Linear Transformations: What Do They Do?

David I. Inouye

Dynamic Content · Last Updated: September 11, 2026

A linear representation turns the problem into a matrix question

We began with a representation function:

$$f : \mathbb{R}^d \to \mathbb{R}^m, \qquad \boldsymbol{y} = f(\boldsymbol{x}).$$

For this first representation model, assume f is linear:

$$\boldsymbol{y} = A\boldsymbol{x}, \qquad A \in \mathbb{R}^{m \times d}.$$

> **What can this transformation do to a high-dimensional feature space?**

Matrix multiplication generalizes scalar multiplication

One dimension

$$y = ax$$

- x, y, a are scalars.
- a scales one number line.
- The sign can reverse direction.

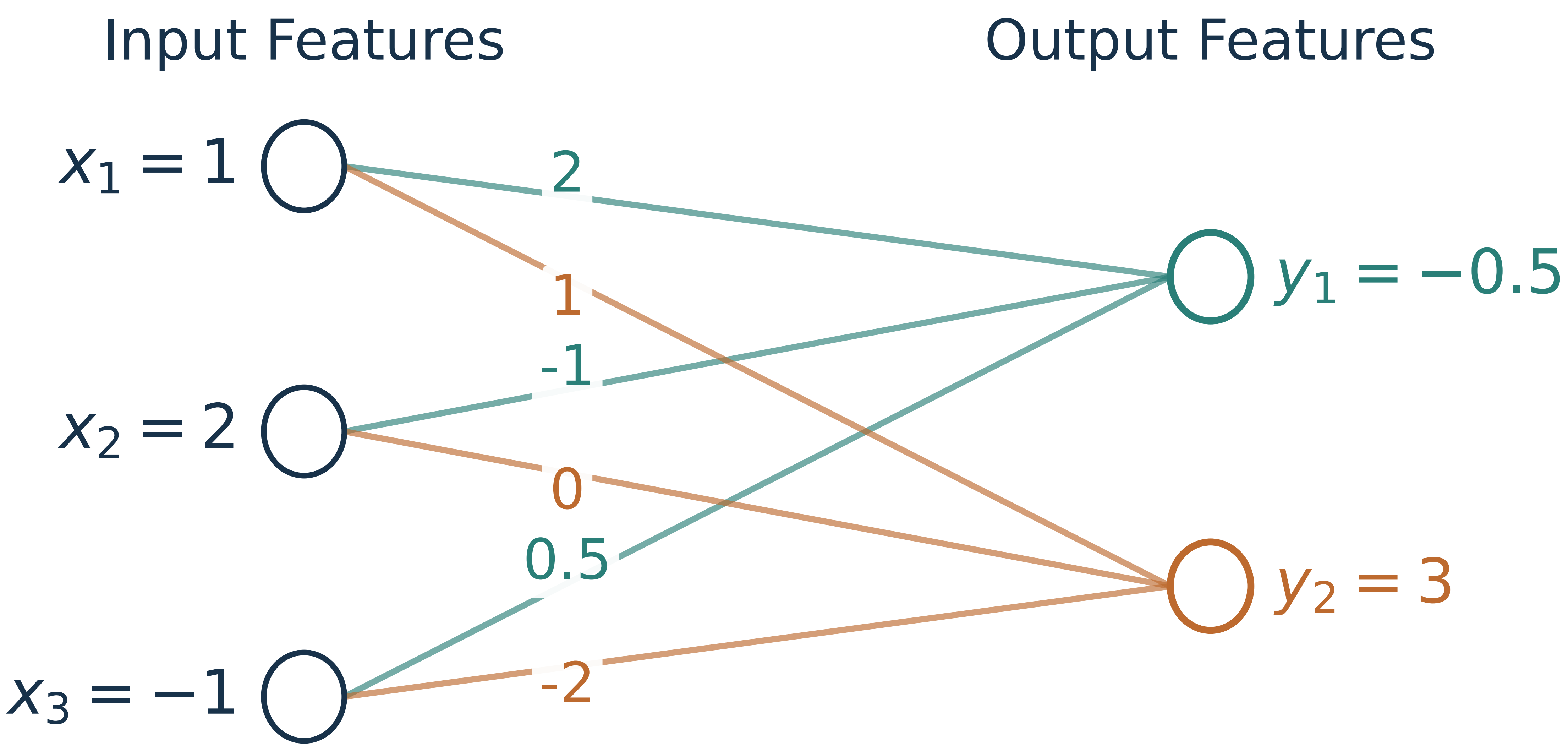
Higher dimensions

$$\boldsymbol{y} = A\boldsymbol{x}$$

- $\boldsymbol{x}$ and $\boldsymbol{y}$ are vectors.
- A can scale different directions differently.
- Input and output can have different dimensions.

a becomes A: one scalar action becomes a transformation of a feature space.

Each output feature receives a weighted connection from every input



$$A = \begin{bmatrix} 2 & -1 & 0.5 \\ 1 & 0 & -2 \end{bmatrix}$$

- A teal edge contributes to y_1.
- An orange edge contributes to y_2.
- Every edge weight is one entry a_{ji}.

Every output is a different linear combination of the inputs

$$\begin{aligned} y_1 &= 2x_1 - x_2 + 0.5x_3 \\ &= 2(1) - 2 + 0.5(-1) \\ &= -0.5 \end{aligned}$$

$$\begin{aligned} y_2 &= x_1 + 0x_2 - 2x_3 \\ &= 1 + 0 - 2(-1) \\ &= 3 \end{aligned}$$

$$\underbrace{\begin{bmatrix} -0.5 \\ 3 \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} 2 & -1 & 0.5 \\ 1 & 0 & -2 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} 1 \\ 2 \\ -1 \end{bmatrix}}_{x}$$

One matrix applies the same transformation to every observation

For observation i:

$$\boldsymbol{y}_i = A\boldsymbol{x}_i, \qquad \boldsymbol{x}_i \in \mathbb{R}^d, \quad \boldsymbol{y}_i \in \mathbb{R}^m.$$

The same entries of A define every output feature for every observation.

- **Convention:** Vectors such as $\boldsymbol{x}_i$ are usually written as columns.
- **Transpose:** A^T swaps rows and columns: $A \in \mathbb{R}^{m \times d} \Rightarrow A^T \in \mathbb{R}^{d \times m}$.

One linear operator gives a consistent representation rule for the entire dataset.

Which batch equation has the right shape?

Observations are rows:

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

Use the same transformation:

$$A = \begin{bmatrix} 2 & -1 & 0.5 \\ 1 & 0 & -2 \end{bmatrix}$$

Compute Y with one matrix multiplication

$$Y = ???$$

$$\underbrace{Y}_{3\times 2} = \underbrace{X}_{3\times 3} \underbrace{A^T}_{3\times 2} = \begin{bmatrix} -0.5 & 3 \\ 0 & -4 \\ 5.5 & 0 \end{bmatrix}$$

Each row of Y is $(A\boldsymbol{x}_i)^T$.

Matrix algebra preserves some scalar rules and rejects others

Valid for scalars and matrices

$$A(B\boldsymbol{x}) = (AB)\boldsymbol{x}$$

$$A(\boldsymbol{x} + \boldsymbol{z}) = A\boldsymbol{x} + A\boldsymbol{z}$$

$$AI = A$$

Order matters for matrices

$$AB \neq BA \quad \text{in general}$$

$$(AB)^T = B^T A^T$$

Changing the order can change the value or make the product undefined.

Dimensions expose many invalid manipulations

Let $A \in \mathbb{R}^{m \times d}$ and $B \in \mathbb{R}^{d \times p}$.

Valid

$$AB \in \mathbb{R}^{m \times p}$$

$$(AB)^T = B^T A^T$$

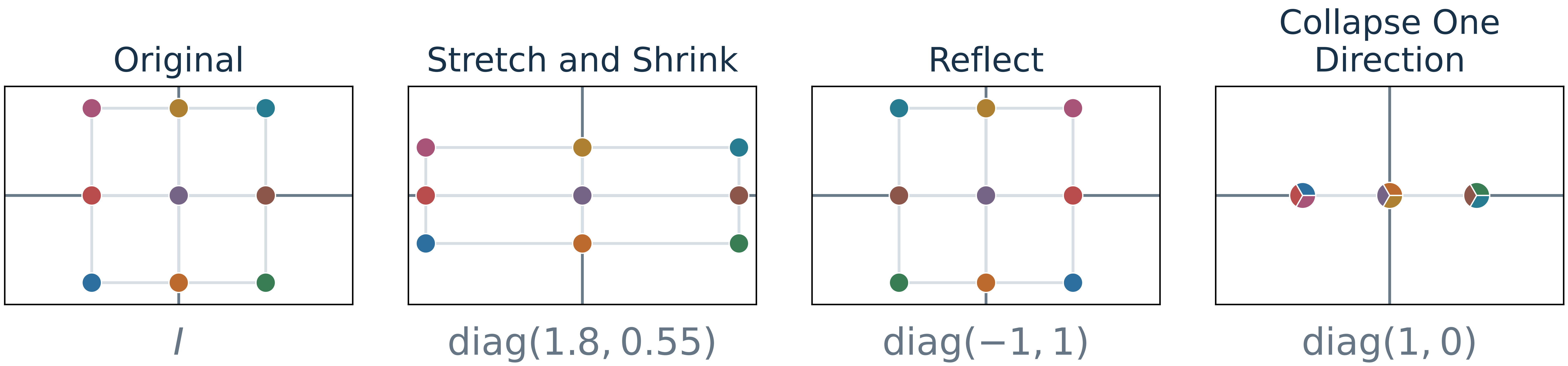
Not guaranteed to exist

$$BA$$

The inner dimensions would require $p = m$.

> **Shape is part of the mathematical claim, not an implementation detail.**

A diagonal matrix applies a separate 1D scaling to each feature



Colors track the same inputs. Three colored sectors at one output location show that **three inputs have collapsed together**.

The Euclidean norm measures size and distance

For $\boldsymbol{x} \in \mathbb{R}^d$:

$$\|\boldsymbol{x}\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_d^2}$$

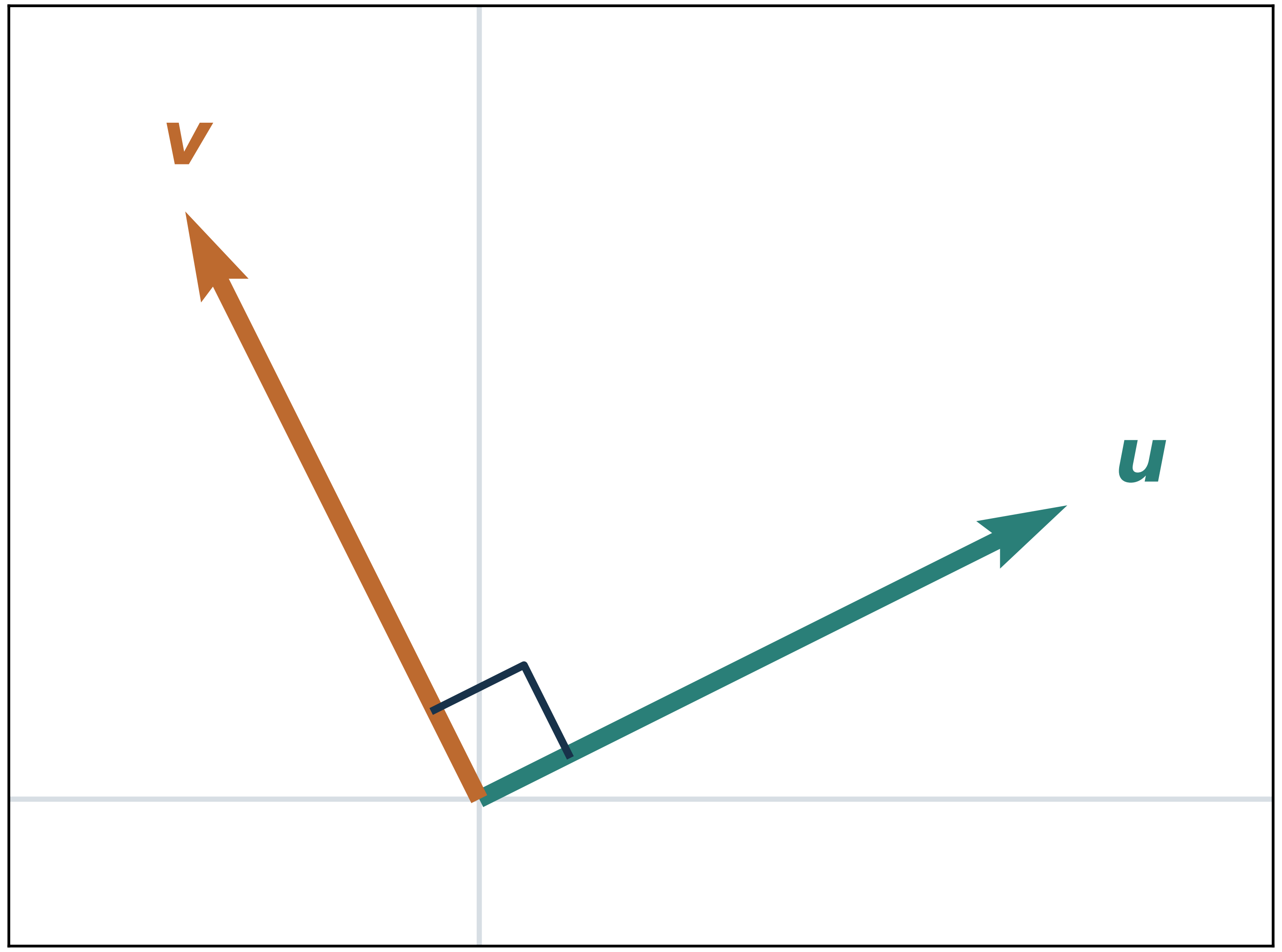
For two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$:

$$\mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2$$

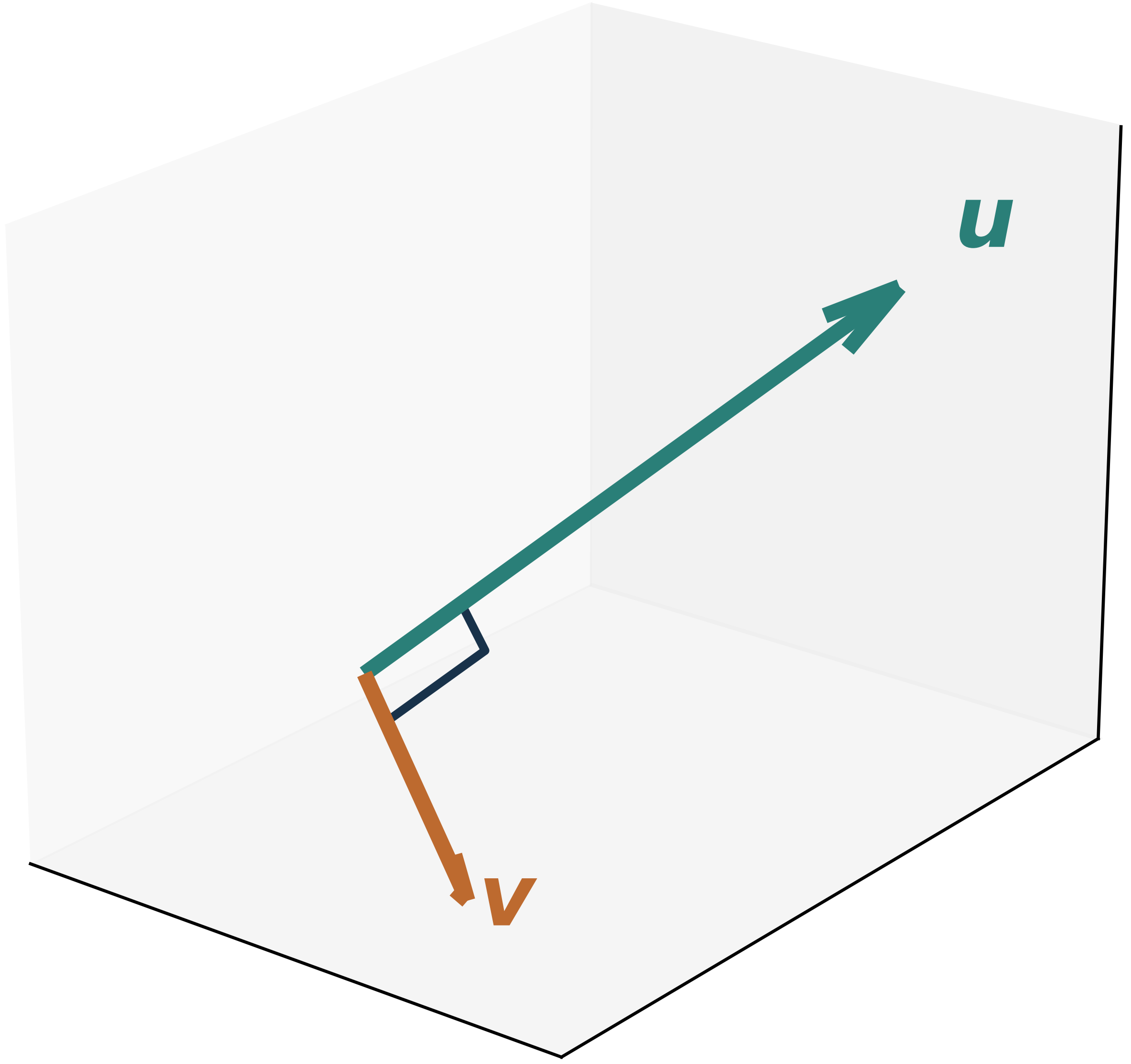
The subscript 2 identifies the Euclidean norm. Other norms measure size differently.

Orthogonal vectors meet at a right angle

2D: $\boldsymbol{u}^T\boldsymbol{v} = 0$



3D: $\boldsymbol{u}^T\boldsymbol{v} = 0$



$$\boldsymbol{u} \perp \boldsymbol{v} \quad \iff \quad \boldsymbol{u}^T\boldsymbol{v} = 0$$

In higher dimensions, we compute orthogonality

Are these dense five-dimensional vectors orthogonal?

$$\boldsymbol{u} = \begin{bmatrix} 1 \\ 2 \\ -1 \\ 3 \\ 4 \end{bmatrix}, \qquad \boldsymbol{v} = \begin{bmatrix} 2 \\ -1 \\ 3 \\ 2 \\ -\frac{3}{4} \end{bmatrix}$$

$$\boldsymbol{u}^T\boldsymbol{v} = 2 - 2 - 3 + 6 - 3 = 0 \qquad \Longrightarrow \qquad \boldsymbol{u} \perp \boldsymbol{v}$$

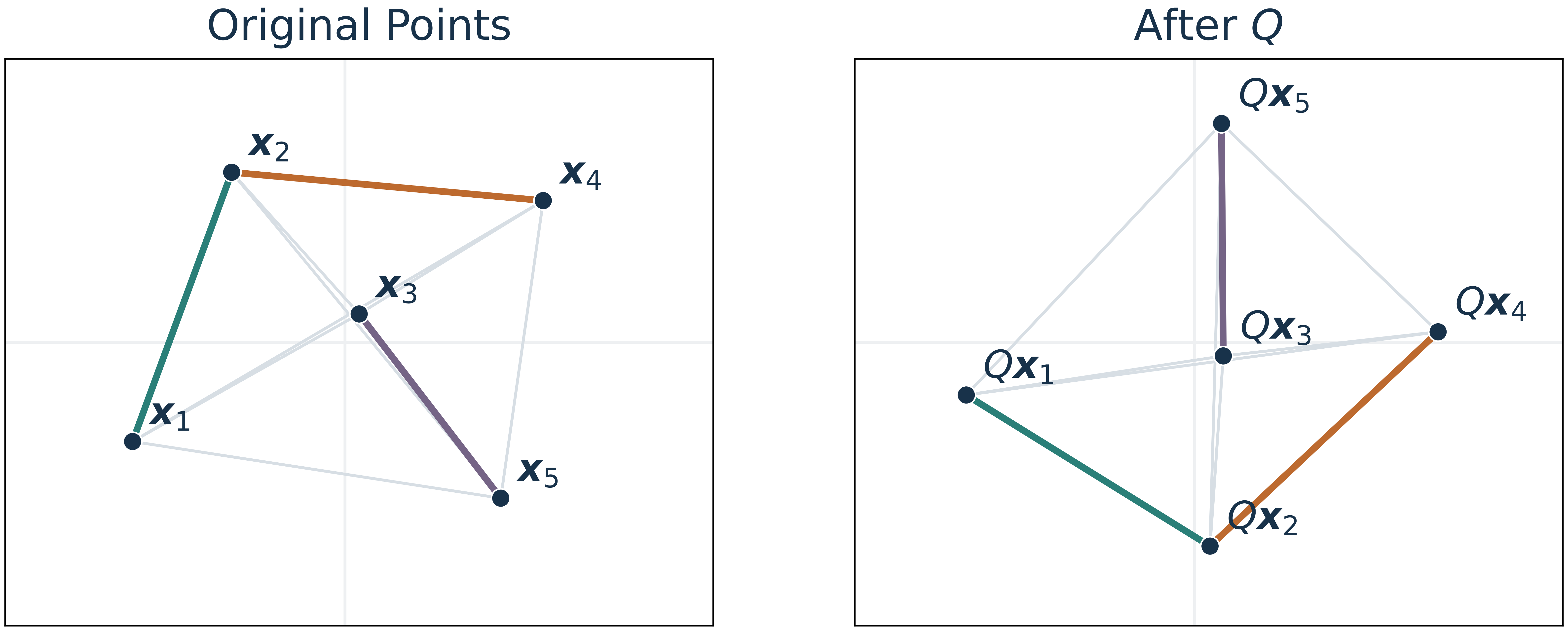
> **Orthogonality is defined in any dimension, even when geometry cannot be visualized.**

An orthogonal matrix has orthonormal columns

For a square matrix $Q = [\boldsymbol{q}_1 \ \cdots \ \boldsymbol{q}_d] \in \mathbb{R}^{d\times d}$, **orthonormal** means mutually orthogonal columns, each of unit length.

Higher dimensions	Interpretation	1D special case	Interpretation
$I\boldsymbol{x} = \boldsymbol{x}$	The identity matrix leaves a vector unchanged.	$1x = x$	The scalar 1 leaves a number unchanged.
$(Q^TQ)_{ij} = \boldsymbol{q}_i^T\boldsymbol{q}_j$	Multiplication collects all column inner products.	$q\,q = q^2$	There is only one entry to multiply by itself.
$Q^TQ = I$	Inner products are 1 for the same column, 0 for distinct columns.	$q^2 = 1$	Hence $q \in \{-1, 1\}$.
$\boldsymbol{y} = Q\boldsymbol{x}$	Rotates or reflects about the origin, preserving length.	$y = qx$	Keeps or reverses direction while preserving length.

Do orthogonal transformations preserve every pairwise distance?



Is $\|Q\boldsymbol{x}_i - Q\boldsymbol{x}_j\|_2$ equal to $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2$ for every pair?

Orthogonality preserves all pairwise Euclidean distances

For any $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^d$ and orthogonal Q:

$$
\begin{aligned}
\|Q\boldsymbol{x}_i - Q\boldsymbol{x}_j\|_2 &= \|Q(\boldsymbol{x}_i - \boldsymbol{x}_j)\|_2 && \text{(Distributivity)} \\
&= \sqrt{(\boldsymbol{x}_i - \boldsymbol{x}_j)^T Q^T Q (\boldsymbol{x}_i - \boldsymbol{x}_j)} && \text{(Euclidean norm)} \\
&= \sqrt{(\boldsymbol{x}_i - \boldsymbol{x}_j)^T (\boldsymbol{x}_i - \boldsymbol{x}_j)} && (Q^T Q = I) \\
&= \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2 && \text{(Euclidean norm)}
\end{aligned}
$$

Diagonal and orthogonal matrices have simple interpretations

Diagonal: Scale Each Coordinate

$$D = \mathrm{diag}(d_1, \dots, d_d), \qquad y_j = d_j x_j$$

- Each coordinate undergoes ordinary scalar multiplication.
- The sign keeps or reverses its direction.
- The magnitude stretches or shrinks it; zero collapses it.

Orthogonal: Rotate or Reflect

$$\boldsymbol{y} = Q\boldsymbol{x}, \qquad Q^T Q = I$$

- Rotate, reflect, or combine both about the origin.
- Keep lengths, angles, and pairwise distances unchanged.
- Change orientation without stretching or shrinking.

In 1D, sign and magnitude explain what multiplication does

One Dimension

For $a \neq 0$:

$$y = ax = \underbrace{\operatorname{sign}(a)}_{\text{keep or reverse}} \quad \underbrace{|a|}_{\text{scale}} \; x$$

For example, $-3x = (-1)(3)x$:

- 3: stretch by a factor of three.
- -1: reverse direction.

Higher Dimensions

$$\boldsymbol{y} = A\boldsymbol{x}$$

We know how to **calculate** the output. But for an arbitrary matrix:

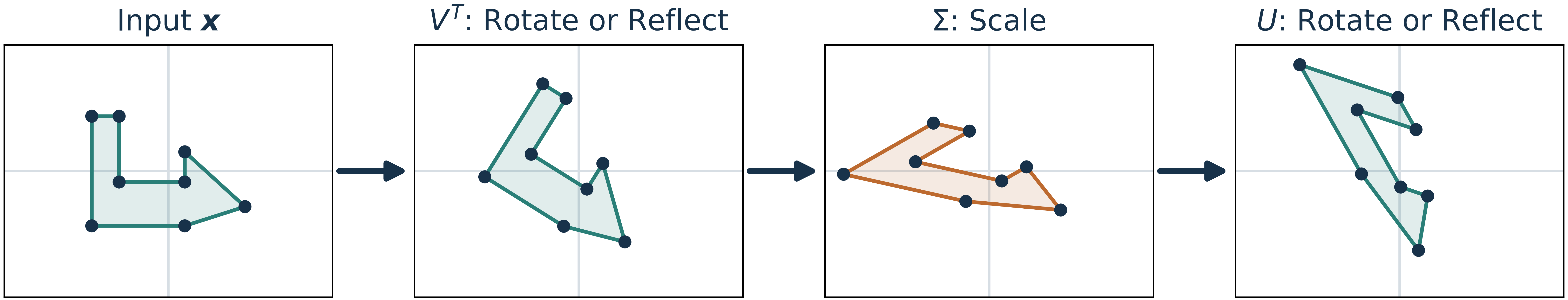
- Which directions change, and how?
- How much does each direction stretch or shrink?
- Can any direction collapse entirely?

Can we decompose any A into rotations or reflections and independent scalings so that we can **understand and analyze** what it does?

SVD separates every linear transformation into three understandable steps

Let $U\Sigma V^T$ be the **singular value decomposition (SVD)** of A, i.e., $A = U\Sigma V^T$.

U and V are orthogonal; Σ is diagonal (possibly rectangular).



$$\boldsymbol{y} = A\boldsymbol{x} = U\Sigma V^T\boldsymbol{x} = U\Big(\Sigma(V^T\boldsymbol{x})\Big)$$

Each SVD factor has a precise shape and role

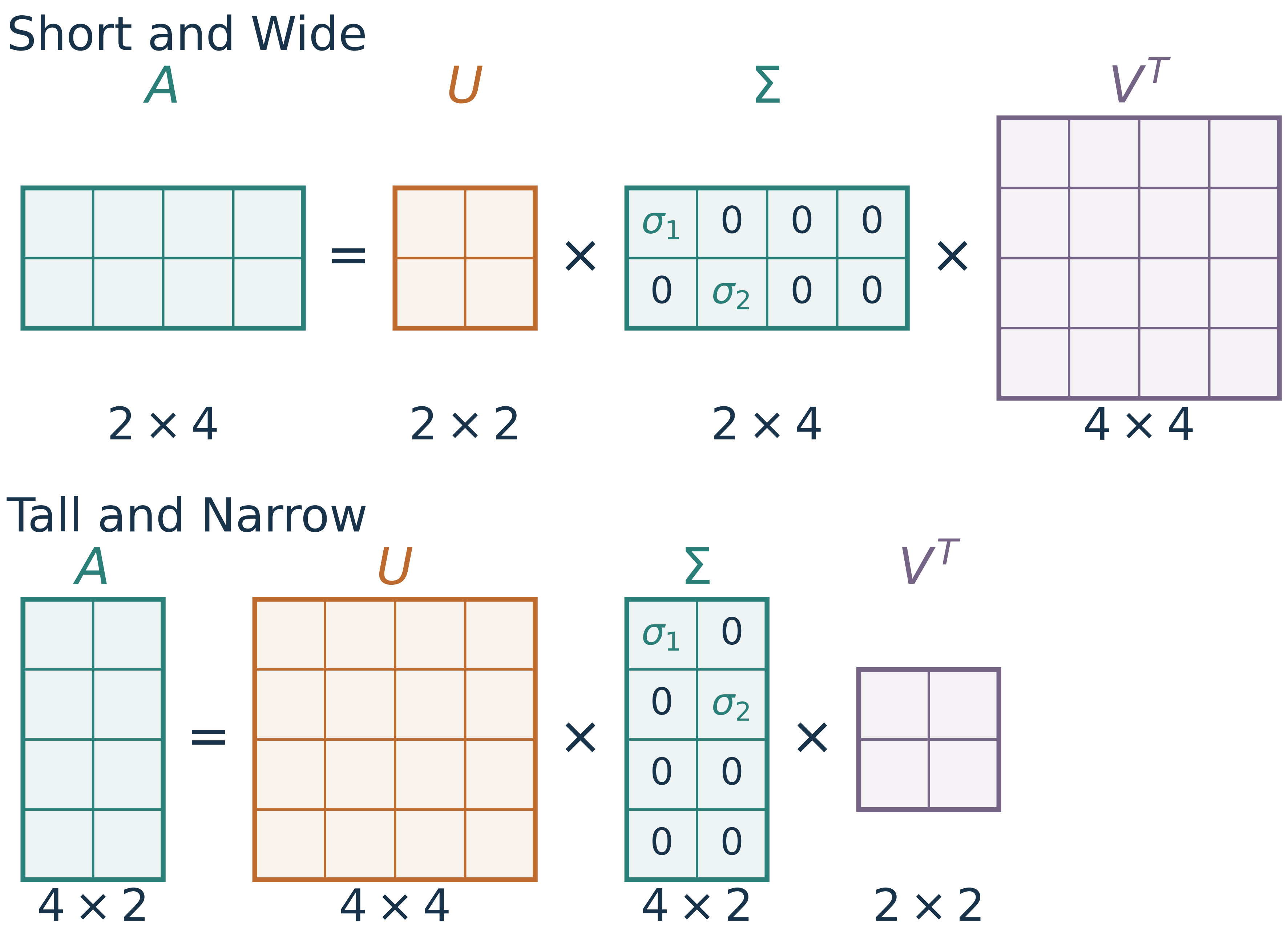
For $A \in \mathbb{R}^{m \times d}$, the **full SVD** is:

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

Factor	Mathematical Property	What It Does
U	Square and orthogonal: $U^TU = I_m$	Rotates or reflects in the output space.
Σ	Same shape as A; $\Sigma_{ij} = 0$ whenever $i \neq j$	Scales directions; zero scalings can collapse them.
V^T	Square and orthogonal: $V^TV = VV^T = I_d$	Rotates or reflects in the input space.

Σ is **rectangular diagonal**: only $\Sigma_{ii} = \sigma_i \geq 0$ may be nonzero, for $i = 1, \dots, \min(m, d)$.

A and Σ share a shape; the orthogonal factors are square



U: square, sized by the **rows** of A.

V^T: square, sized by the **columns** of A.

Singular values tell us how much each direction scales

$j = 2$ Illustrated

$$A = \underbrace{\begin{bmatrix} \boldsymbol{u}_1 & \boldsymbol{u}_2 & \boldsymbol{u}_3 \end{bmatrix}}_{\text{Columns}}^{U} \; \underbrace{\begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix}}_{\text{Diagonal Entries}}^{\Sigma} \; \underbrace{\begin{bmatrix} \boldsymbol{v}_1^T \\ \boldsymbol{v}_2^T \\ \boldsymbol{v}_3^T \end{bmatrix}}_{\text{Rows}}^{V^T}$$

$$\boldsymbol{v}_2 \xrightarrow{\;V^T\;} \underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}}_{\boldsymbol{e}_2} \xrightarrow{\;\Sigma\;} \underbrace{\begin{bmatrix} 0 \\ \sigma_2 \\ 0 \end{bmatrix}}_{\sigma_2 \boldsymbol{e}_2} \xrightarrow{\;U\;} \sigma_2 \boldsymbol{u}_2$$

Select Column 2

$$V^T \boldsymbol{v}_j = \boldsymbol{e}_j \quad \Longrightarrow \quad A\boldsymbol{v}_j = U(\Sigma \boldsymbol{e}_j) = \sigma_j \boldsymbol{u}_j, \qquad j \leq \min(m, d).$$

The largest singular value gives the maximum stretching

Order the singular values: $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$.

$$\|A\boldsymbol{x}\|_2 \leq \sigma_1 \|\boldsymbol{x}\|_2, \qquad \|A\|_2 := \max_{\|\boldsymbol{x}\|_2 = 1} \|A\boldsymbol{x}\|_2 = \sigma_1.$$

- V^T and U preserve lengths.
- Each coordinate scaling in Σ is at most σ_1.
- The input $\boldsymbol{v}_1$ attains the largest scaling.

Rank counts the independent directions that survive

$$\operatorname{rank}(A) = \#\{j : \sigma_j > 0\} \le \min(m, d)$$

For example:

$$\Sigma = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad \Longrightarrow \qquad \operatorname{rank}(A) = 2$$

- Two independent input directions produce nonzero output directions.
- The third input direction disappears: $A\boldsymbol{v}_3 = \mathbf{0}$.
- Rotations and reflections change which directions these are, but not the rank.

A null direction makes different inputs indistinguishable

The null space contains all directions that map to zero:

$$\operatorname{null}(A) = \{\boldsymbol{h} : A\boldsymbol{h} = \mathbf{0}\}.$$

For any nonzero $\boldsymbol{h} \in \operatorname{null}(A)$:

$$\begin{aligned} A(\boldsymbol{x} + \boldsymbol{h}) &= A\boldsymbol{x} + A\boldsymbol{h} && \text{(Distributivity)} \\ &= A\boldsymbol{x} && \text{(Null direction)} \end{aligned}$$

A square operator is invertible when no direction collapses

In 1D, $y = ax$ can be undone by $x = y/a$ exactly when $a \neq 0$.

For square $A = U\Sigma V^T \in \mathbb{R}^{d\times d}$:

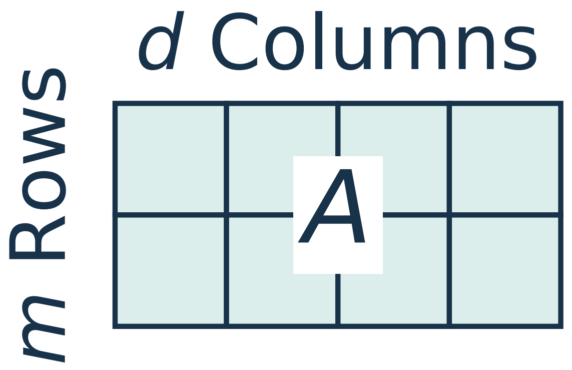
$$\text{all } \sigma_j > 0 \quad \Longrightarrow \quad A^{-1} = V\Sigma^{-1}U^T, \qquad \Sigma^{-1} = \operatorname{diag}(1/\sigma_1, \dots, 1/\sigma_d).$$

$$\begin{aligned}
A^{-1}A &= (V\Sigma^{-1}U^T)(U\Sigma V^T) && \text{(Substitute)} \\
&= V\Sigma^{-1}\Sigma V^T && (U^TU = I_d) \\
&= VV^T && (\Sigma^{-1}\Sigma = I_d) \\
&= I_d && (VV^T = I_d)
\end{aligned}$$

Recovery depends on preserving every input direction

$y = Ax$ maps d input coordinates to m output coordinates.

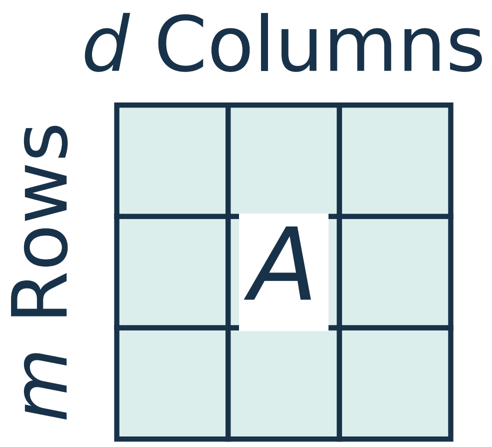
$m < d$



Recovery Impossible

Some Input Direction Is Lost

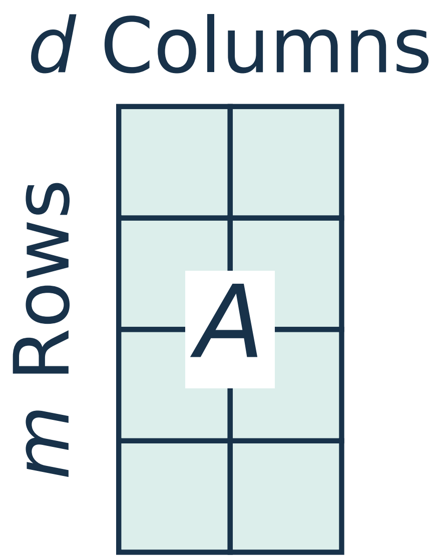
$m = d$



Recovery If $\mathrm{rank}(A) = d$

Every Input Direction Must Survive

$m > d$



Recovery If $\mathrm{rank}(A) = d$

Extra Outputs Can Be Redundant

Recovering **every** $x \in \mathbb{R}^d$ requires $\mathrm{rank}(A) = d.$

The pseudoinverse undoes every nonzero scaling

For **any** $A = U\Sigma V^T \in \mathbb{R}^{m\times d}$, define:

$$\underbrace{A^\dagger}_{d\times m} = \underbrace{V}_{d\times d}\,\underbrace{\Sigma^\dagger}_{d\times m}\,\underbrace{U^T}_{m\times m}.$$

$\Sigma^\dagger$ has the **transposed shape** of Σ, with zero off-diagonal entries:

$$(\Sigma^\dagger)_{jj} = \begin{cases} 1/\sigma_j, & \sigma_j > 0, \\ 0, & \sigma_j = 0. \end{cases}$$

- Reverse the output reorientation, undo nonzero scalings, then reorient back.
- A zero scaling destroyed information; leaving it zero cannot restore it.
- The **Moore–Penrose pseudoinverse** exists even when A has no inverse.

The pseudoinverse recovers inputs exactly when no input direction is lost

For $\boldsymbol{y} = A\boldsymbol{x}$, recovery of **every** input means:

$$\widehat{\boldsymbol{x}} = A^\dagger \boldsymbol{y} = \boldsymbol{x} \text{ for all } \boldsymbol{x} \in \mathbb{R}^d \quad \Longleftrightarrow \quad A^\dagger A = I_d.$$

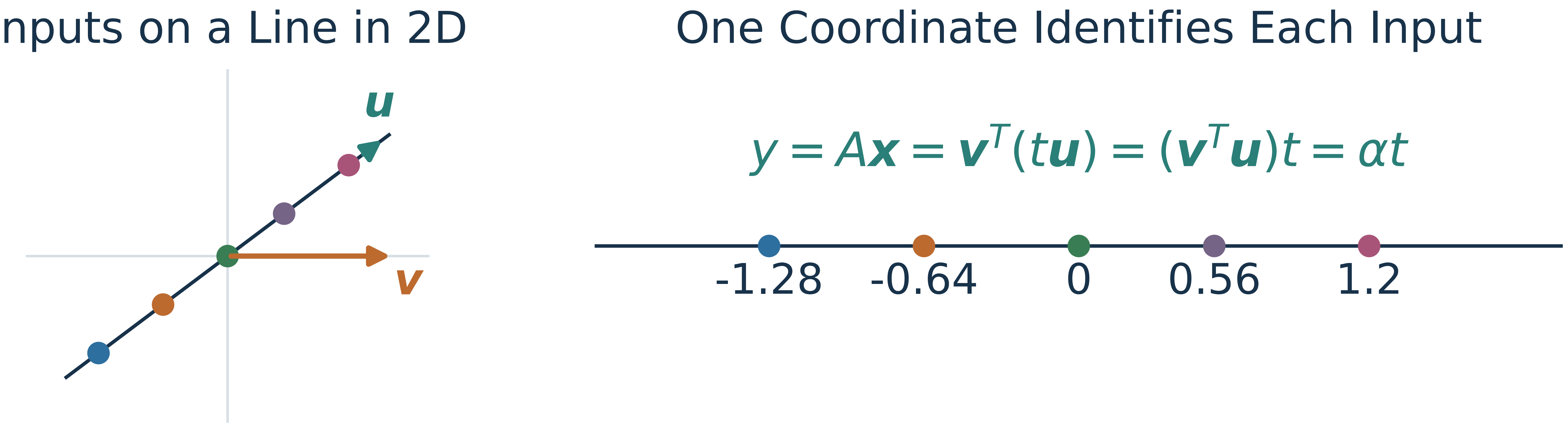
Let $r = \operatorname{rank}(A)$ and $V_r = [\boldsymbol{v}_1 \ \cdots \ \boldsymbol{v}_r]$ contain its surviving input directions.

$$\begin{aligned}
A^\dagger A &= V\Sigma^\dagger U^T U \Sigma V^T && \text{(Substitute)} \\
&= V(\Sigma^\dagger \Sigma) V^T && (U^T U = I_m) \\
&= V_r V_r^T && \text{(Keep nonzero directions)}
\end{aligned}$$

$\Sigma^\dagger \Sigma$ has r diagonal ones and $d - r$ zeros. Thus $A^\dagger A = I_d$ **exactly when** $r = d$: full column rank, requiring $m \geq d$.

A restricted input subspace can be **recovered exactly**

Inputs: $\boldsymbol{x} = t\boldsymbol{u} \in \mathbb{R}^2$, $\|\boldsymbol{u}\|_2 = 1$. Encoder: $A = \boldsymbol{v}^T$, with $\alpha = \boldsymbol{v}^T\boldsymbol{u} \neq 0$.



Decode with $B = \boldsymbol{u}/\alpha$: knowing the line lets us undo its nonzero scaling.

$$\widehat{\boldsymbol{x}} = By = \frac{\boldsymbol{u}}{\alpha}(\alpha t) = t\boldsymbol{u} = \boldsymbol{x}.$$

If $\boldsymbol{v} = \boldsymbol{u}$, then $\alpha = 1$, $y = t$, and $B = A^\dagger = \boldsymbol{u}$. **In general, $B \neq A^\dagger$.**

Read what an operator does from its singular values

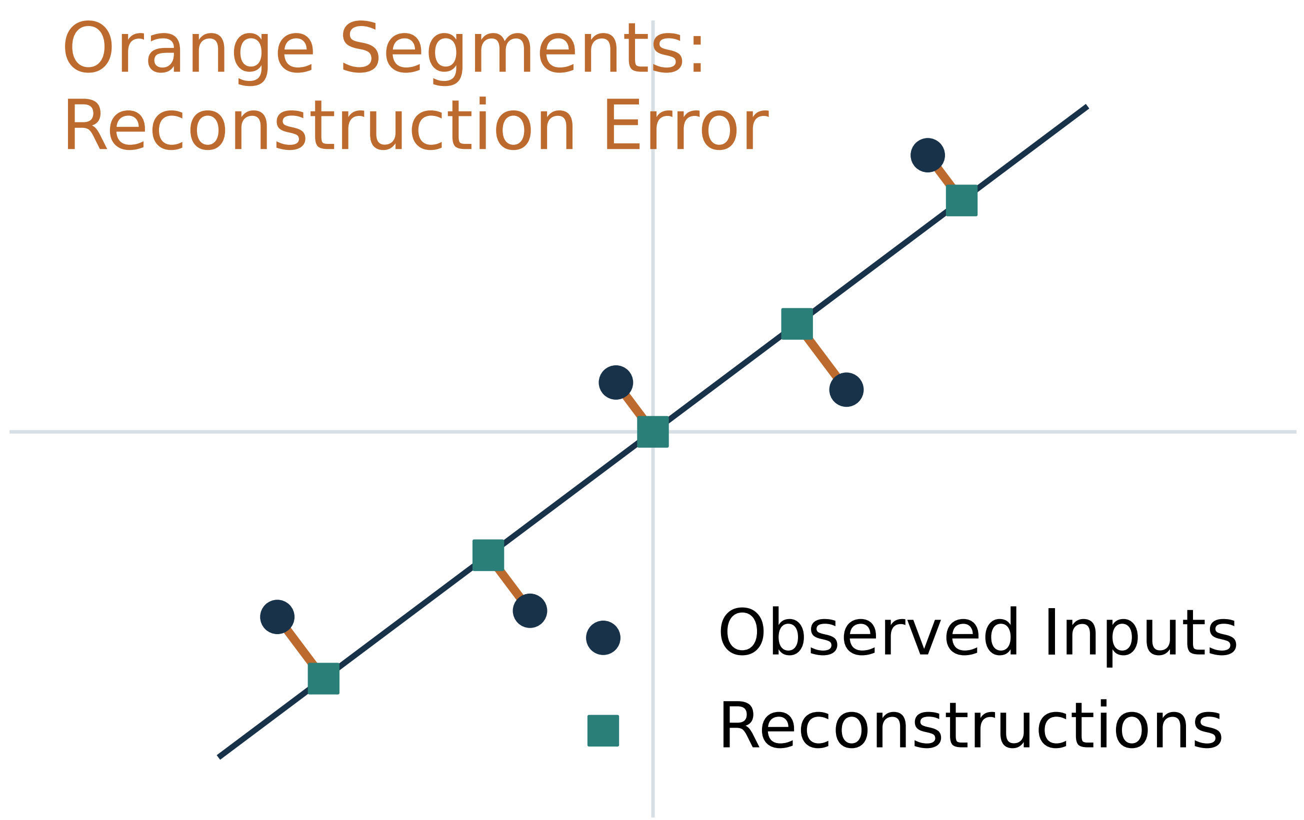
A sensor-processing system uses $A = U\,\mathrm{diag}(4, 1, 0)V^T$.

> **Can you recover every sensor input? What is the largest possible stretching factor? Could this operator preserve every pairwise distance? Explain which input direction is invisible to the system.**

Rank 2; recovery is impossible in general; maximum scaling is 4. The direction $\boldsymbol{v}_3$ disappears. All distances cannot be preserved.

Data near a subspace can have small reconstruction error

What if the inputs are **close to** a line, but do not lie exactly on it?



$$y = \boldsymbol{u}^T \boldsymbol{x}, \qquad \widehat{\boldsymbol{x}} = \boldsymbol{u} y, \qquad \text{error} = \|\boldsymbol{x} - \widehat{\boldsymbol{x}}\|_2^2.$$

Which direction minimizes total reconstruction error?