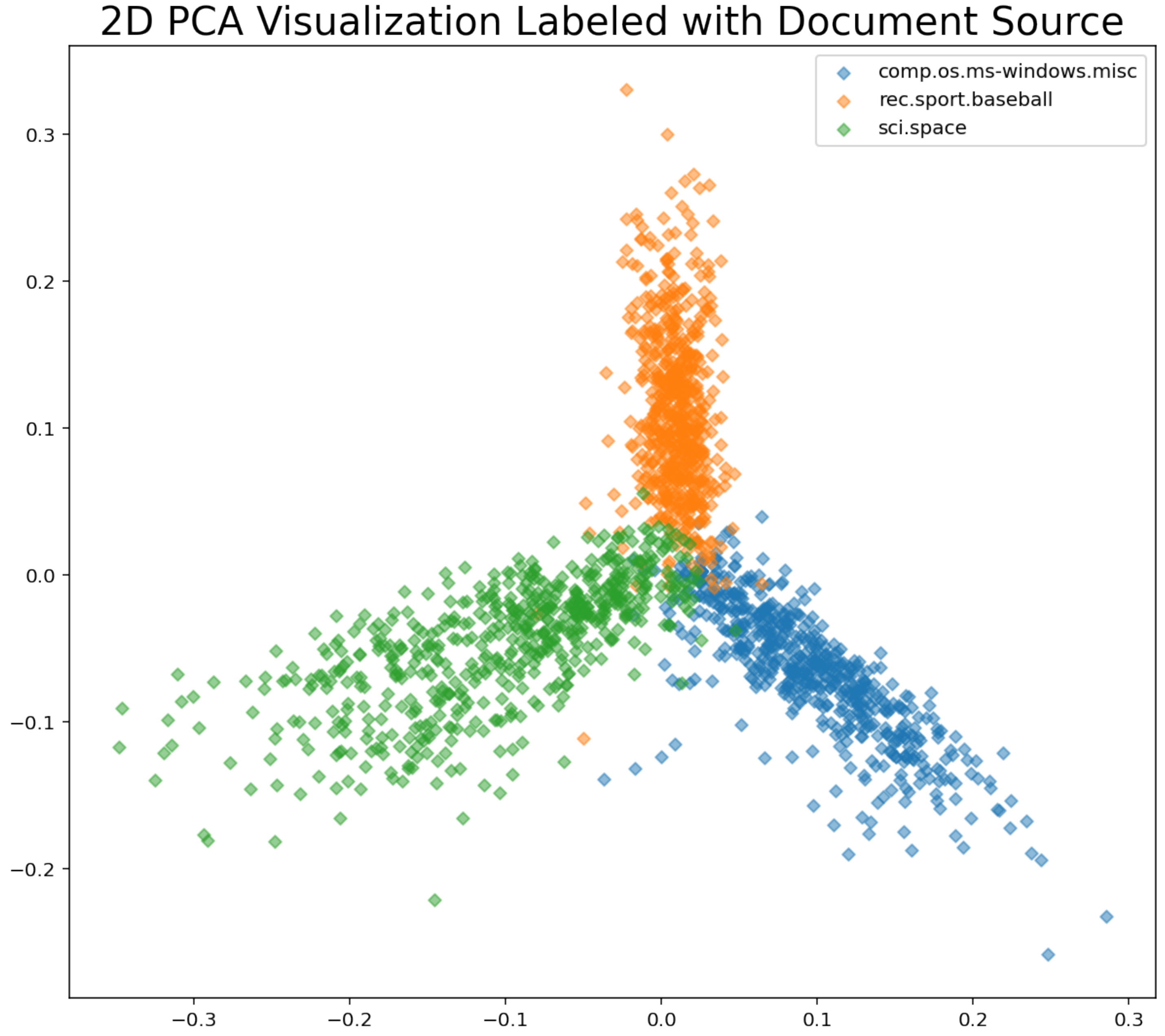

document classification

# Applications of SVD

- Reduced SVD, pseudoinverses and least squares
  - If $A^+ = V\Sigma^{-1}U^T$, then $A^+\mathbf{b}$ is a least squares solution of minimum length
- Low Rank Approximation and Data Compression
  - Replacing small singular values with zero in $\Sigma$ gives a good approximation to $A$.
  - This is used for image compression
- Principle Component Analysis

image compression



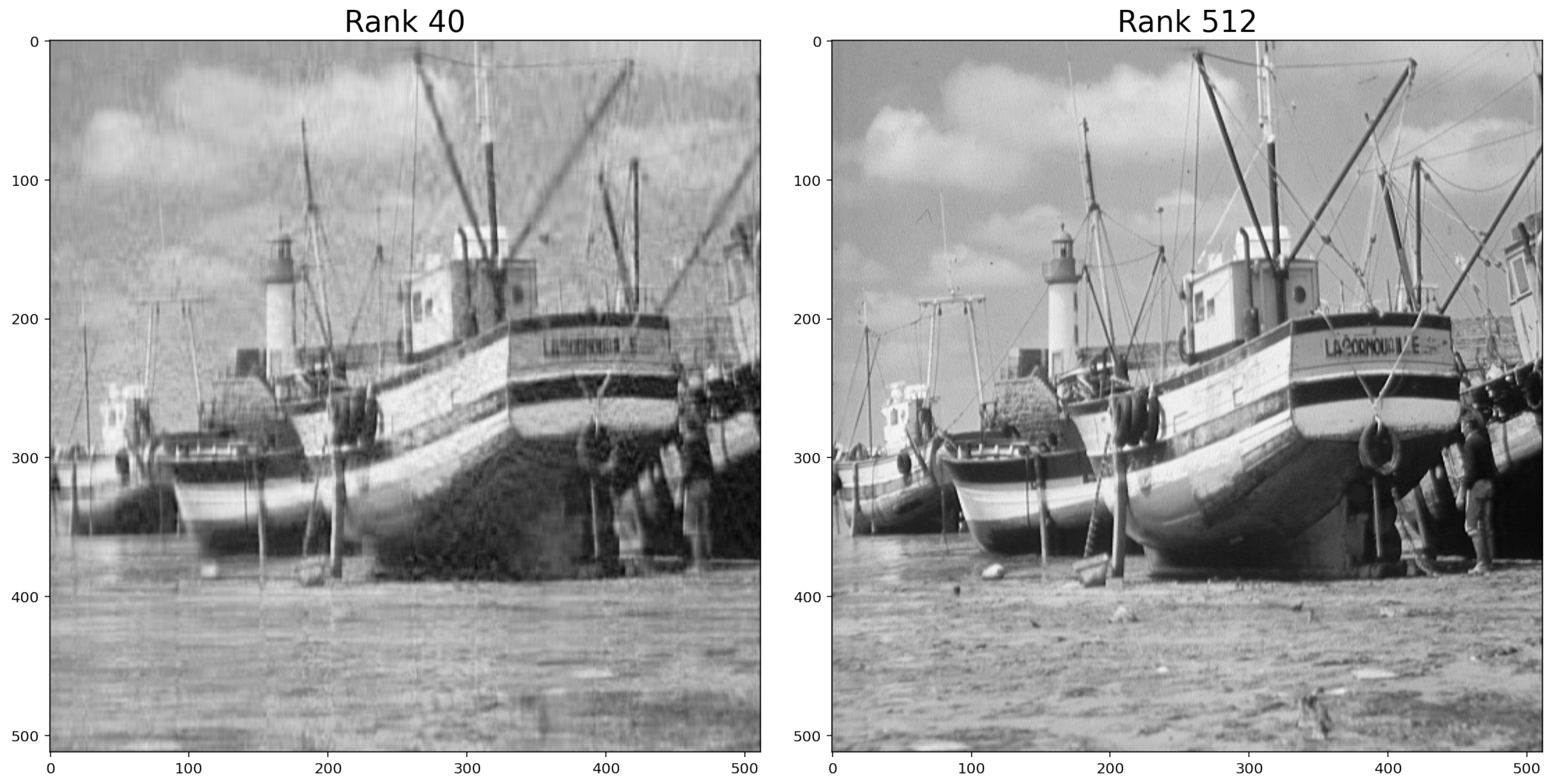



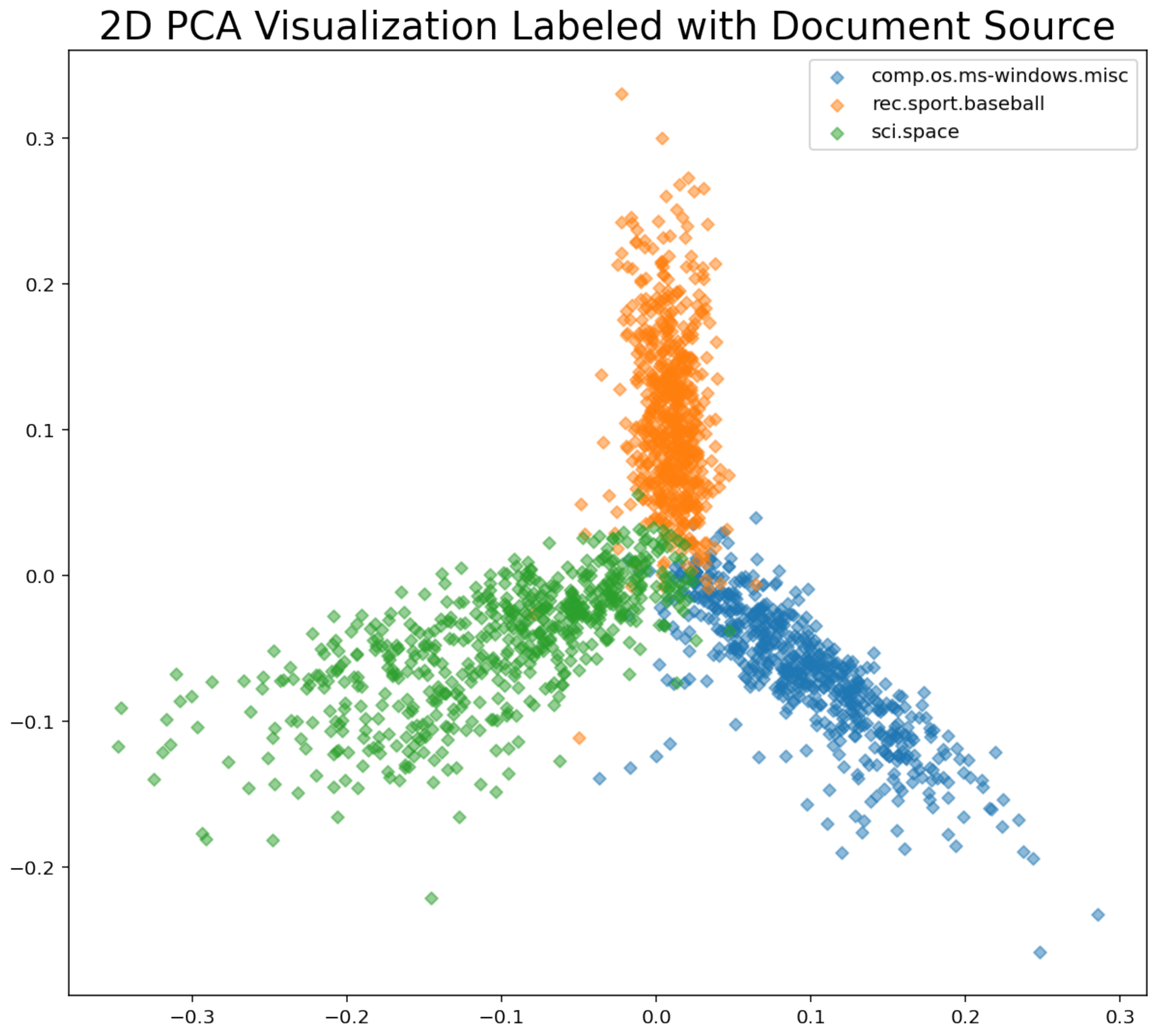

document classification

# Applications of SVD

- Reduced SVD, pseudoinverses and least squares
  - If $A^+ = V\Sigma^{-1}U^T$, then $A^+\mathbf{b}$ is a least squares solution of minimum length
- Low Rank Approximation and Data Compression
  - Replacing small singular values with zero in $\Sigma$ gives a good approximation to $A$.
  - This is used for image compression
- Principle Component Analysis
  - Large singular vectors are "most affected."

image compression



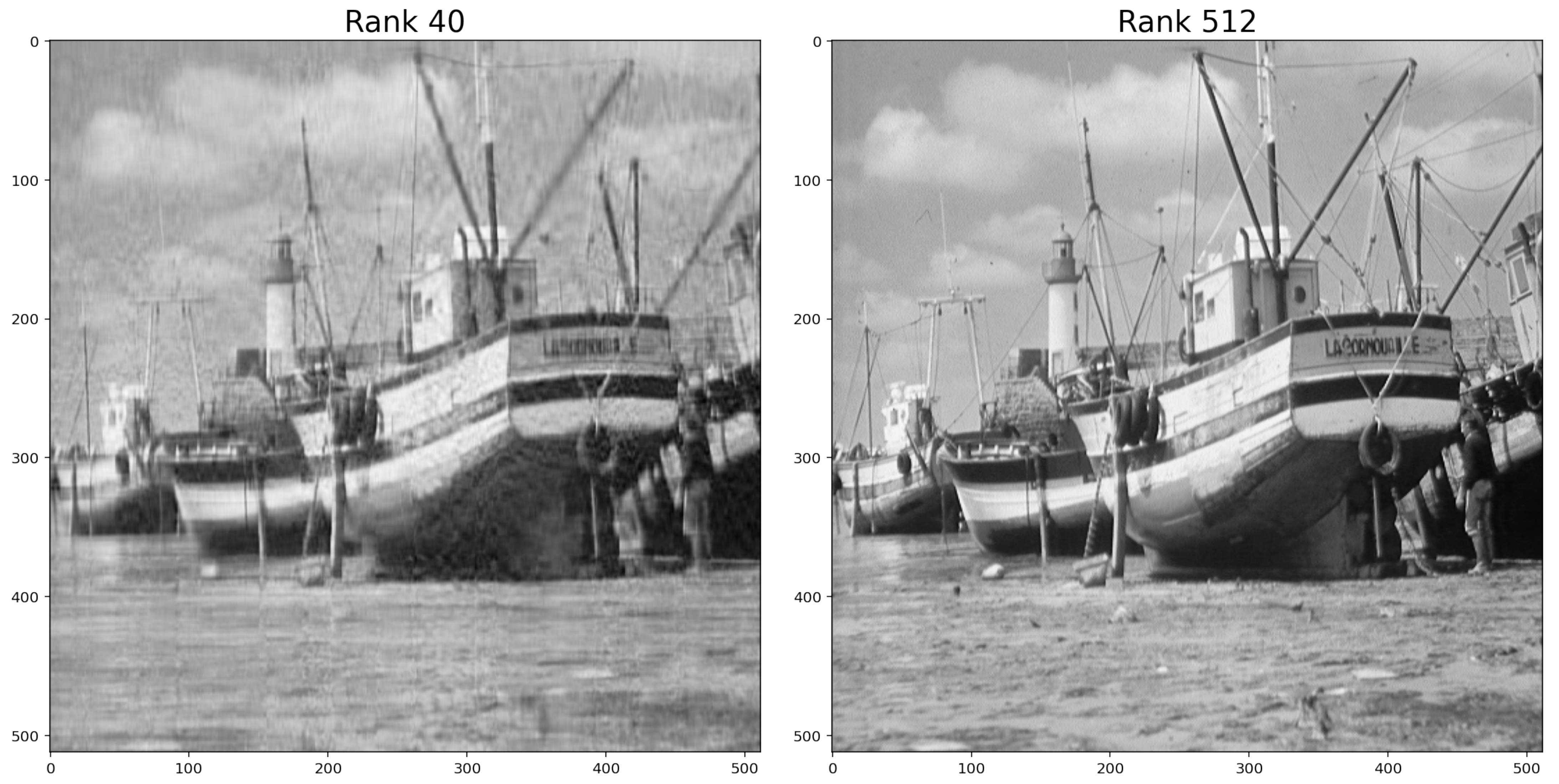



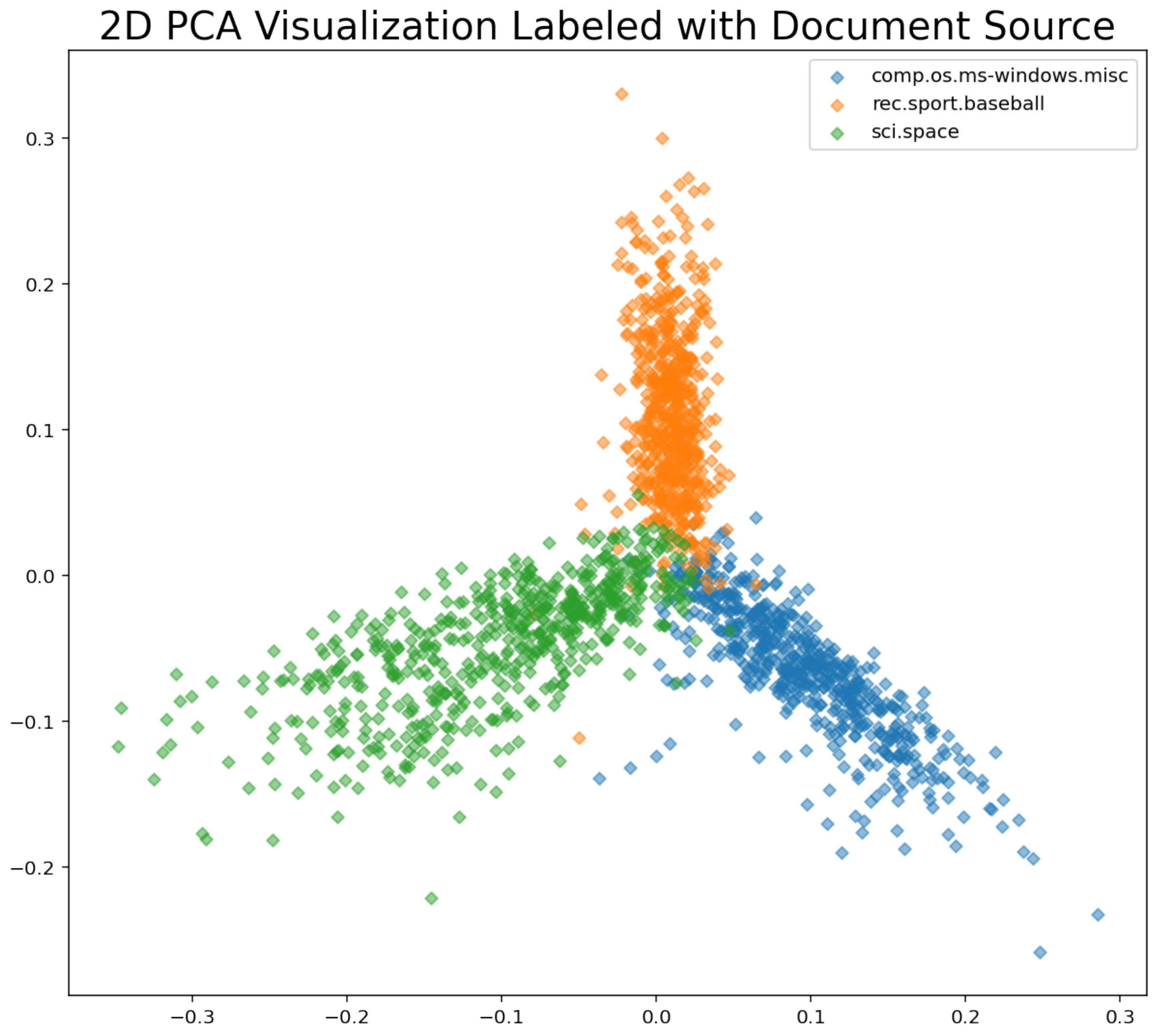

document classification

# Applications of SVD

- Reduced SVD, pseudoinverses and least squares
  - If $A^+ = V\Sigma^{-1}U^T$, then $A^+\mathbf{b}$ is a least squares solution of minimum length
- Low Rank Approximation and Data Compression
  - Replacing small singular values with zero in $\Sigma$ gives a good approximation to $A$.
  - This is used for image compression
- Principle Component Analysis
  - Large singular vectors are "most affected."
  - These are good vectors to look at for classifying data

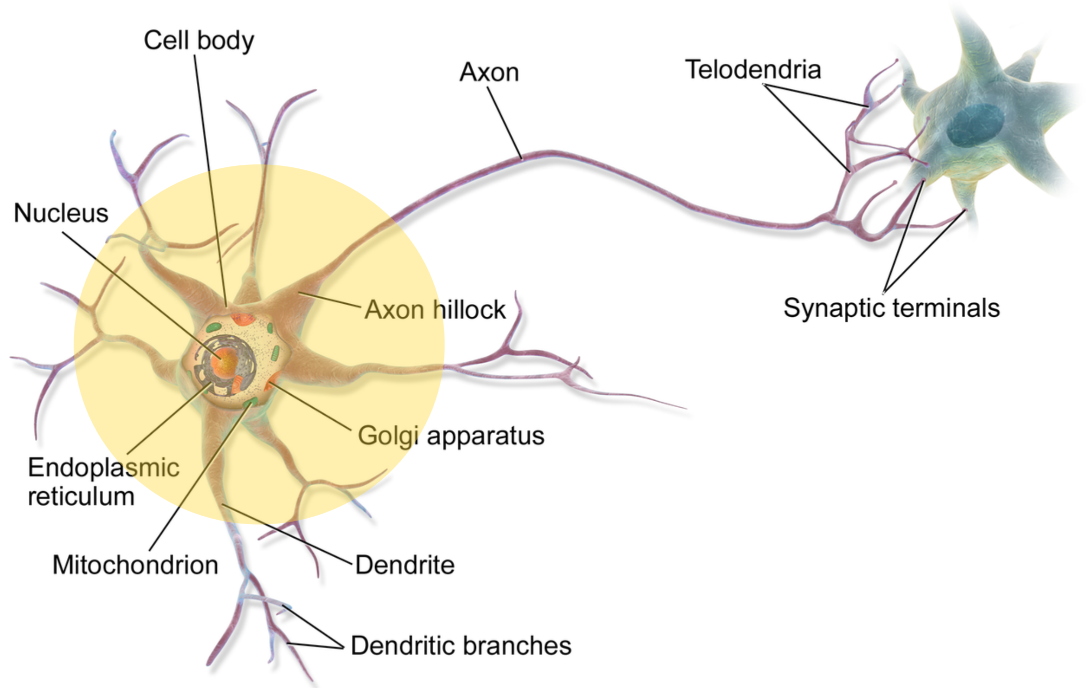
image compression

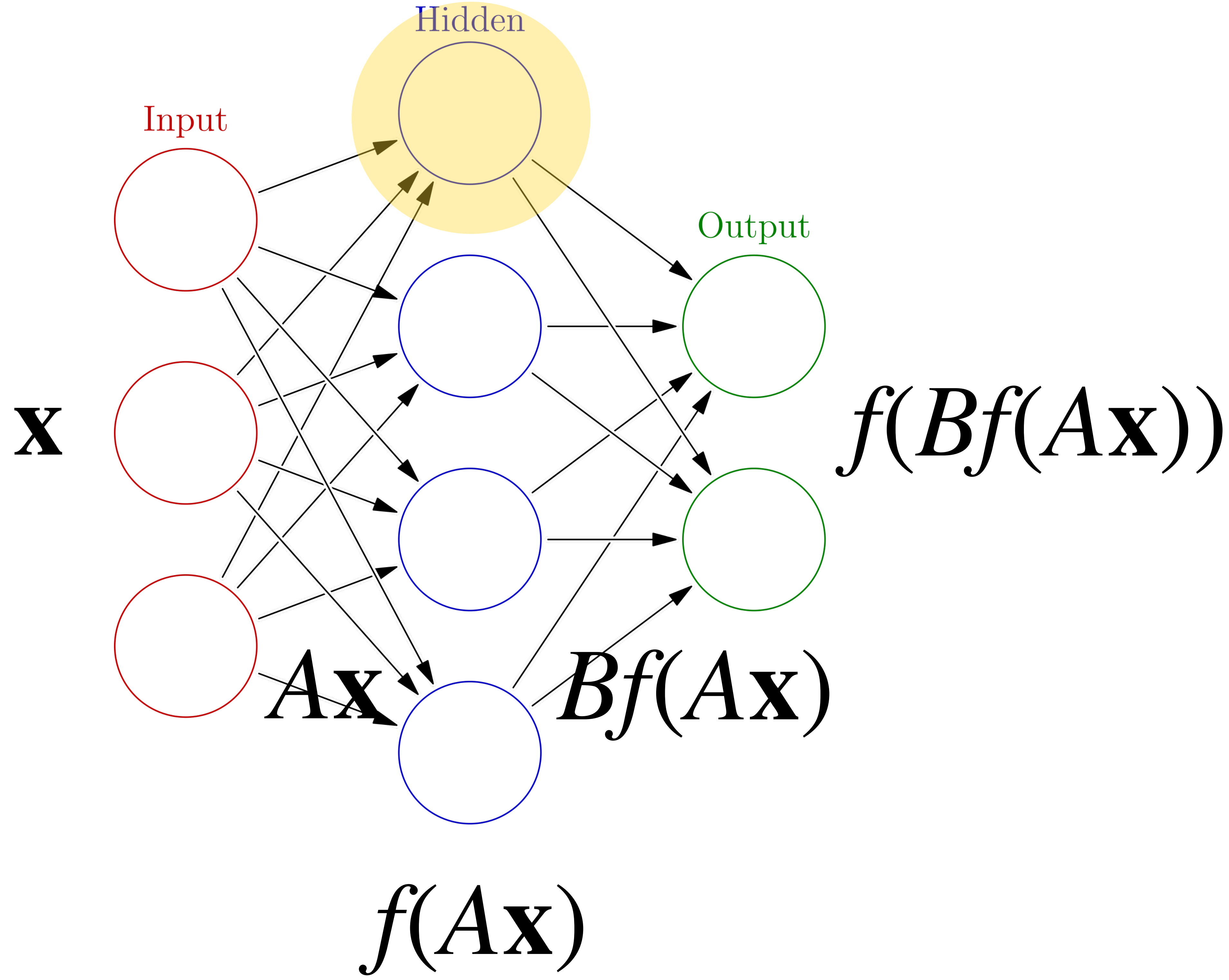



document classification

# Neural Networks (Non-Linearity)

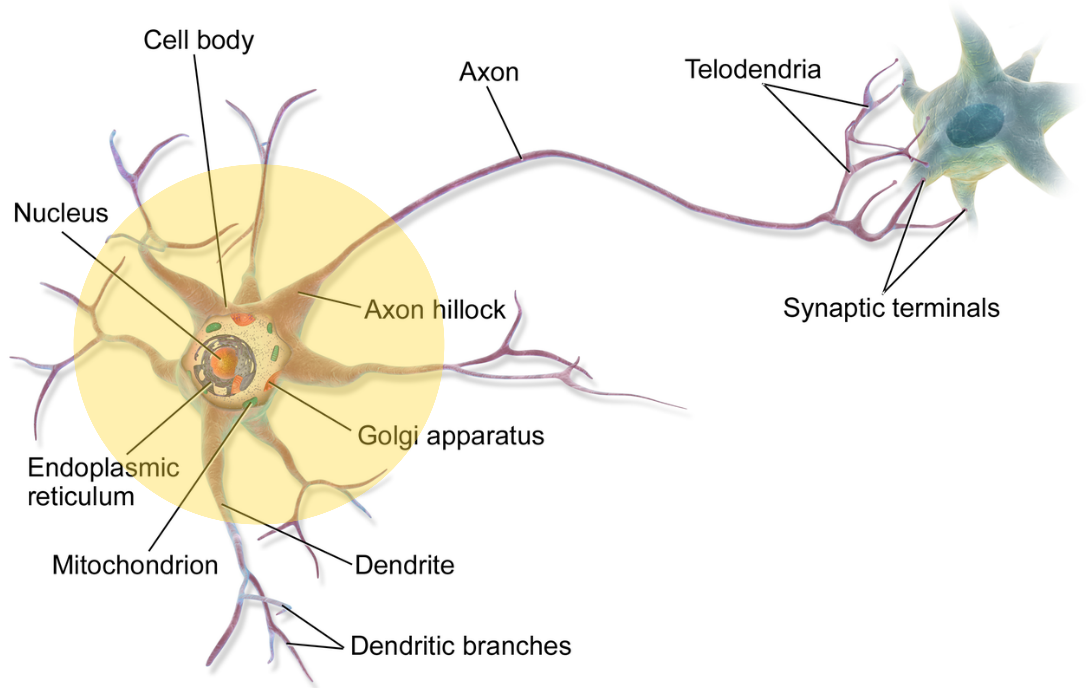


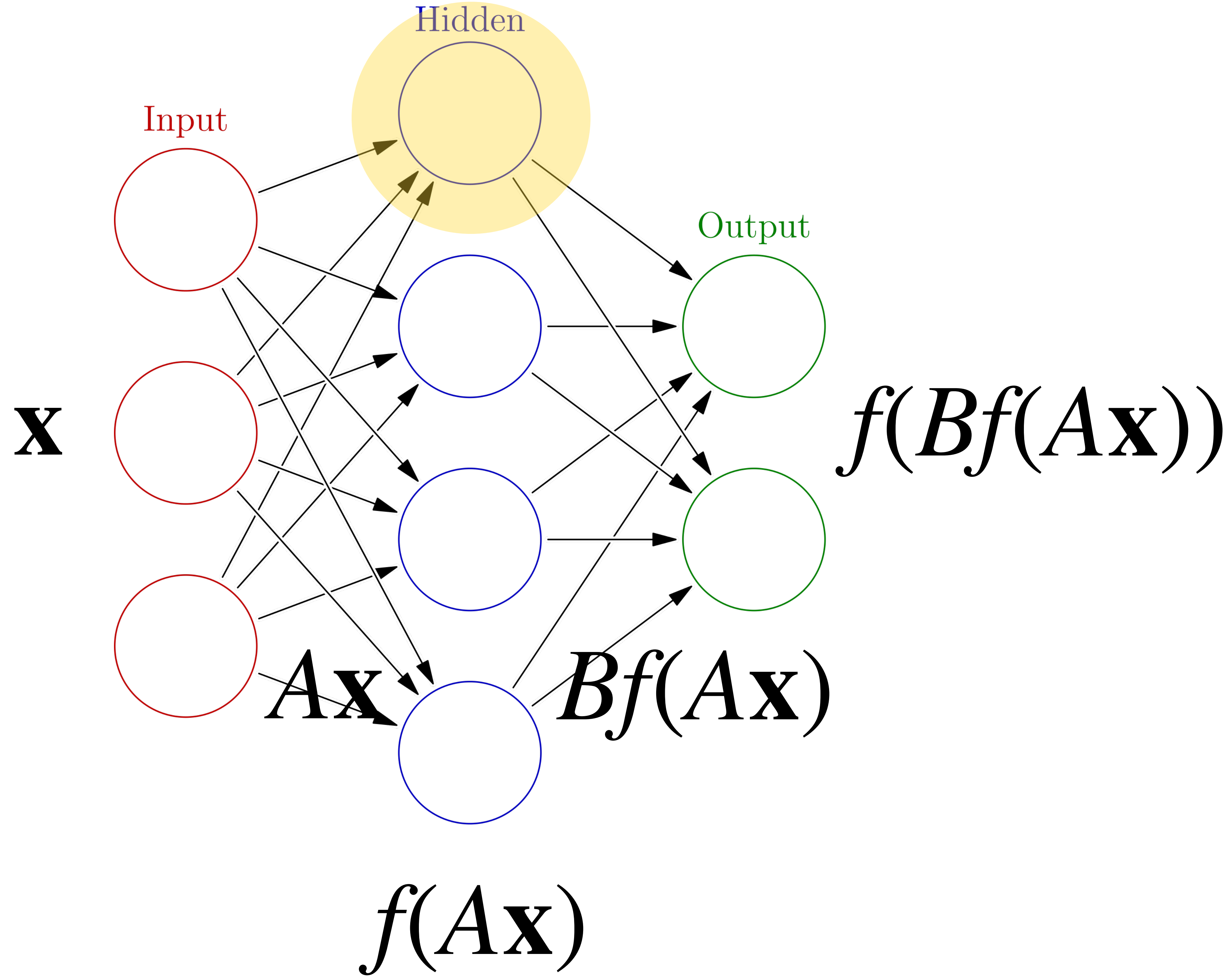




https://commons.wikimedia.org/wiki/File:Blausen_0657_MultipolarNeuron.png
https://commons.wikimedia.org/wiki/File:Colored_neural_network.svg

# Neural Networks (Non-Linearity)

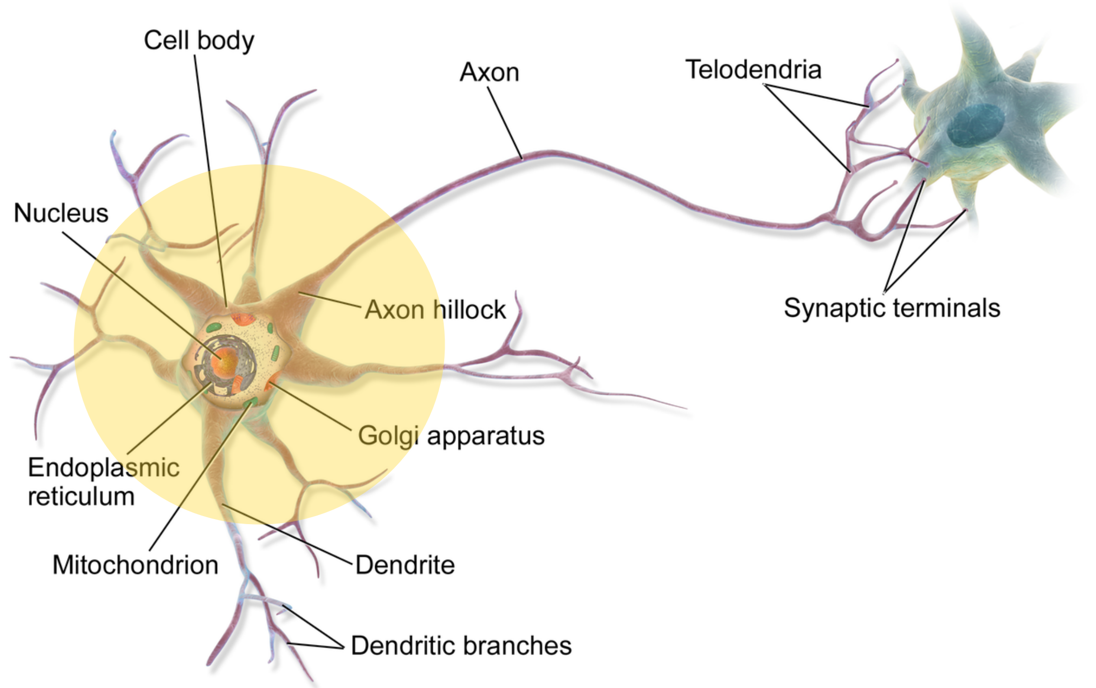
Neural networks are models of artificial neurons bundles.



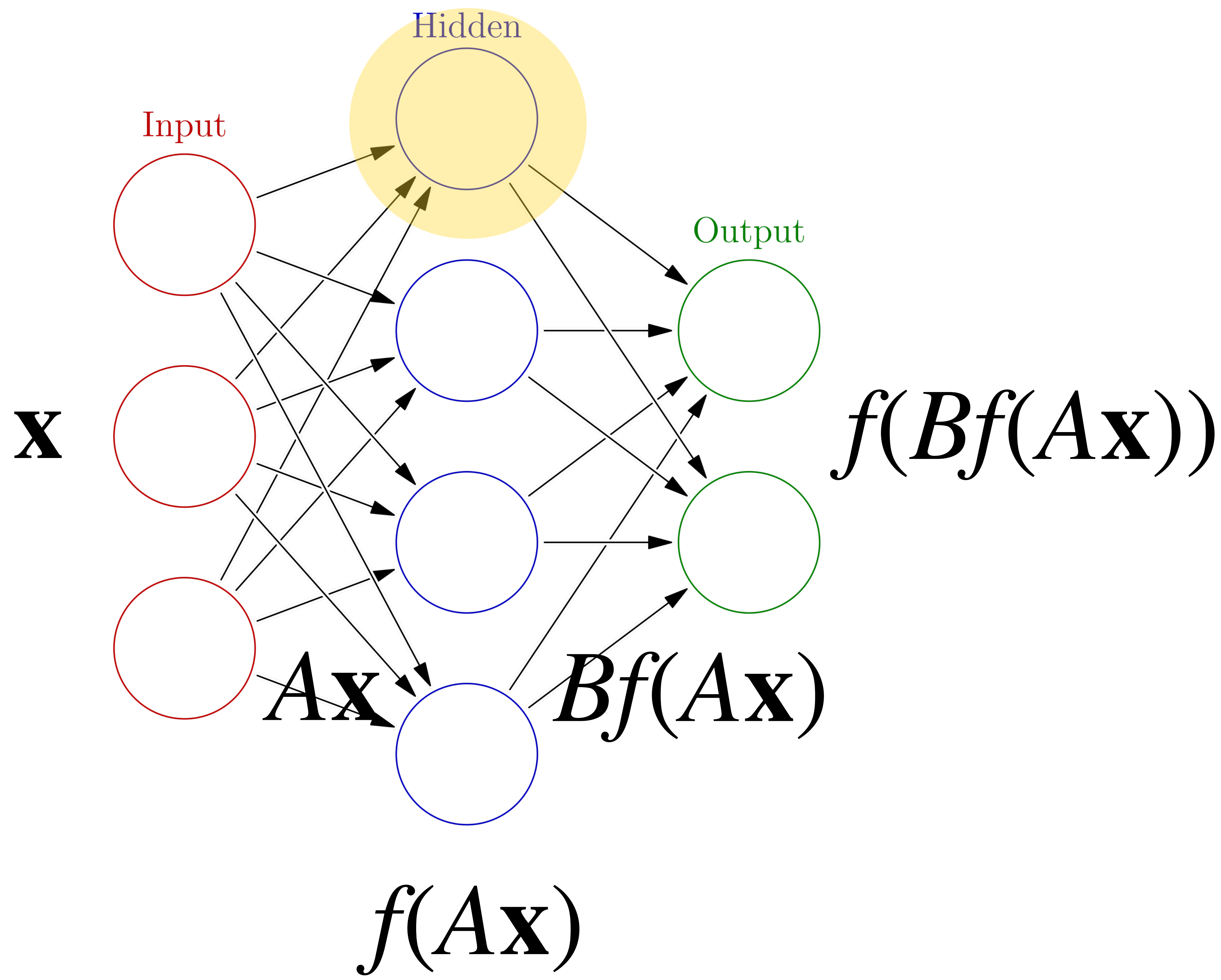

https://commons.wikimedia.org/wiki/File:Blausen_0657_MultipolarNeuron.png
https://commons.wikimedia.org/wiki/File:Colored_neural_network.svg

# Neural Networks (Non-Linearity)



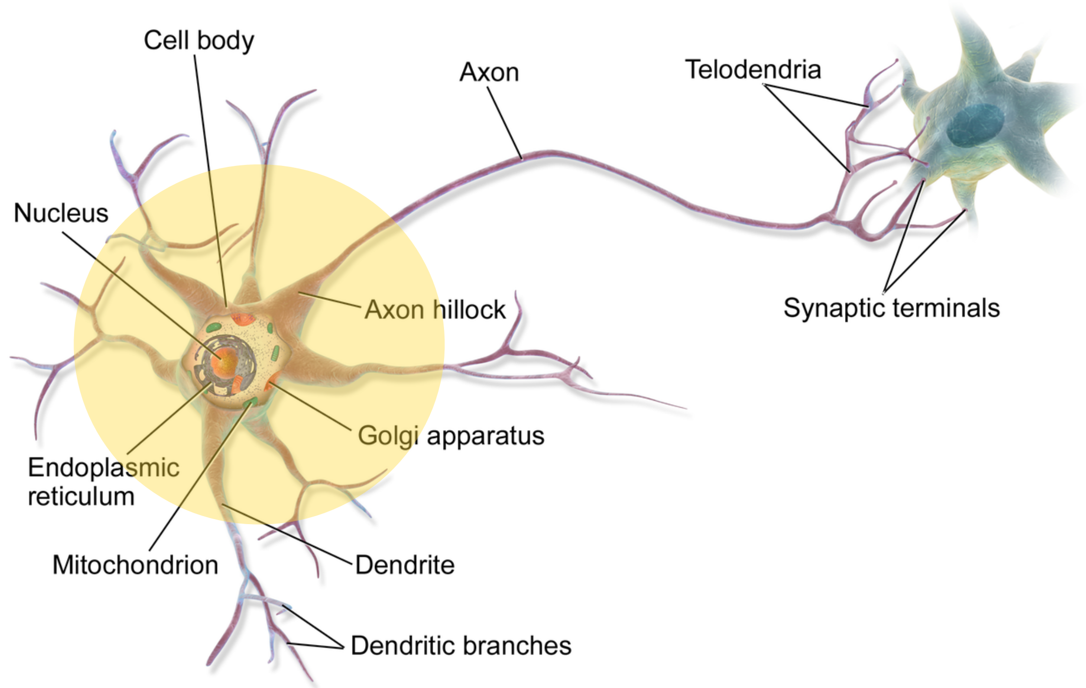

Neural networks are models of artificial neurons bundles.

Given an input vector $\mathbf{x}$, it is transformed into a *hidden* vector $A\mathbf{x}$ by a linear transformation, and then an *activation function* $f$ is applied to the result.




https://commons.wikimedia.org/wiki/File:Blausen_0657_MultipolarNeuron.png
https://commons.wikimedia.org/wiki/File:Colored_neural_network.svg

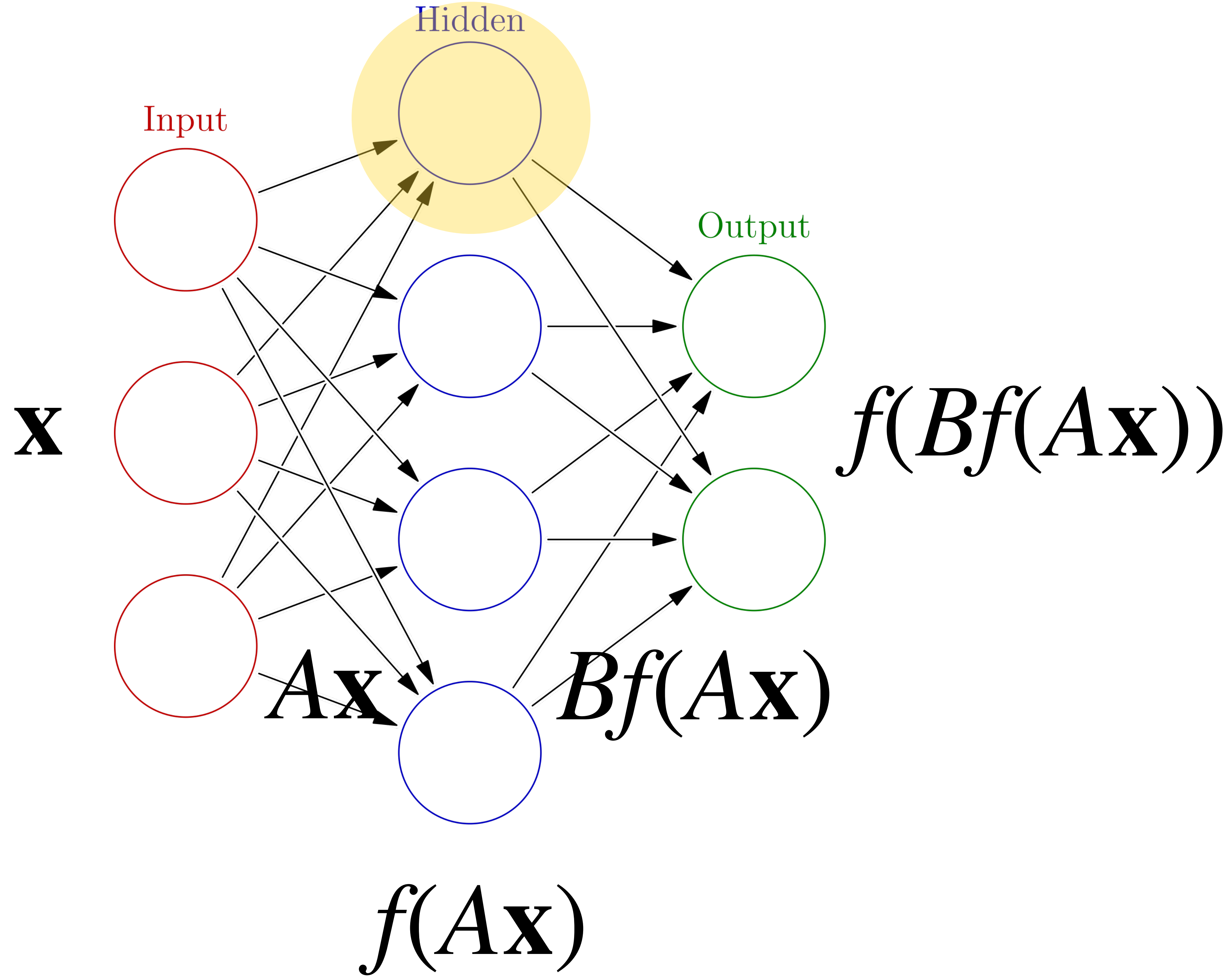
# Neural Networks (Non-Linearity)




Neural networks are models of artificial neurons bundles.

Given an input vector $\mathbf{x}$, it is transformed into a *hidden* vector $A\mathbf{x}$ by a linear transformation, and then an *activation function* $f$ is applied to the result.

**Neural networks are just matrix multiplications with intermediate calls to a nonlinear function $f$.**




https://commons.wikimedia.org/wiki/File:Blausen_0657_MultipolarNeuron.png
https://commons.wikimedia.org/wiki/File:Colored_neural_network.svg

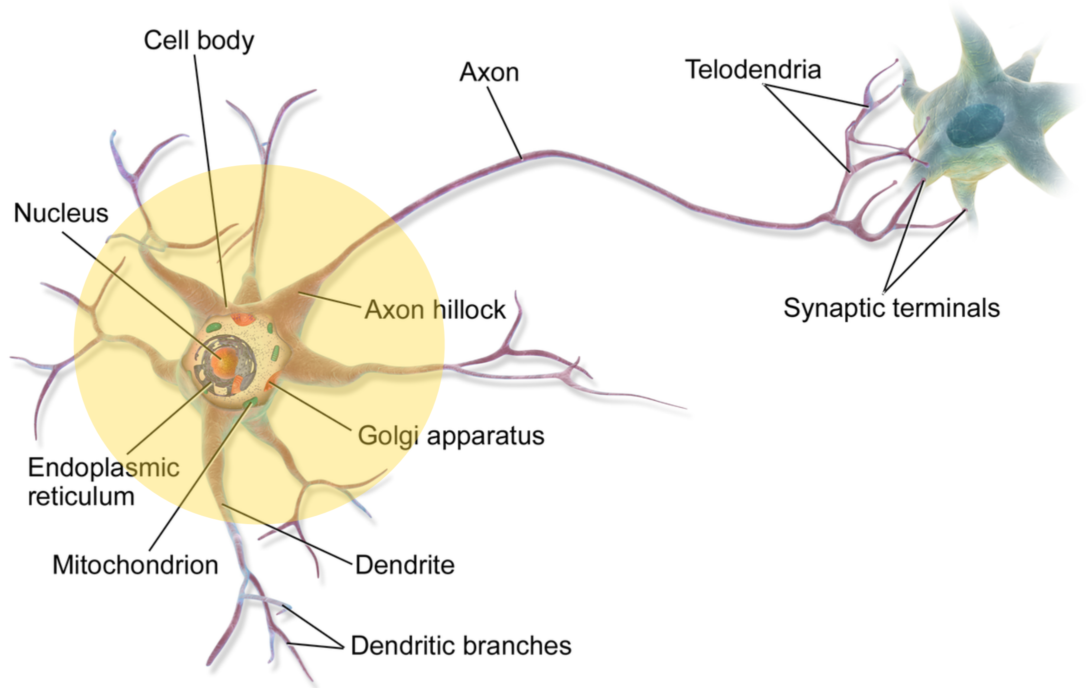
# Neural Networks (Non-Linearity)

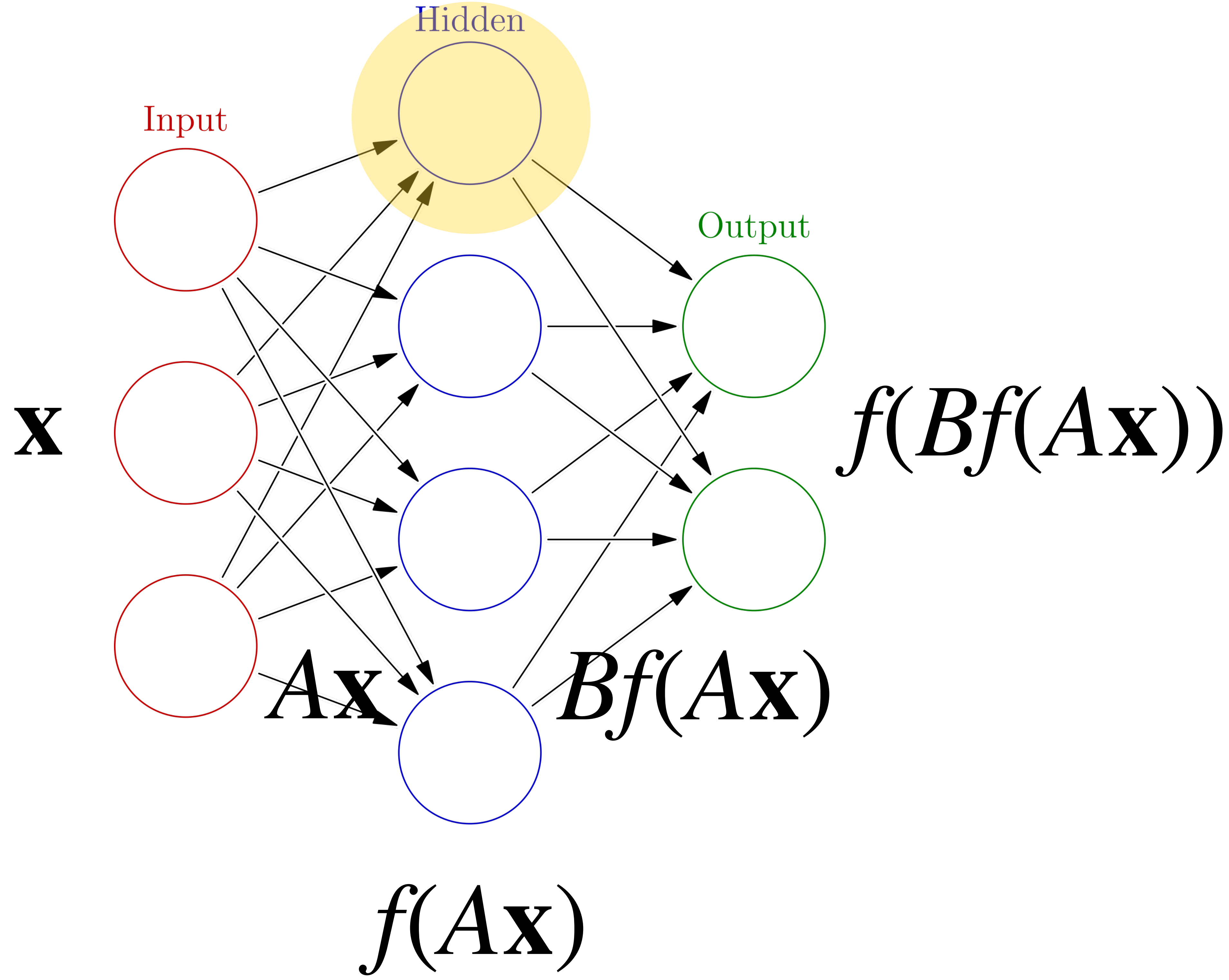
Neural networks are models of artificial neurons bundles.

Given an input vector $\mathbf{x}$, it is transformed into a *hidden* vector $A\mathbf{x}$ by a linear transformation, and then an *activation function* $f$ is applied to the result.

**Neural networks are just matrix multiplications with intermediate calls to a nonlinear function $f$.**

$$\mathrm{NN}(\mathbf{x}) = f(A_k(f(A_{k-1} \ldots f(A_1\mathbf{x}))$$



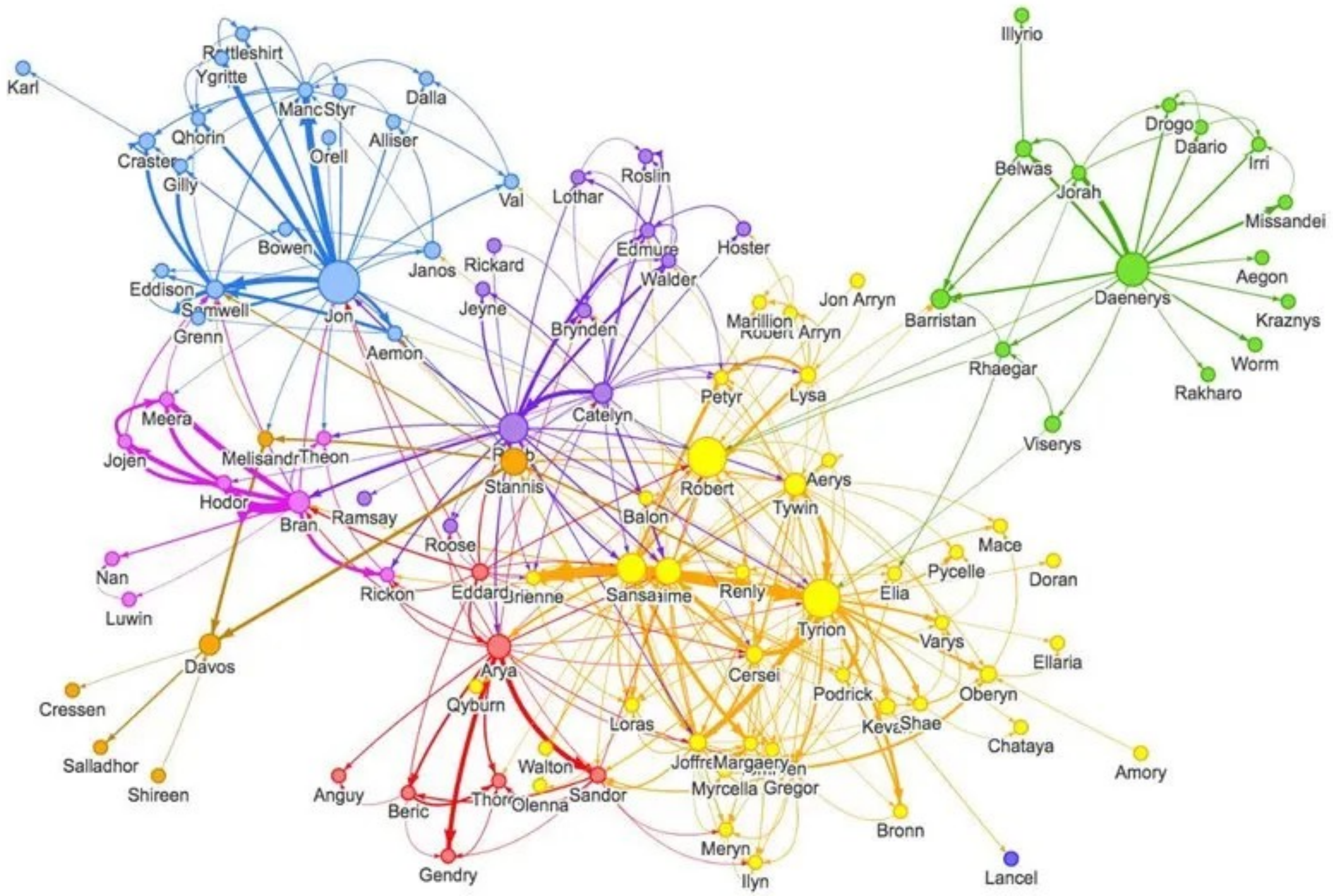




https://commons.wikimedia.org/wiki/File:Blausen_0657_MultipolarNeuron.png
https://commons.wikimedia.org/wiki/File:Colored_neural_network.svg

# Spectral/Algebraic Graph Theory

Graphs can be viewed as matrices.

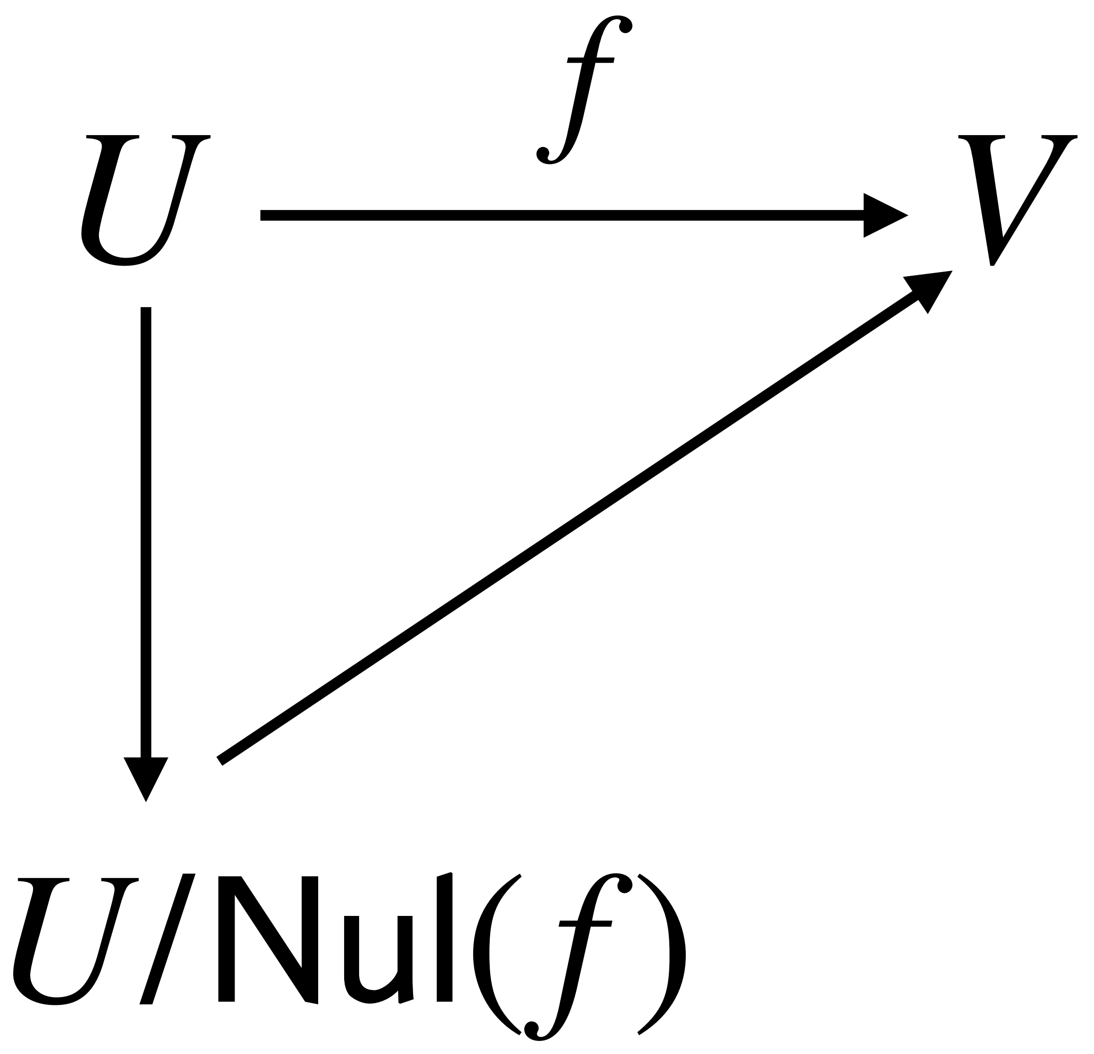
The finding eigenvalues in graphs can gives use better clustering and cutting algorithms.




https://medium.com/@n.rajadhyaksha/a-gentle-introduction-to-graph-spectral-filtering-df03ddc9d3f7

# Abstract Algebra

$$\frac{U}{\mathrm{Nul}(f)} \cong \mathrm{Range}(f)$$




There's a lot of beautiful structure in the algebra we've done in this course.

And there are lots of directions to go from here (infinite dimensional spaces, less restrictive settings like groups and modules,...)

# Course List

- CS 365 Foundations of Data Science
- CS 440 Intro to Artificial Intelligence
- CS 480 Intro to Computer Graphics
- CS 505 Intro to Natural Language Processing
- CS 506 Tools for Data Science
- CS 507 Intro to Optimization in ML
- CS 523 Deep Learning
- CS 530 Advanced Algorithms
- CS 531 Advanced Optimization Algorithms
- CS 542 Machine Learning
- CS 565 Algorithmic Data Mining
- CS 581 Computational Fabrication
- CS 583 Audio Computation

*Some of these may not exist anymore...*

# Appreciations

# The Course Staff

I'd like to thank:

**Rahul Mitra, Ryan Yu, Vishesh Jain, Jincheng Zhang, Reshab Chhabra, Rachel Du, Yi Du, Eugene Jung, Chris Min, Ieva Sagaitis, Aparna Singh, Kevin Wrenn**

If you see them around you should thank them as well.

# The CS Department Staff

```
If you're ever in the CS Department office, be
kind to the people who work there. They work
very hard to keep all our courses running.
```

# The Students of CS132

```
Thanks for sticking with it.

For giving feedback.

For adjusting and re-adjusting.
```

fin