Estimating Transforms

Computer Vision: ~~CS652~~

CS473/CSE573

Last few classes we looked at transforms.

Image Transforms

Transform	Matrix	Degrees of Freedom	Invariants
Scaling	$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	2	Angles when s_x=s_y Parallel Lines
Rotation	$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	1	Lengths Angles Parallel Lines
Translation	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$	2	Lengths Angles Parallel Lines
Reflection	$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	0	Lengths Parallel Lines
Shear	$\begin{bmatrix} 1 & r_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	1	Parallel Lines
Affine Transform	$\begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix}$	6	Parallel Lines
Homographies (Projective Transforms)	$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$	8	Cross Ratio

Now we will see how to **estimate transforms**.

Image Transforms

Transform	Matrix	Degrees of Freedom	Invariants
Scaling	$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	2	Angles when s_x=s_y Parallel Lines
Rotation	$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	1	Lengths Angles Parallel Lines
Translation	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$	2	Lengths Angles Parallel Lines
Reflection	$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	0	Lengths Parallel Lines
Shear	$\begin{bmatrix} 1 & r_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	1	Parallel Lines
Affine Transform	$\begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix}$	6	Parallel Lines
Homographies (Projective Transforms)	$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$	8	Cross Ratio

Assignment 2

Panaroma Stitching By Estimating Transforms

Inputs: Images Taken By Rotating Camera About Center

Assignment 2

Panaroma Stitching By Estimating Transforms

Inputs: Images Taken By Rotating Camera About Center

Need Overlapping Points!

Assignment 2

Panaroma Stitching By Estimating Transforms

Output: Image(s) Warped to Match and Stitched

Assignment 2: Steps

Start with Input Images

Assignment 2: Steps

Mark Corresponding Points

Also Called **Correspondences**

Assignment 2: Steps

Mark Corresponding Points

Also Called **Correspondences**

In this assignment, you will mark correspondences automatically.

Assignment 2: Steps

Mark Corresponding Points

Also Called **Correspondences**

Later, we will see how the automatic method works.

Assignment 2: Steps

Used Correspondences to Estimate Transform

Topic of Today!

Assignment 2: Steps

Transform One Image to Match the Other Image

Assignment 2: Steps

Transform One Image to Match the Other Image

Assignment 2: Steps

Expand the Other Image To Make All the Same Size

Assignment 2: Steps

Blend Images

Assignment 2: Steps

Blend Images

Assignment 2: Steps

Used Correspondences to Estimate Transform

Topic of Today!

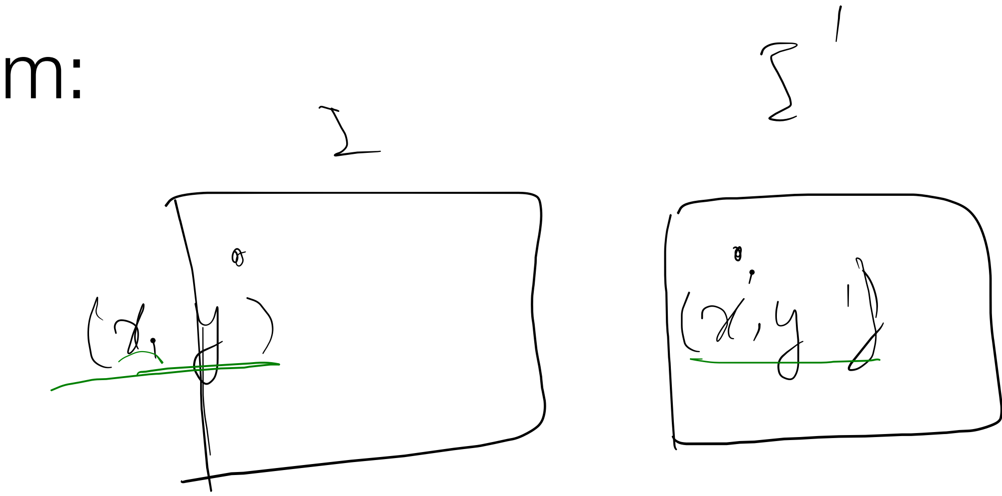
Estimate Transform From Correspondences

Set up System of Linear Equations to Estimate Transform

We will look at two methods

1) Use **least squares** with the following system:

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad x' = \frac{\hat{x}'}{\hat{w}'}, \quad y' = \frac{\hat{y}'}{\hat{w}'}$$



2) Use **homogeneous least squares** with the following system:

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad x' = \frac{\hat{x}'}{\hat{w}'}, \quad y' = \frac{\hat{y}'}{\hat{w}'}$$

We will look at two methods

1) Use **least squares** with the following system:

$$\begin{bmatrix}\hat{x}'\\ \hat{y}'\\ \hat{w}'\end{bmatrix} = \begin{bmatrix}a & b & c\\ d & e & f\\ g & h & 1\end{bmatrix}\begin{bmatrix}x\\ y\\ 1\end{bmatrix}, x' = \frac{\hat{x}'}{\hat{w}'}, y' = \frac{\hat{y}'}{\hat{w}'}$$

Specific form of homography matrix with last element 1.

2) Use **homogeneous least squares** with the following system:

$$\begin{bmatrix}\hat{x}'\\ \hat{y}'\\ \hat{w}'\end{bmatrix} = \begin{bmatrix}a & b & c\\ d & e & f\\ g & h & i\end{bmatrix}\begin{bmatrix}x\\ y\\ 1\end{bmatrix}, x' = \frac{\hat{x}'}{\hat{w}'}, y' = \frac{\hat{y}'}{\hat{w}'}$$

General form of homography matrix

Note the difference!

1) Least Squares

1) Use **least squares** with the following system:

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, x' = \frac{\hat{x}'}{\hat{w}'}, y' = \frac{\hat{y}'}{\hat{w}'}$$

Specific form of homography matrix with last element 1.

$$\begin{bmatrix}\hat{x}'\\ \hat{y}'\\ \hat{w}'\end{bmatrix} = \begin{bmatrix} a & b & c\\ d & e & f\\ g & h & 1\end{bmatrix}\begin{bmatrix}x\\ y\\ 1\end{bmatrix}, \quad x' = \frac{\hat{x}'}{\hat{w}'}, \quad y' = \frac{\hat{y}'}{\hat{w}'}$$

$$\hat{x}' = ax + by + c$$
$$\hat{y}' = dx + ey + f$$
$$\hat{w}' = gx + hy + 1$$

$$x' = \frac{ax+by+c}{gx+hy+1} \quad (1)$$

$$y' = \frac{dx+ey+f}{gx+hy+1} \quad (2)$$

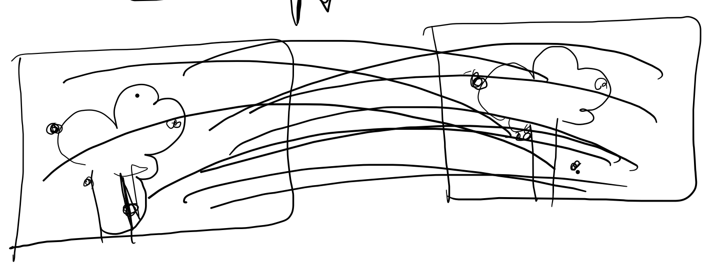
DIRECT LINEAR TRANSFORM (DLT)

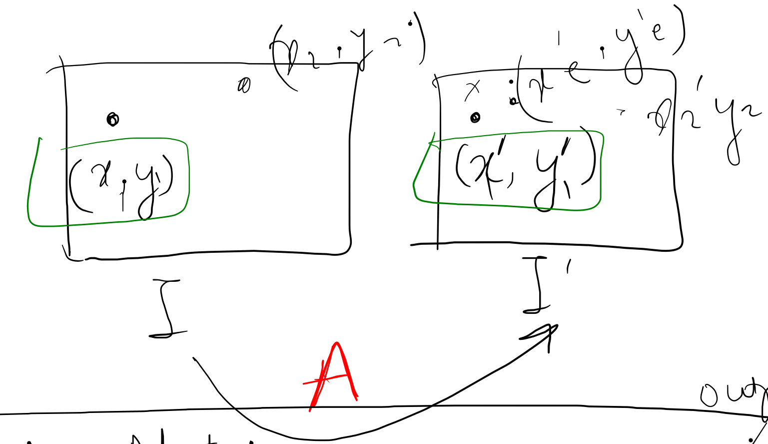
(1) → $\underbrace{x'(gx+hy+1)}_{L} = \underbrace{ax+by+c}_{R}$

$$x'gx + x'hy + x' = ax + by + c$$
$$x'gx + x'hy = -x'$$
$$-ax - by - c + x'gx + x'hy = -x'$$
$$(-x)a + (-y)b + (-1)c + (0)\cdot d + (0)\cdot e + (0)\cdot f + (x'x)g + (x'y)h = -x'$$

(2) →

I N I'





Design Matrix

output

$$\begin{bmatrix} -x_1 & -y_1 & -1 & 0 & 0 & 0 & x_1'x_1 & x_1'y_1\\ 0 & 0 & 0 & -x_1 & -y_1 & -1 & y_1'x_1 & y_1'y_1\\ -x_2 & -y_2 & -1 & 0 & 0 & 0 & x_2'x_2 & x_2'y_2\\ 0 & 0 & 0 & -x_2 & -y_2 & -1 & y_2'x_2 & y_2'y_2\\ \vdots & & & & & & & \vdots \end{bmatrix}_{2N \times 8} \begin{bmatrix} a\\ b\\ c\\ d\\ e\\ f\\ g\\ h \end{bmatrix}_{8\times 1} = \begin{bmatrix} -x_1'\\ -y_1'\\ -x_2'\\ -y_2'\\ \vdots \end{bmatrix}_{2N\times 1}$$

8 columns, 2N rows.

Set up system of equations by creating design matrix

Convert

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, x' = \frac{\hat{x}'}{\hat{w}'}, y' = \frac{\hat{y}'}{\hat{w}'}$$

to the form

$$\begin{bmatrix} \quad \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} \, \end{bmatrix}$$

Design Matrix

Set up system of equations by creating design matrix

Convert

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, x' = \frac{\hat{x}'}{\hat{w}'}, y' = \frac{\hat{y}'}{\hat{w}'}$$

to the form

$$\begin{bmatrix} \quad\quad\quad\quad\quad\quad \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} \; \end{bmatrix}$$

Design Matrix

Two equations are not enough! How many do we need? We need to make the matrix square at the very least.

Set up system of equations by creating design matrix

Use many correspondences

$$(x_1, y_1) \rightarrow (x'_1, y'_1), (x_2, y_2) \rightarrow (x'_2, y'_2), ...$$

to set up

$$\begin{bmatrix} & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$$

Set up system of equations by creating design matrix

Use many correspondences

$$(x_1, y_1) \rightarrow (x_1', y_1'), (x_2, y_2) \rightarrow (x_2', y_2'), ...$$

to set up

$$\begin{bmatrix} & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$$

Need
at least 4
correspondences

Set up system of equations by creating design matrix

Use many correspondences

$$(x_1, y_1) \rightarrow (x'_1, y'_1), (x_2, y_2) \rightarrow (x'_2, y'_2), \ldots$$

to set up

$$\begin{bmatrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{bmatrix}$$

Need at least 4 correspondences

More is better for numerical stability

$$\underbrace{\begin{bmatrix} \\ P \\ \\ \end{bmatrix}}_{2N \times 8} \underbrace{\begin{bmatrix} q \\ \\ \end{bmatrix}}_{8 \times 1} = \underbrace{\begin{bmatrix} \\ r \\ \\ \end{bmatrix}}_{2N \times 1}$$

$$\mathbf{P}\mathbf{q} = \mathbf{r}$$

We know **P** and **r**.

P is a tall matrix.

Find **q**.

$$\mathbf{P}\mathbf{q} = \mathbf{r}$$

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

What is **q**?

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

The best

$$5q_1 + q_2 \approx 6$$
$$q_1 \approx 6$$
$$3q_1 + 4q_2 \approx 0$$

$$\mathbf{P}\mathbf{q} = \mathbf{r}$$

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Ideally, **q** should be a 2x1 vector that if you multiply **P** with **q**, you get **r**.

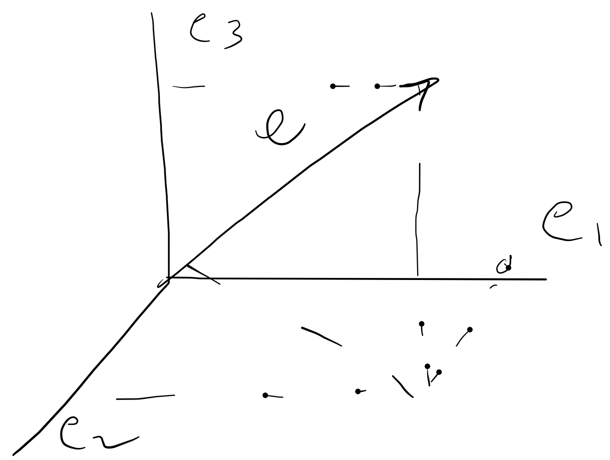
$$\mathbf{P}\mathbf{q} = \mathbf{r}$$

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Ideally, **q** should be a 2x1 vector that if you multiply **P** with **q**, you get **r**.

In practice, that may never happen.

$$\mathbf{Pq} = \mathbf{r}$$



$$\left(\sqrt{e_1^2 + e_2^2 + e_3^2}\right)^2$$

$$\begin{array}{c}P\\ \begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix}\end{array} \mathbf{q} = \begin{array}{c}r\\ \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}\end{array}$$

$r' = Pq^*$

$r' \sim r$

$\| r' - r \|^2$

$e = r' - r$

$\| e \|^2 = \| r' - r \|^2$

Instead, we will find a **q***,
that gives **r'**=**Pq***,
where **r'** is **the closest possible vector** to **r**.

$$\mathbf{Pq} = \mathbf{r}$$

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

What does it mean for **r**’ to be the closest possible vector to **r**?

$$\mathbf{Pq} = \mathbf{r}$$

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

$e_1^2 + e_2^2 + e_3^2$

What does it mean for **r**’ to be the closest possible vector to **r**?

$\|e\|^2 = \|r' - r\|^2$

least squared distance.

Distance between **r**’ and **r** has to be minimum!

$$\mathbf{P}\mathbf{q} = \mathbf{r}$$

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Easier to see this **geometrically**.

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\mathbf{P}\mathbf{q} = \mathbf{r}$$

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Each column is a 3 x 1 vector,
so the vectors live in a **3D**.

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Let us draw them.

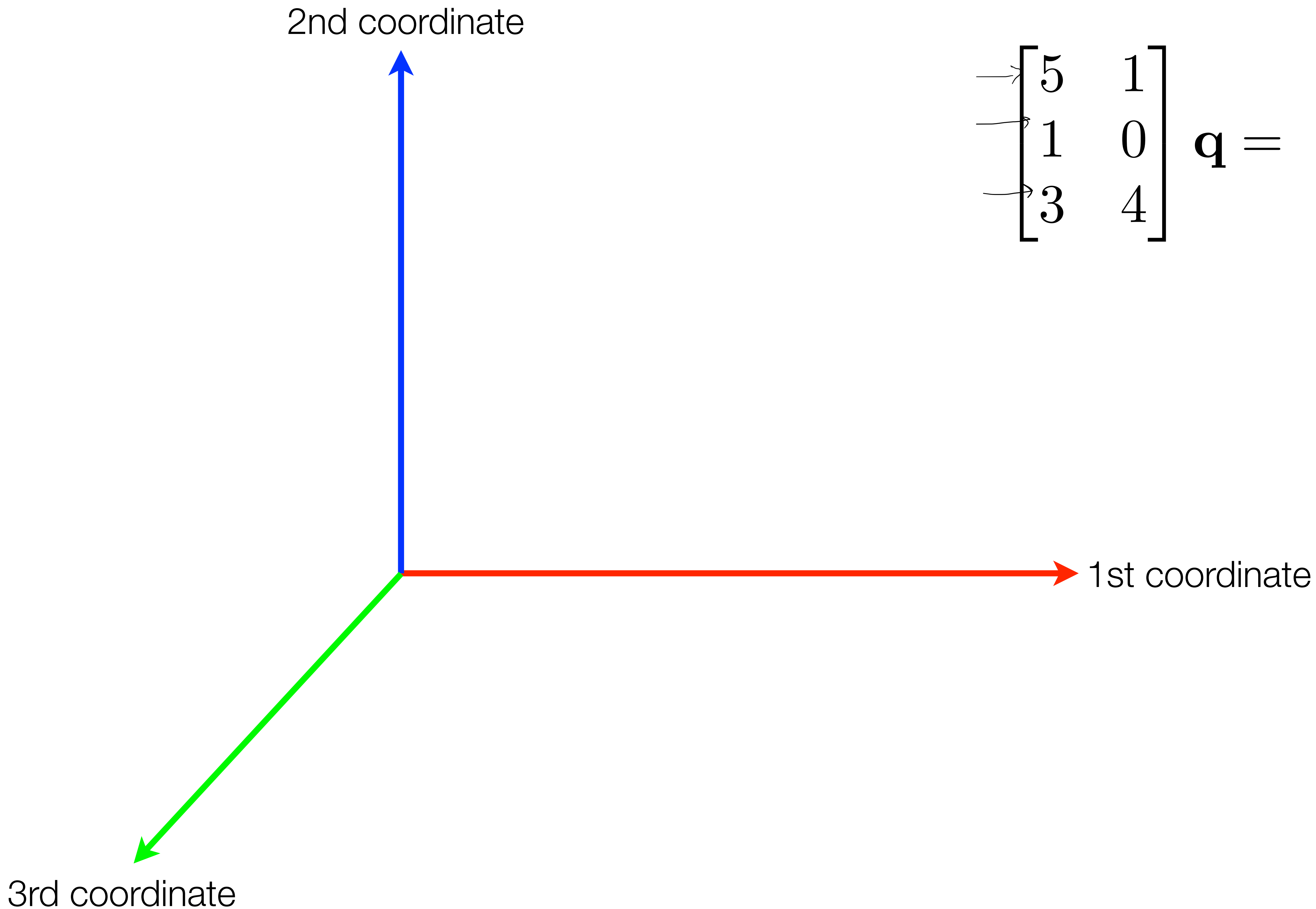
Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Geometric Representation of Equations

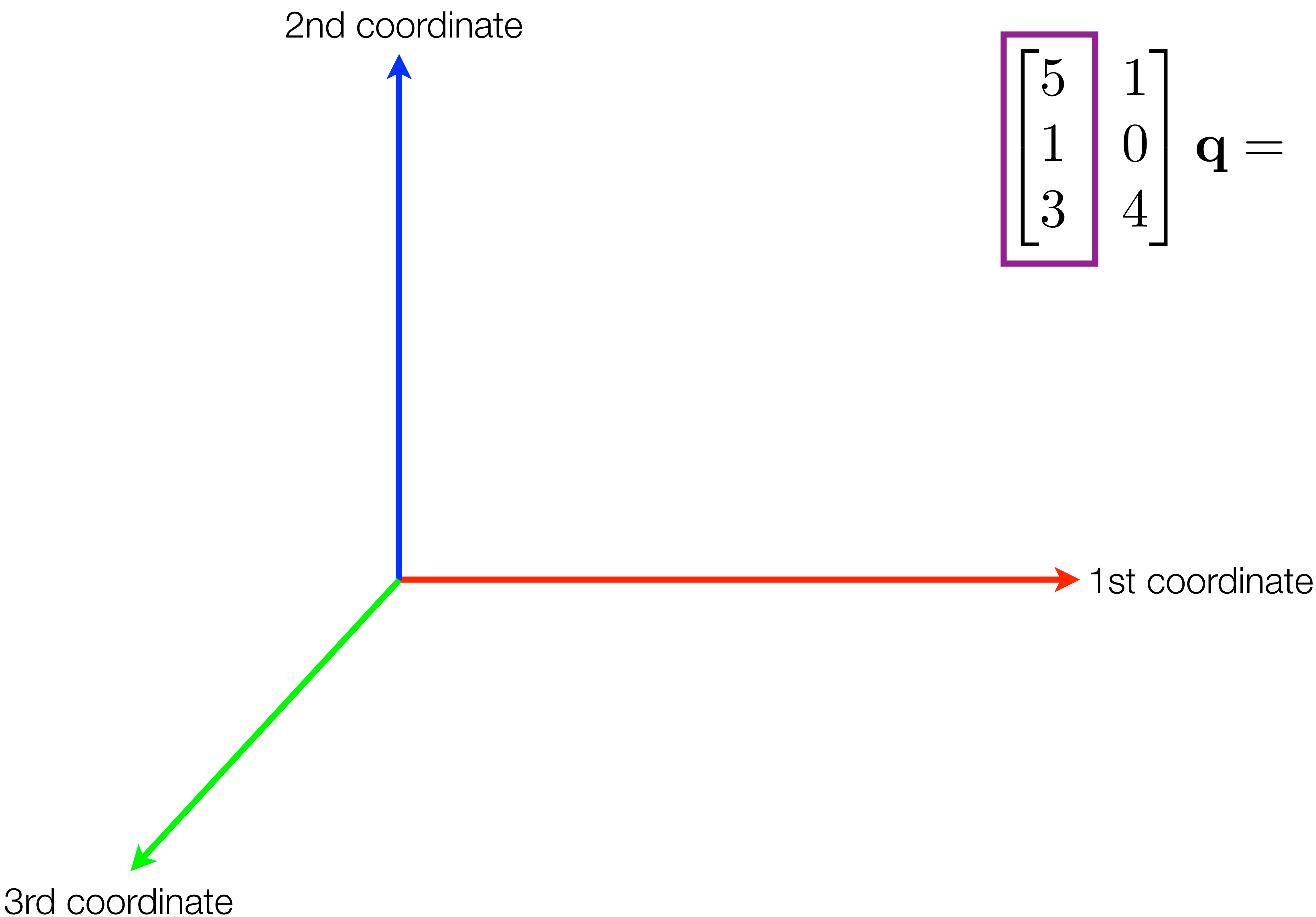
Columns of **P** are a basis for a subspace



$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Geometric Representation of Equations

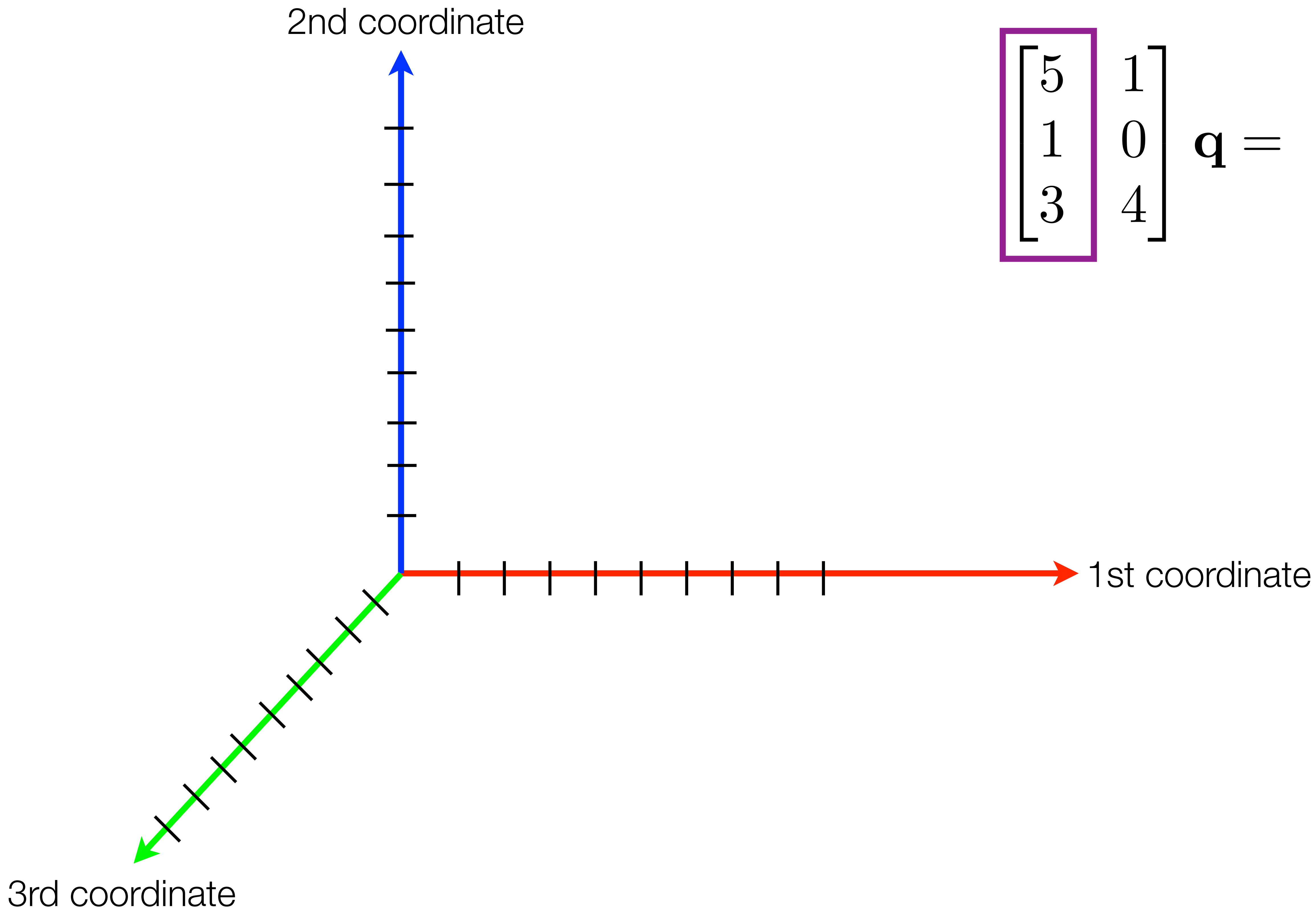
Columns of **P** are a basis for a subspace



$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Geometric Representation of Equations

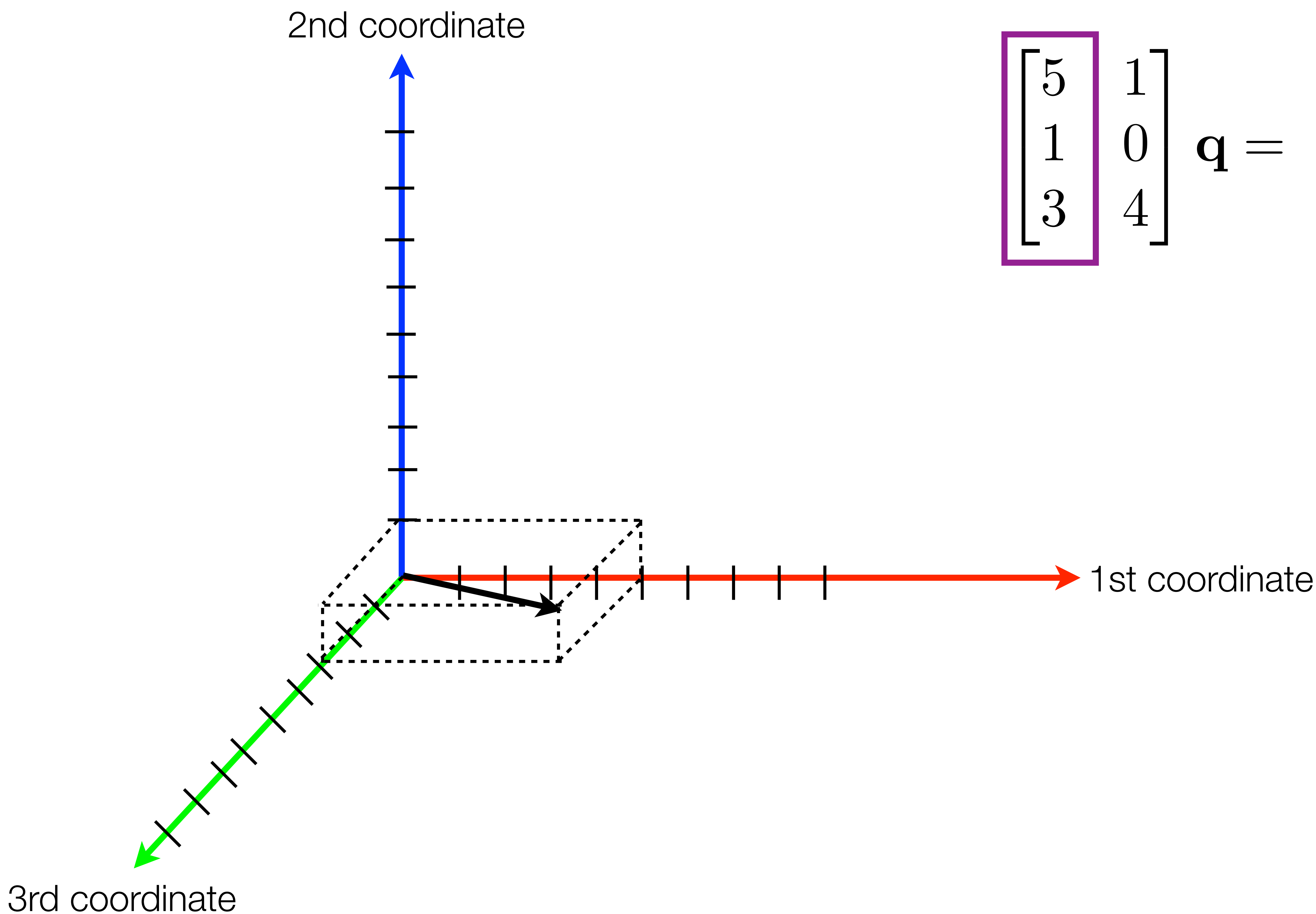
Columns of **P** are a basis for a subspace



$$\begin{bmatrix} \boxed{\begin{matrix} 5 \\ 1 \\ 3 \end{matrix}} & \begin{matrix} 1 \\ 0 \\ 4 \end{matrix} \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

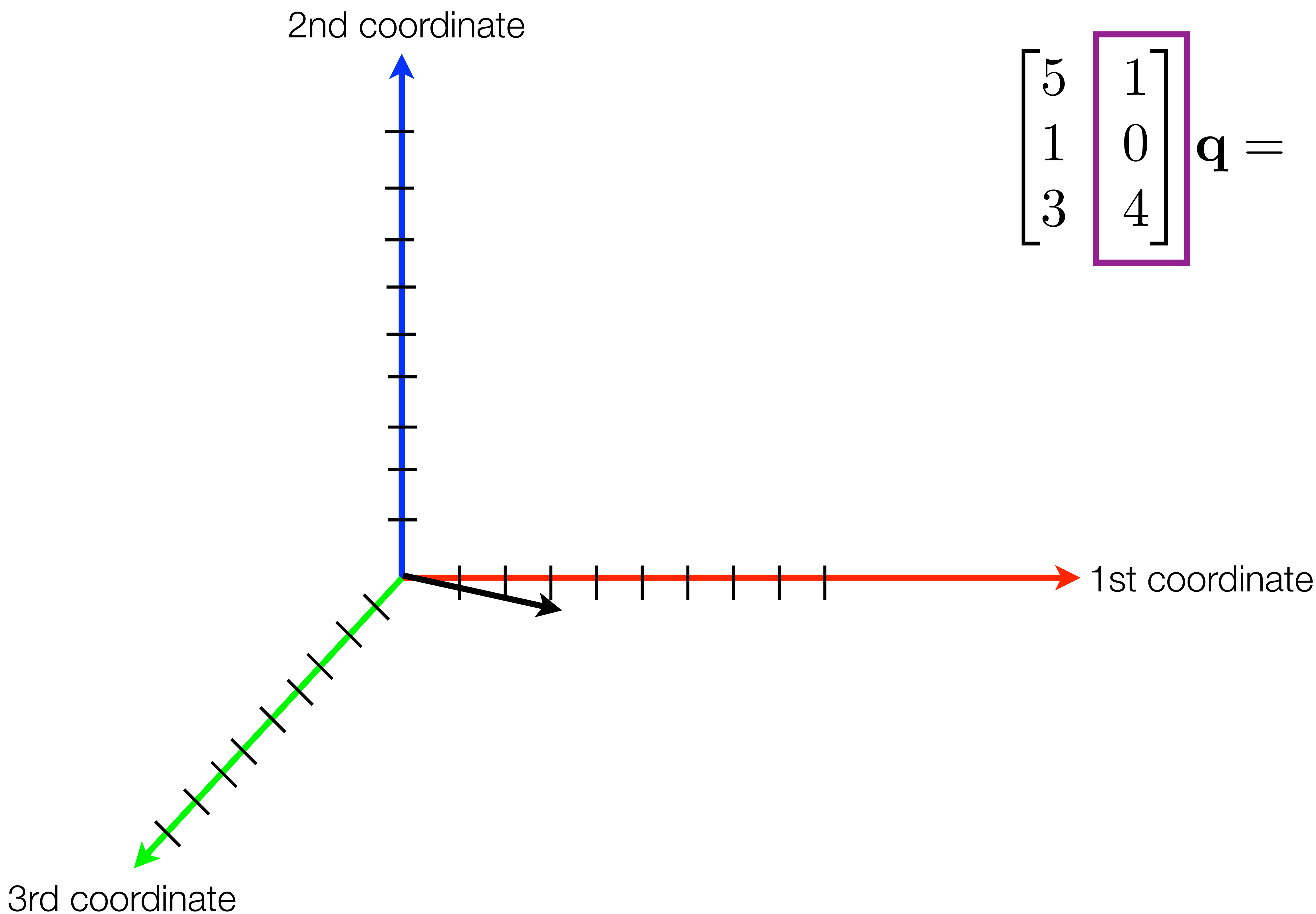


$$\begin{bmatrix} \boxed{\begin{matrix} 5 \\ 1 \\ 3 \end{matrix}} & \begin{matrix} 1 \\ 0 \\ 4 \end{matrix} \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Geometric Representation of Equations

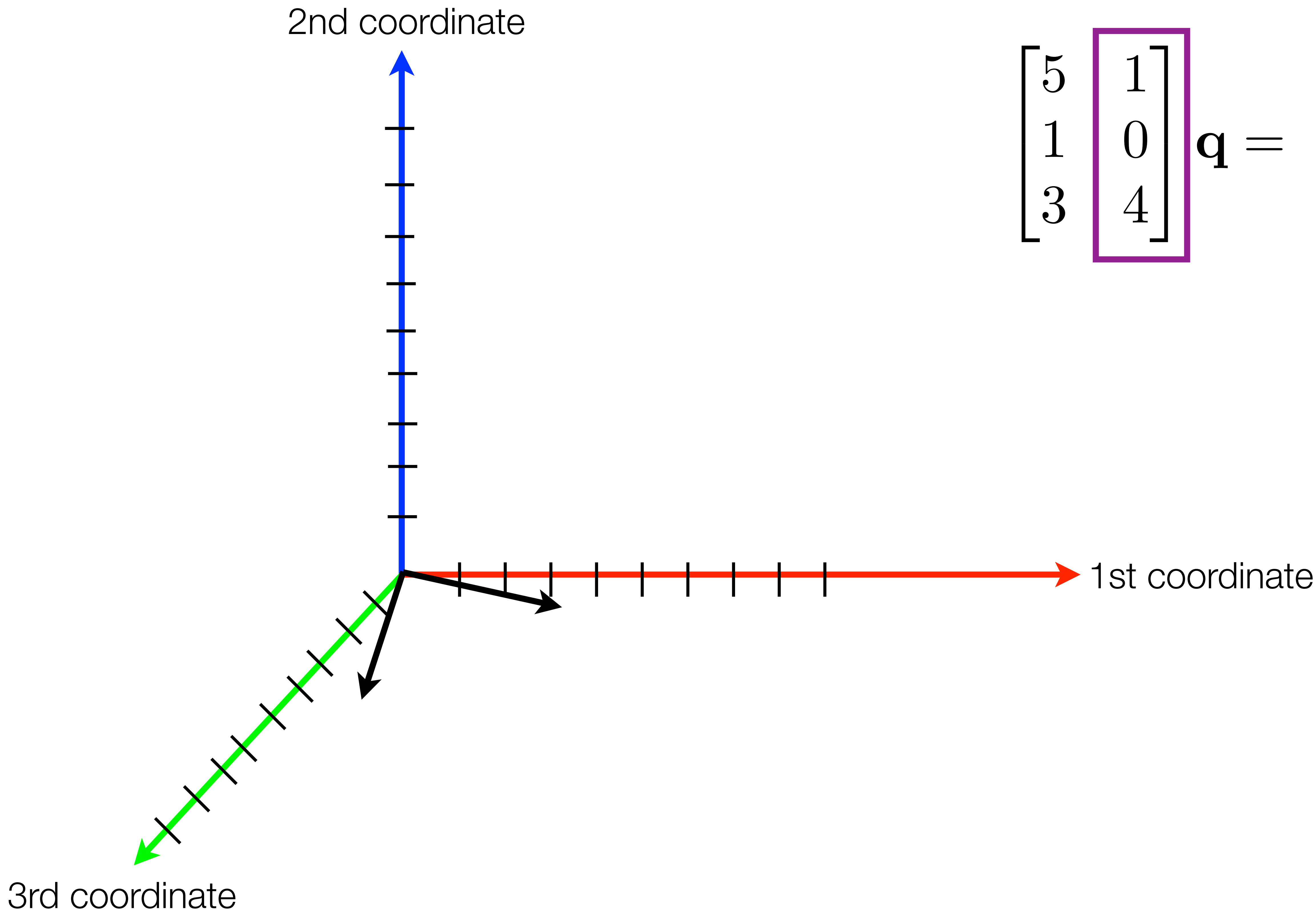
Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & \boxed{1} \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$



Geometric Representation of Equations

Columns of **P** are a basis for a subspace

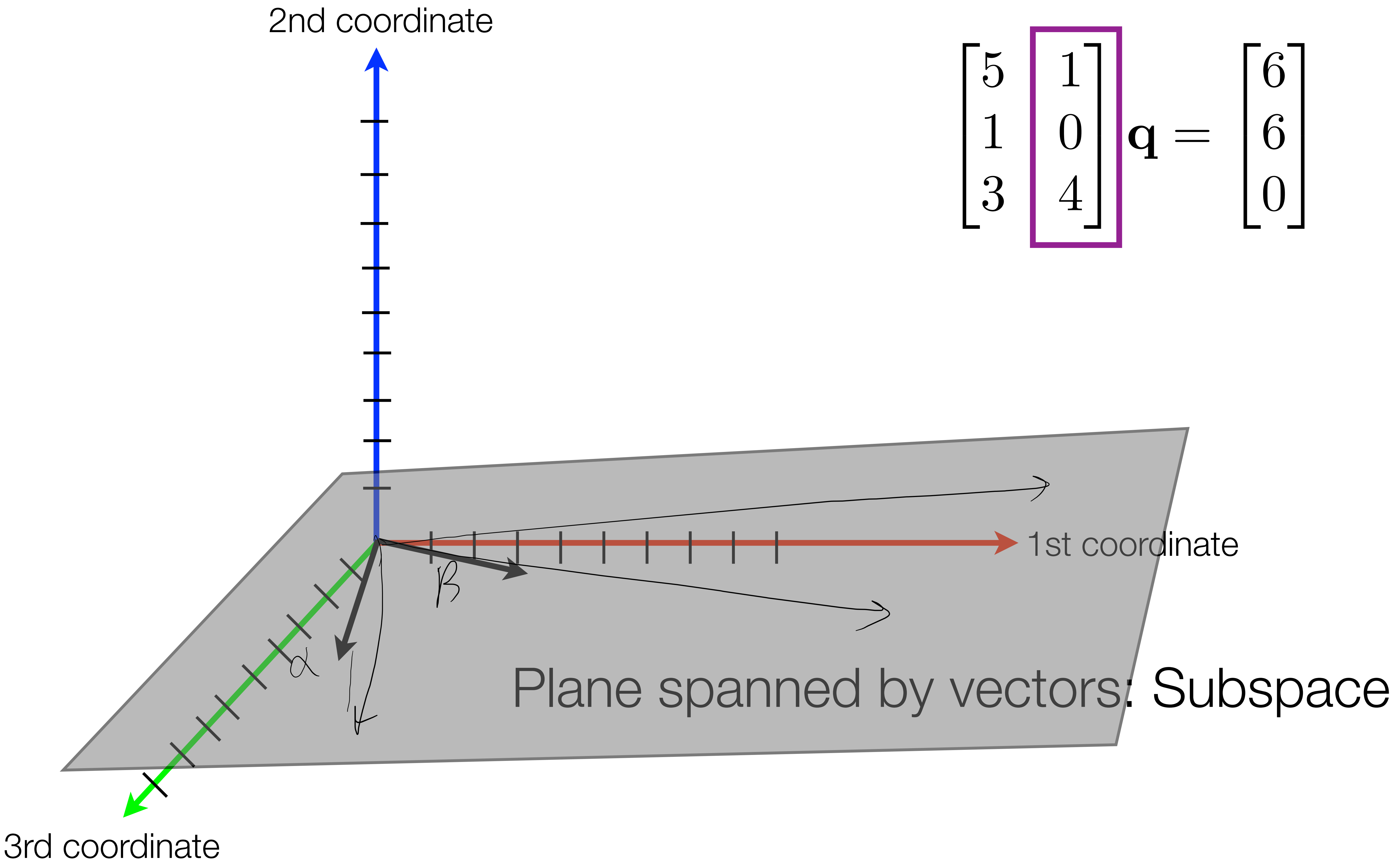


$$\begin{bmatrix} 5 & \boxed{\begin{matrix} 1 \\ 0 \\ 4 \end{matrix}} \\ 1 & \\ 3 & \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

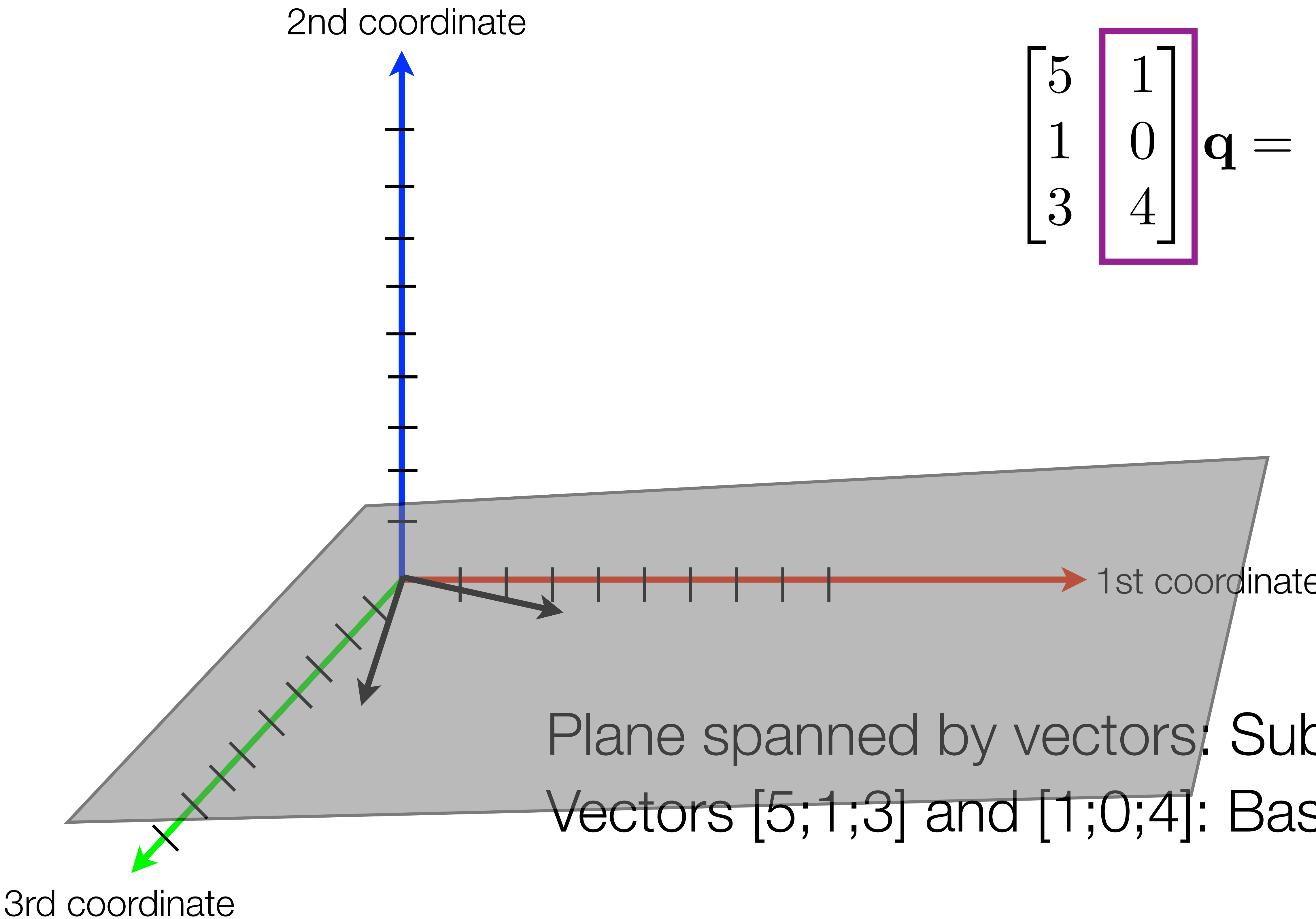
$$\begin{bmatrix} 5 & \boxed{\begin{matrix} 1 \\ 0 \\ 4 \end{matrix}} \\ 1 & \\ 3 & \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$



Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$



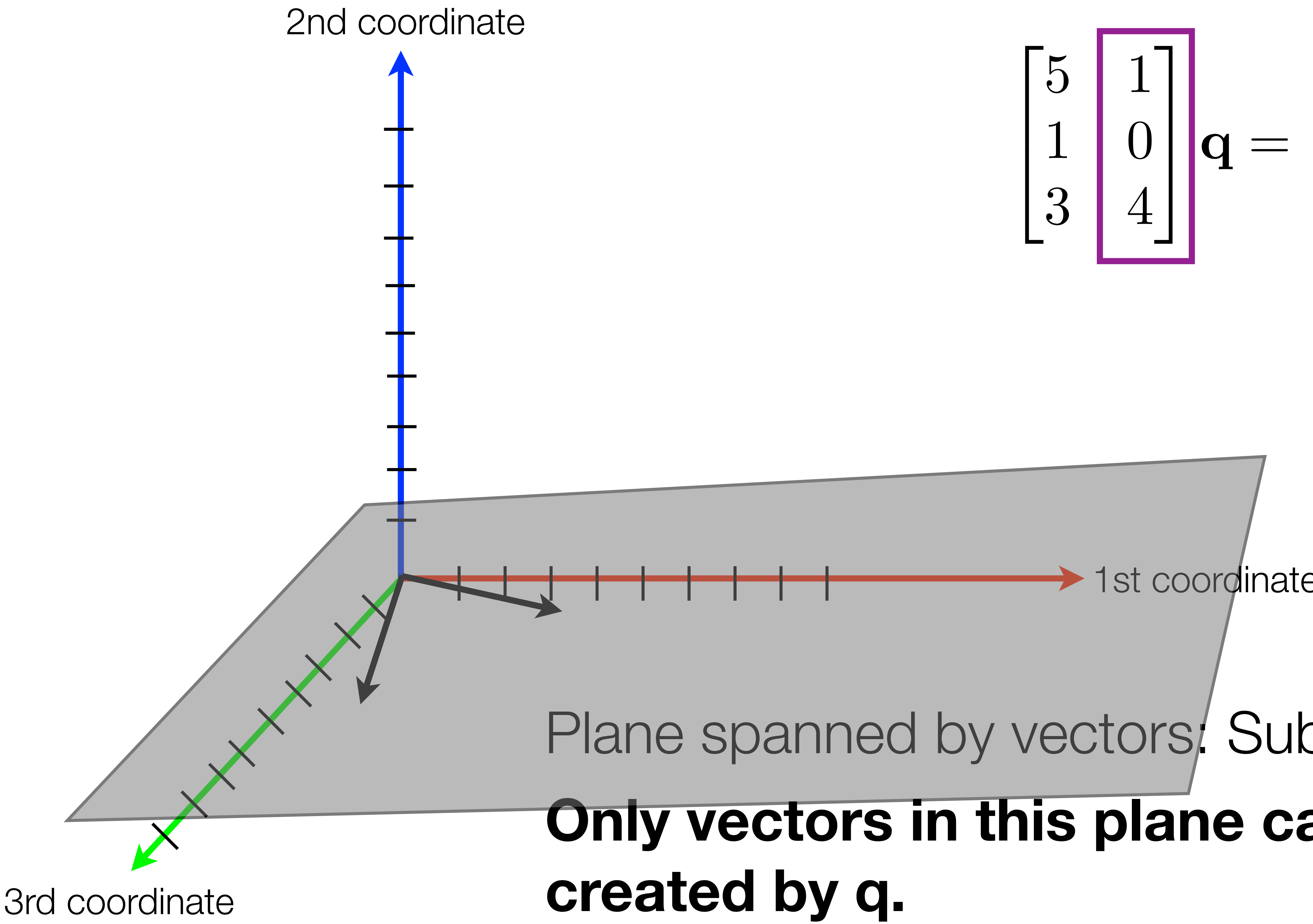
Plane spanned by vectors: Subspace

Vectors [5;1;3] and [1;0;4]: Basis

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$



Plane spanned by vectors: Subspace

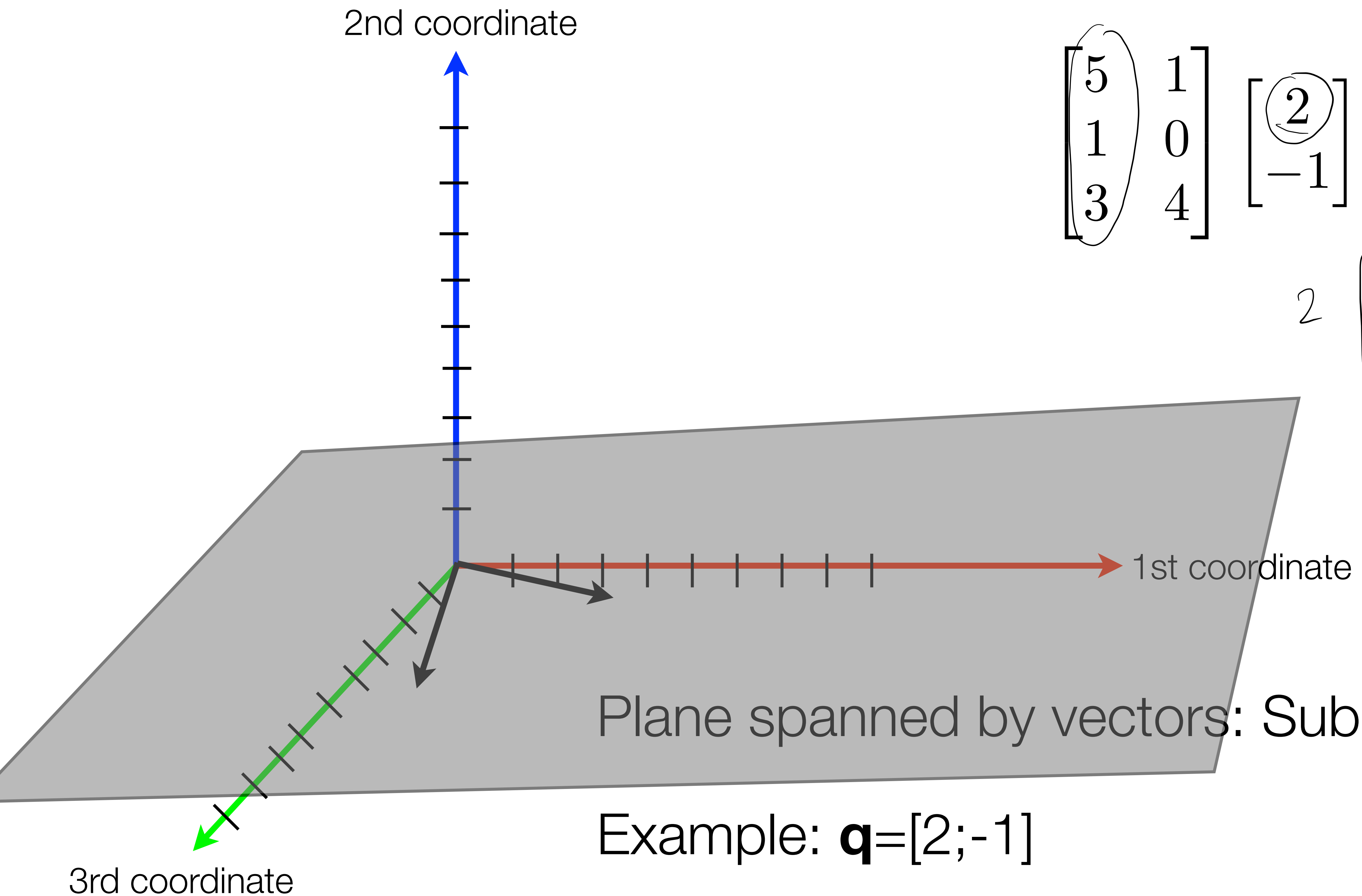
Only vectors in this plane can be created by q.

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

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

$$2\begin{bmatrix}5\\ 1\\ 3\end{bmatrix} + (-1)\begin{bmatrix}1\\ 0\\ 4\end{bmatrix}$$



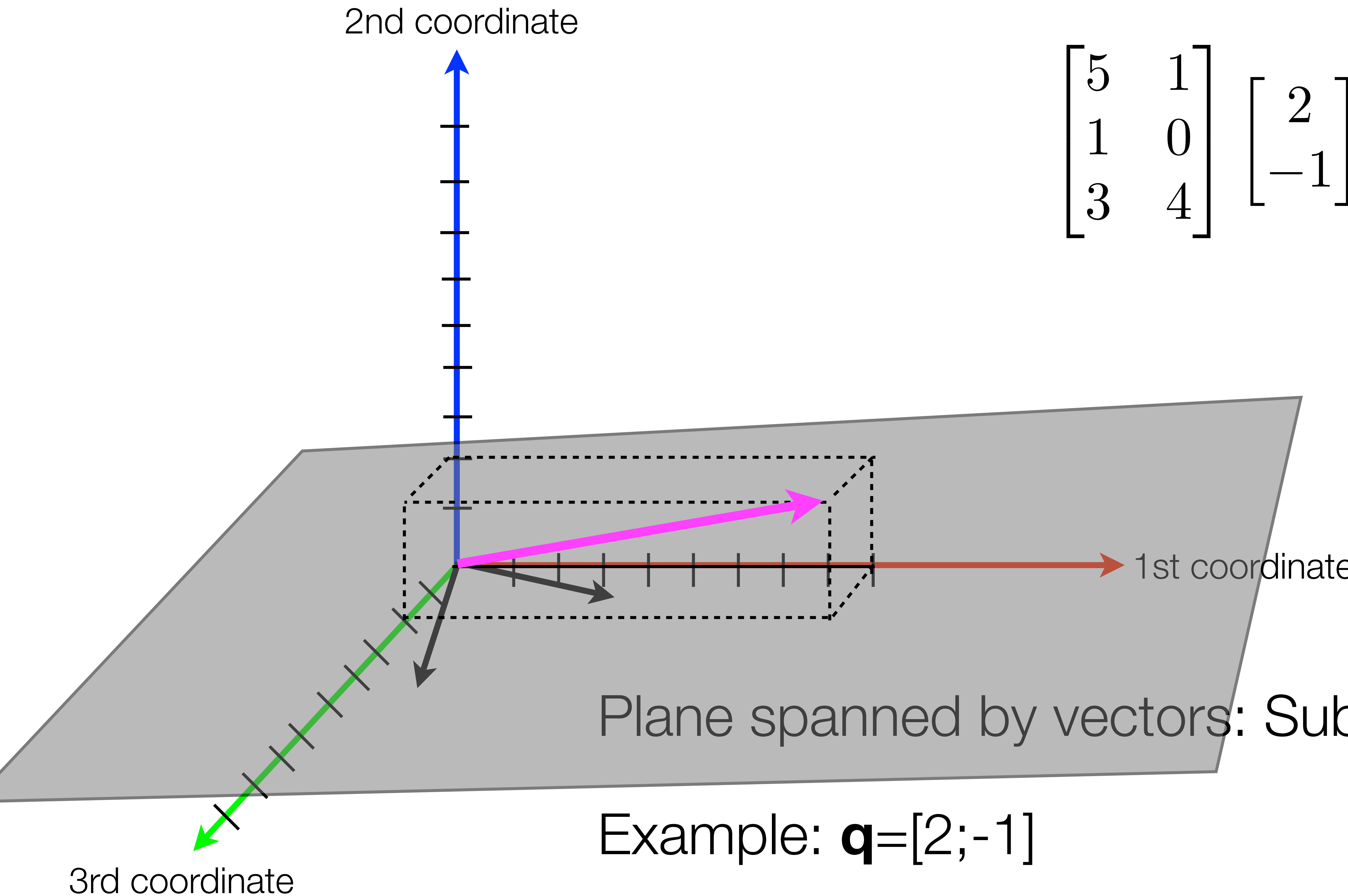
Plane spanned by vectors: Subspace

Example: **q**=[2;-1]

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 2 \\ -1 \end{bmatrix} = \begin{bmatrix} 9 \\ 2 \\ 2 \end{bmatrix}$$

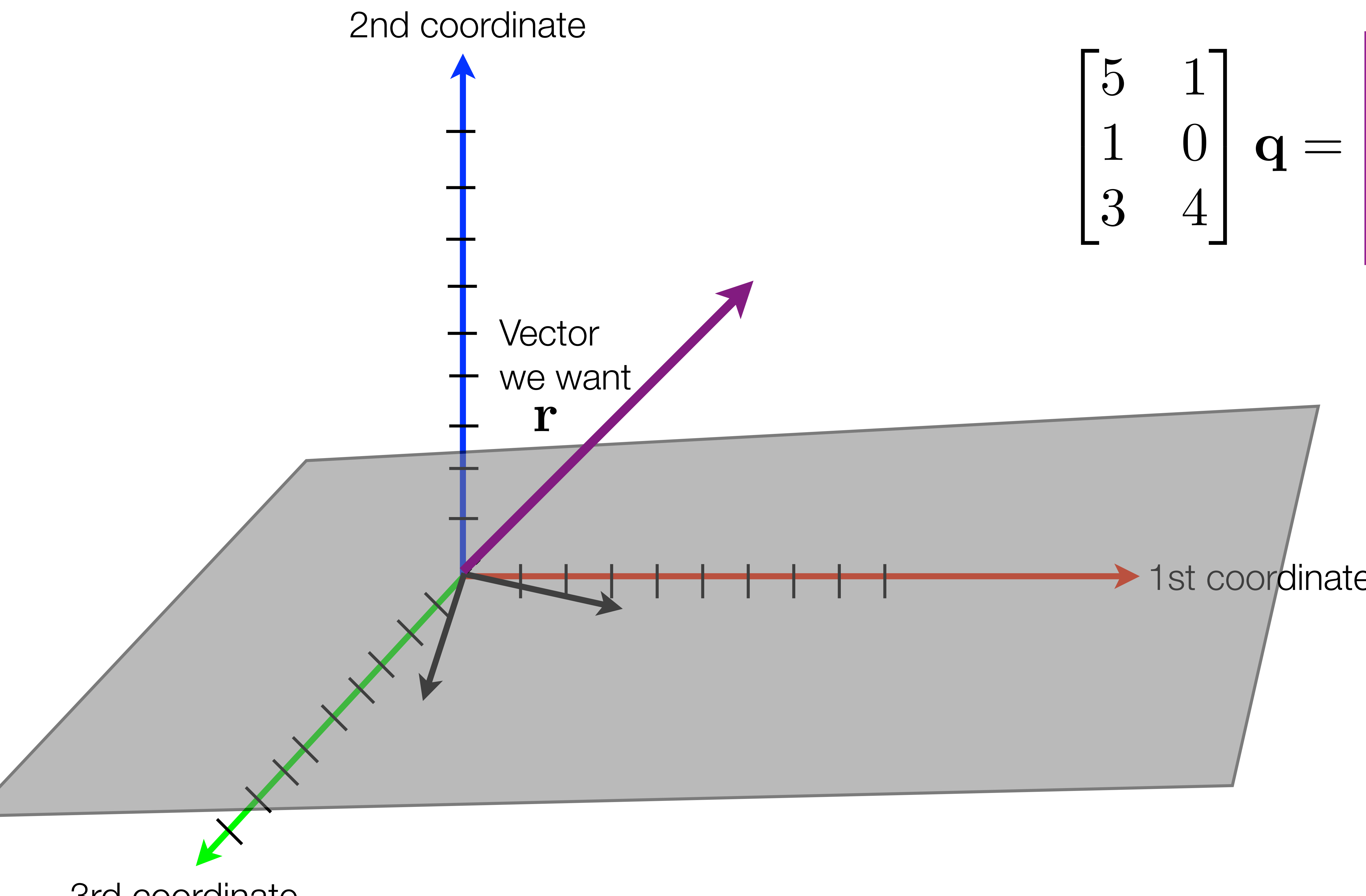


Plane spanned by vectors: Subspace

Example: **q**=[2;-1]

Geometric Representation of Equations

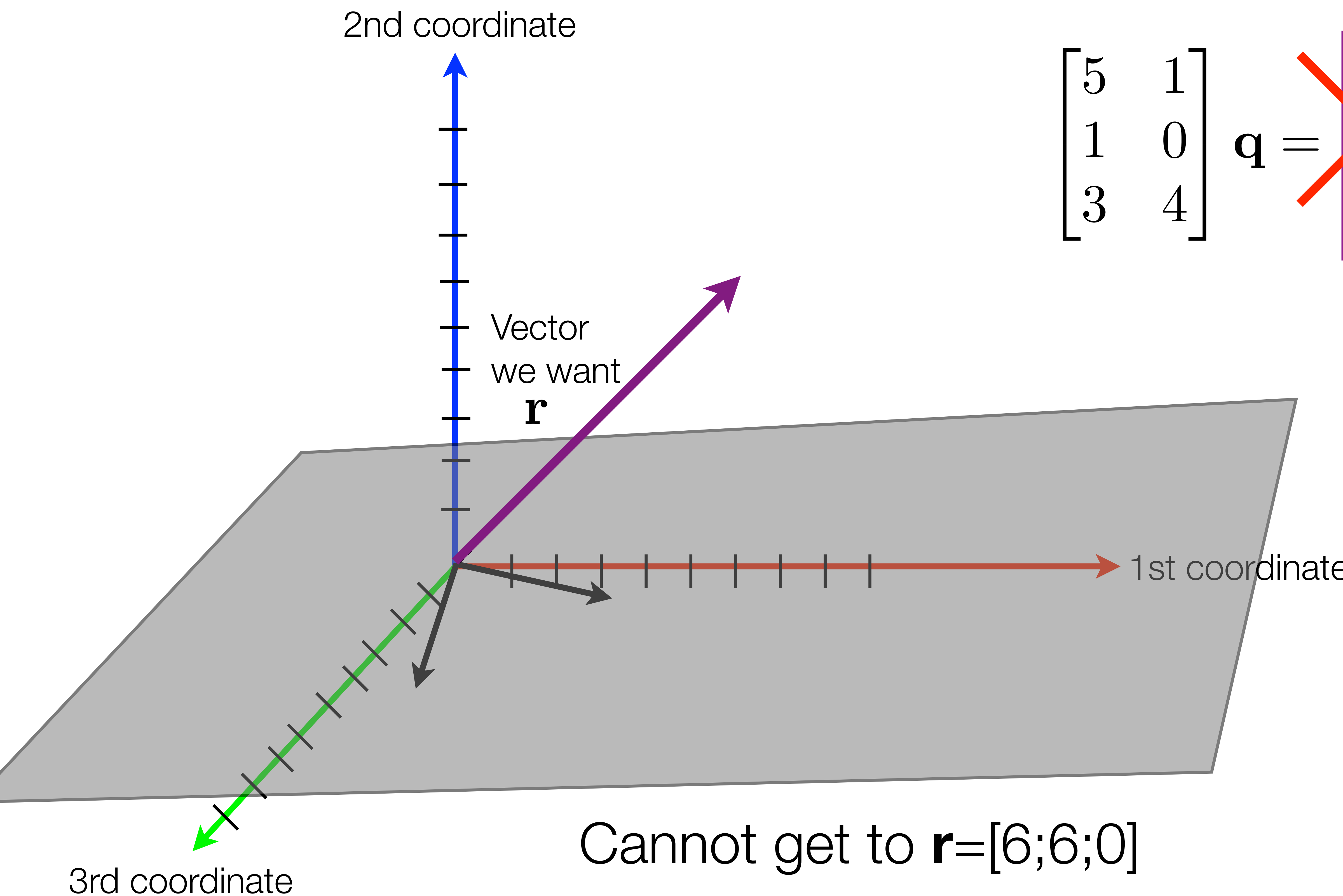
Columns of **P** are a basis for a subspace



$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \boxed{\begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}}$$

Geometric Representation of Equations

Columns of **P** are a basis for a subspace



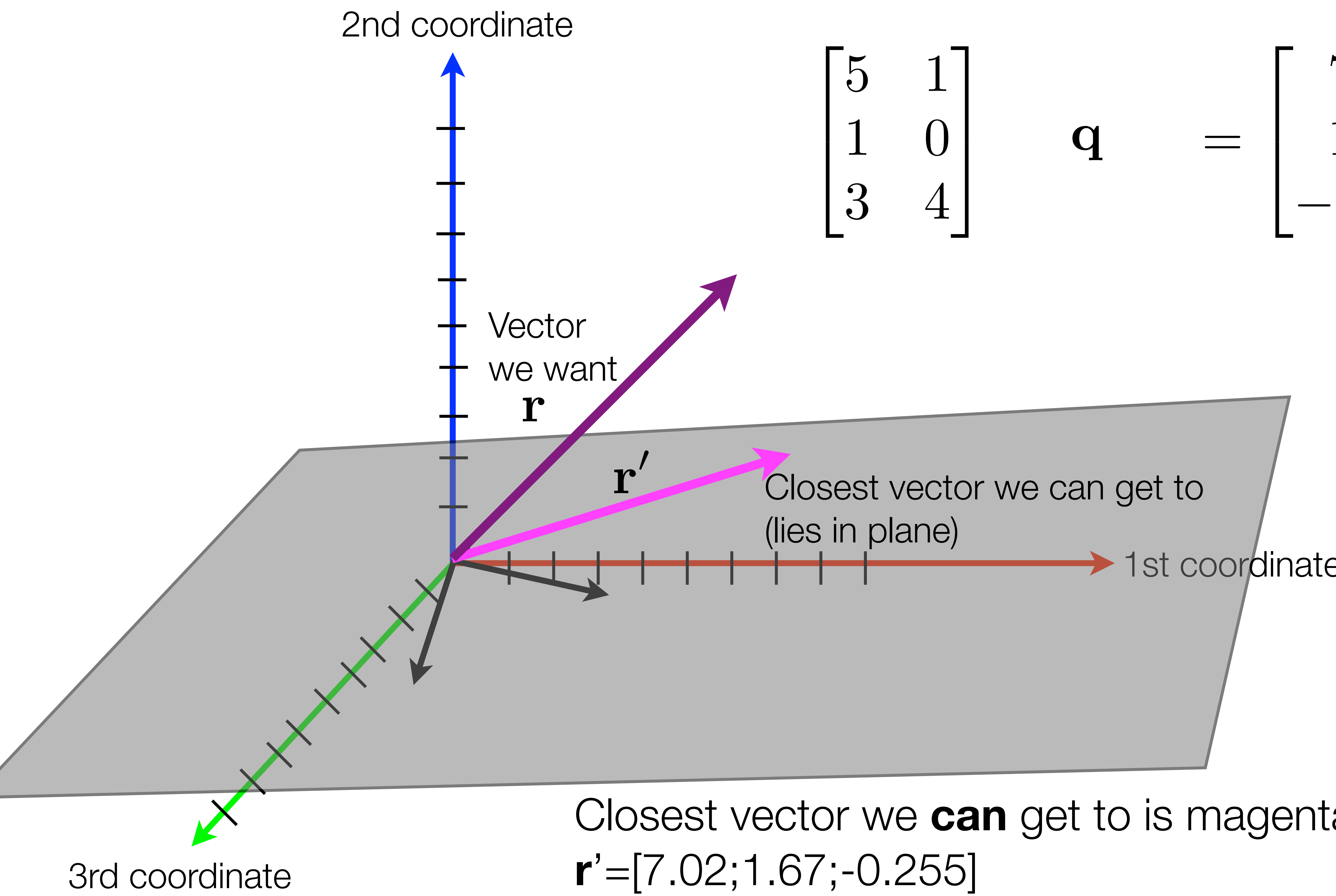
$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \boxed{\cancel{\begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}}}$$

Cannot get to **r**=[6;6;0]

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \quad \mathbf{q} \quad = \begin{bmatrix} 7.02 \\ 1.67 \\ -0.255 \end{bmatrix}$$

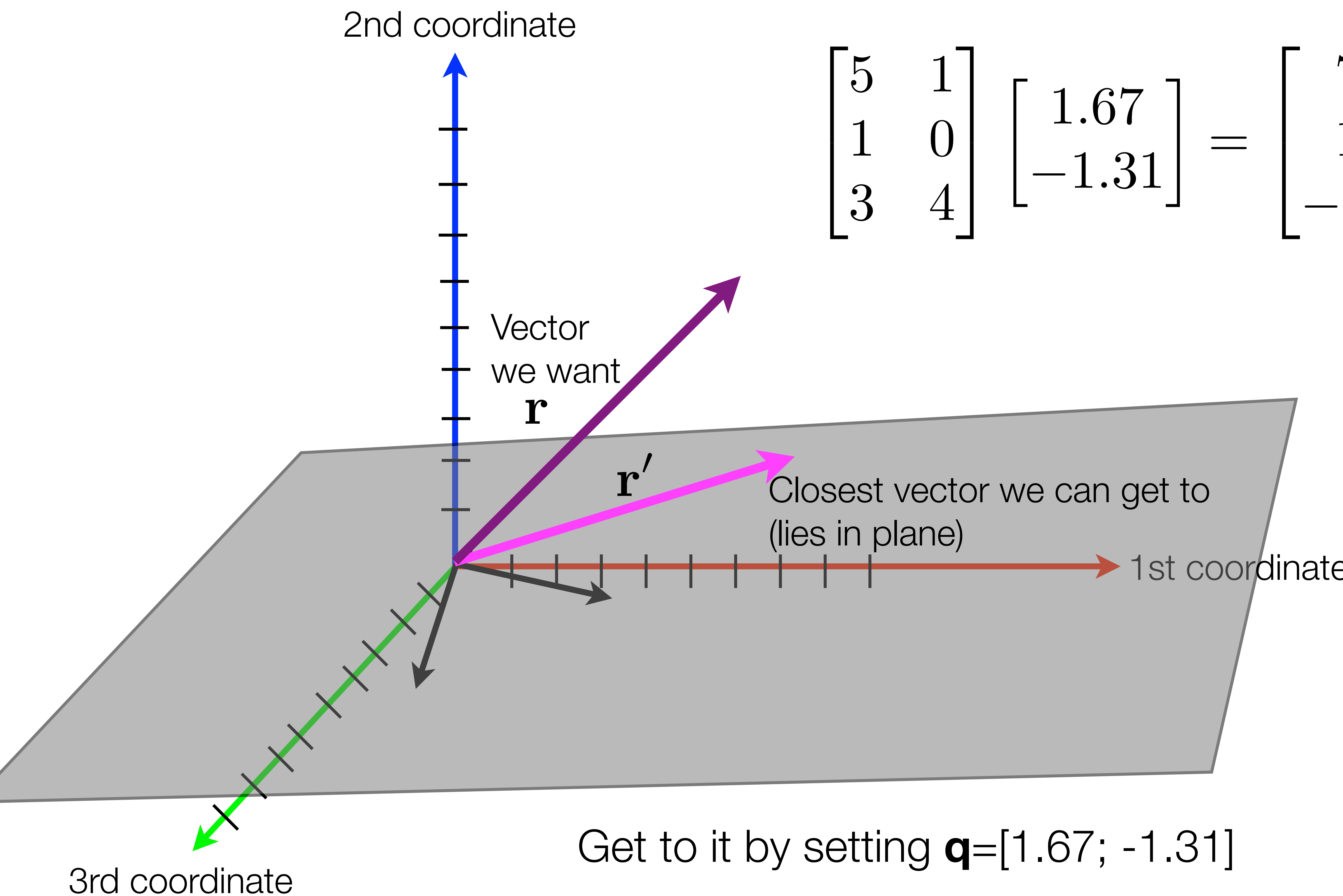


Closest vector we **can** get to is magenta vector
r'=[7.02;1.67;-0.255]

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1.67 \\ -1.31 \end{bmatrix} = \begin{bmatrix} 7.02 \\ 1.67 \\ -0.255 \end{bmatrix}$$



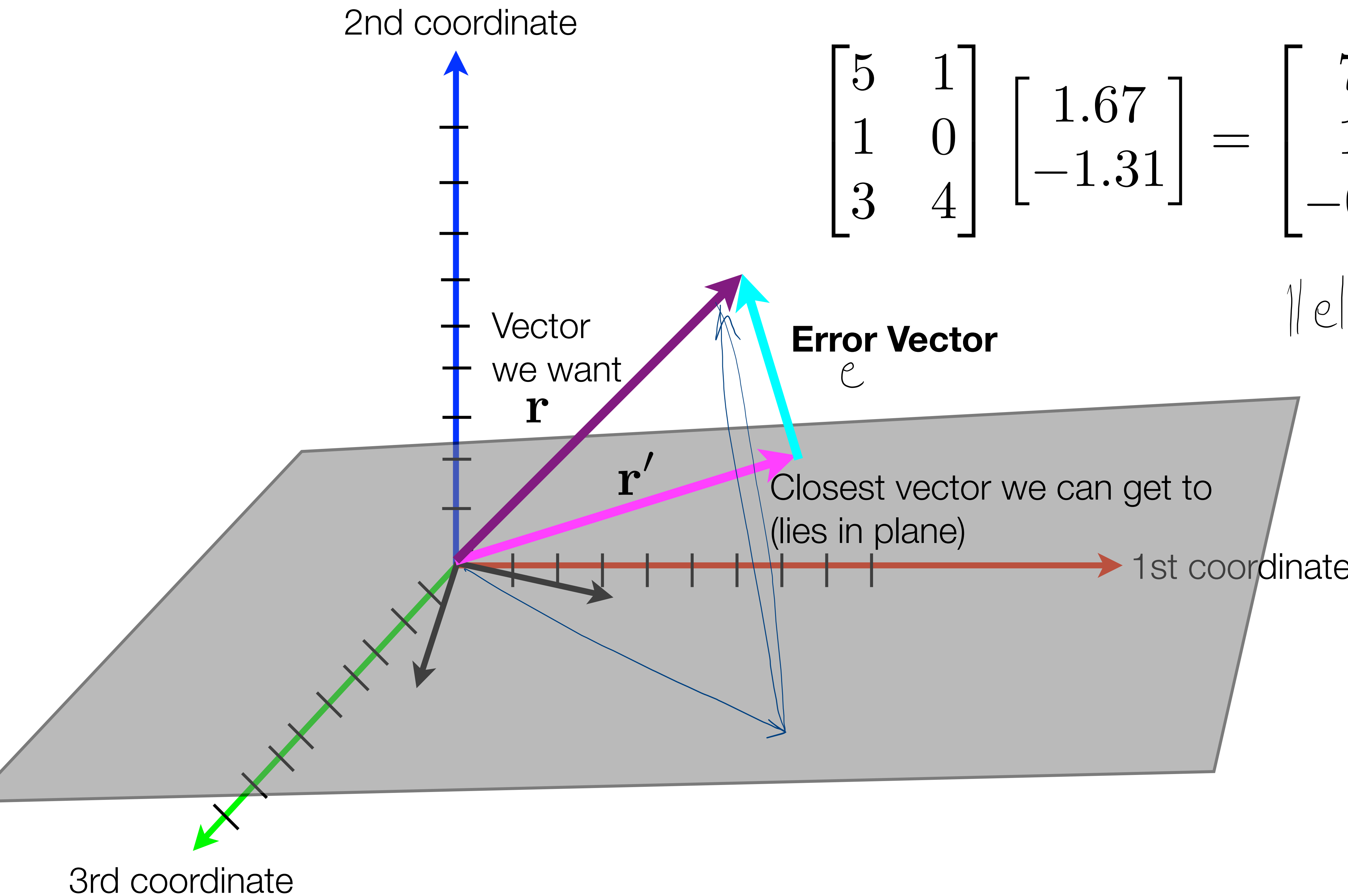
Get to it by setting **q**=[1.67; -1.31]

Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1.67 \\ -1.31 \end{bmatrix} = \begin{bmatrix} 7.02 \\ 1.67 \\ -0.255 \end{bmatrix}$$

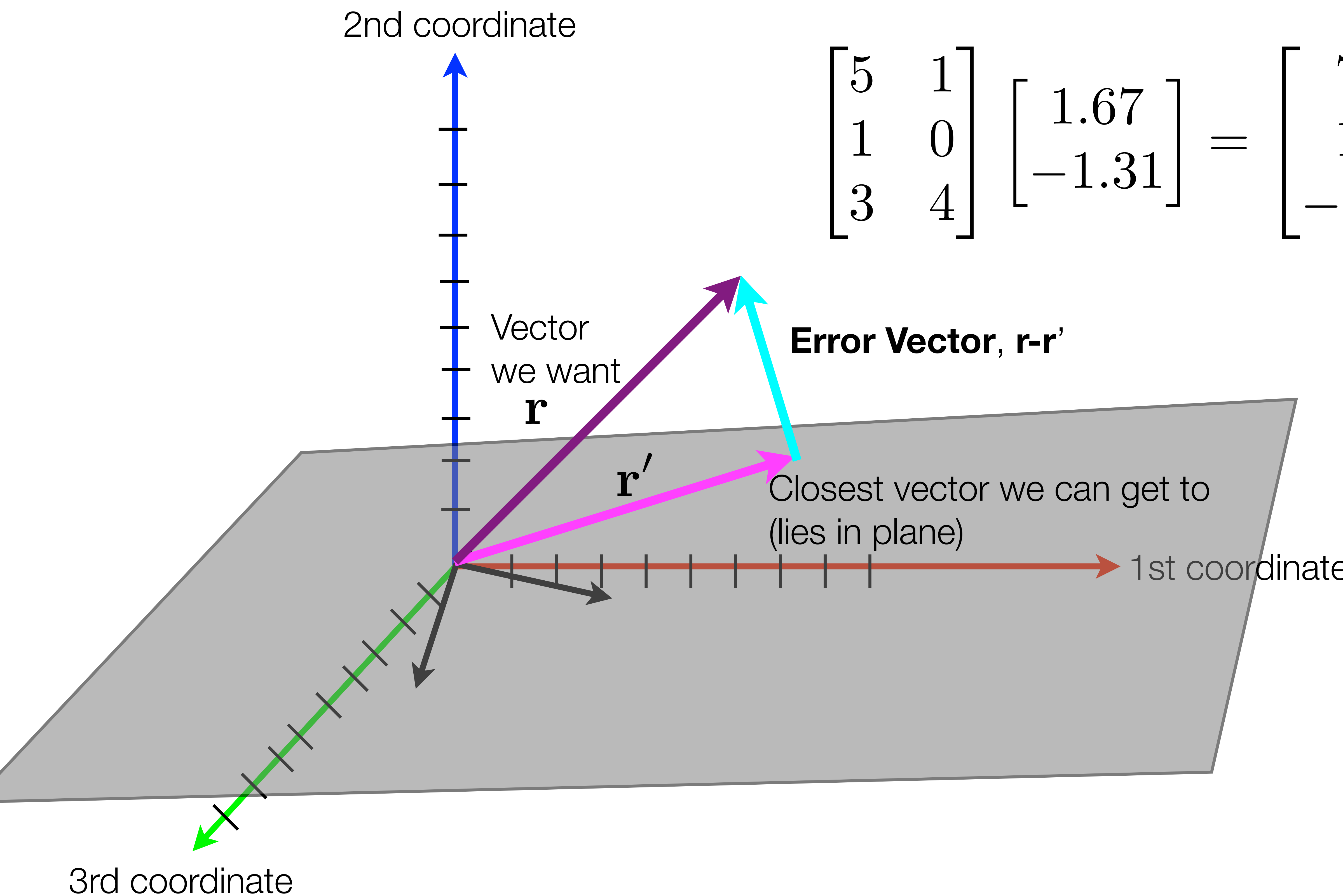
$$\|e\|^2 = \|r - r'\|^2$$



Geometric Representation of Equations

Columns of **P** are a basis for a subspace

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1.67 \\ -1.31 \end{bmatrix} = \begin{bmatrix} 7.02 \\ 1.67 \\ -0.255 \end{bmatrix}$$

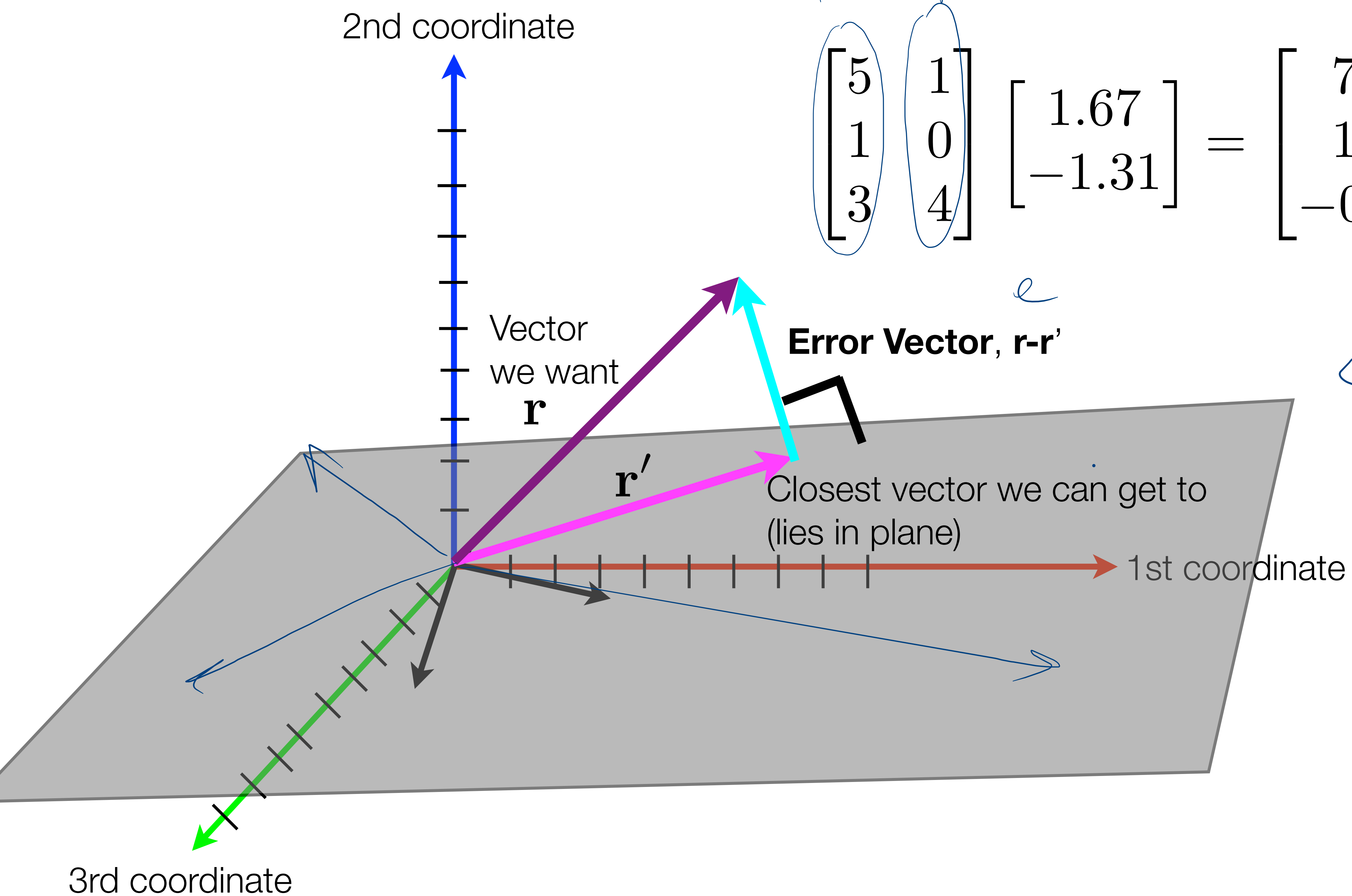


Geometric Representation of Equations

Error Vector is Perpendicular to Subspace (Plane in this Example)

$$\begin{array}{cc} p_1 & p_2 \\ \end{array}$$

$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 1.67 \\ -1.31 \end{bmatrix} = \begin{bmatrix} 7.02 \\ 1.67 \\ -0.255 \end{bmatrix}$$



$\langle e, p_1 \rangle = 0$

$e^T p_1 = 0$

$e^T p_1 = 0$

$e^T p_2 = 0$

$p_1^T e = 0$

$p_2^T e = 0$

$A^T A$

is square and symmetrical.

$$\boxed{\begin{bmatrix} P \end{bmatrix}^T \begin{bmatrix} P \end{bmatrix} = \begin{bmatrix} P^T \end{bmatrix}_3^2 \begin{bmatrix} P \end{bmatrix}_2^3 = \begin{bmatrix} P^T P \end{bmatrix}_2^2}$$

$$\begin{bmatrix} - p_1^T - \end{bmatrix} \begin{bmatrix} | \\ e \\ | \end{bmatrix} = 0$$

$$\begin{bmatrix} - p_2^T - \end{bmatrix} \begin{bmatrix} | \\ e \\ | \end{bmatrix} = 0$$

$$\begin{bmatrix} - p_1^T - \\ - p_2^T - \end{bmatrix} \begin{bmatrix} | \\ e \\ | \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} | & | \\ p_1 & p_2 \\ | & | \end{bmatrix}}_{P}^T \begin{bmatrix} | \\ e \\ | \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$r' = Pq$ exactly

$P^T e = 0$

$P^T (r - r') = 0$

$Pq = r$

$P^T (r - Pq) = 0$

normal equation

$(P^T r) - (P^T P) q = 0$

$(P^T P) q = P^T r$

$$\boxed{\boxed{q} = (P^T P)^{-1} (P^T r) = \boxed{(P^T P)^{-1} P^T}\, r}$$

pseudoinverse

$$q = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix}$$

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

$$\begin{bmatrix} & & \\ & & \\ & & 0 \end{bmatrix}$$

$$\cancel{\begin{array}{ccc} a' & b' & c' \\ \hline \end{array}}$$

$$a' = 2a$$

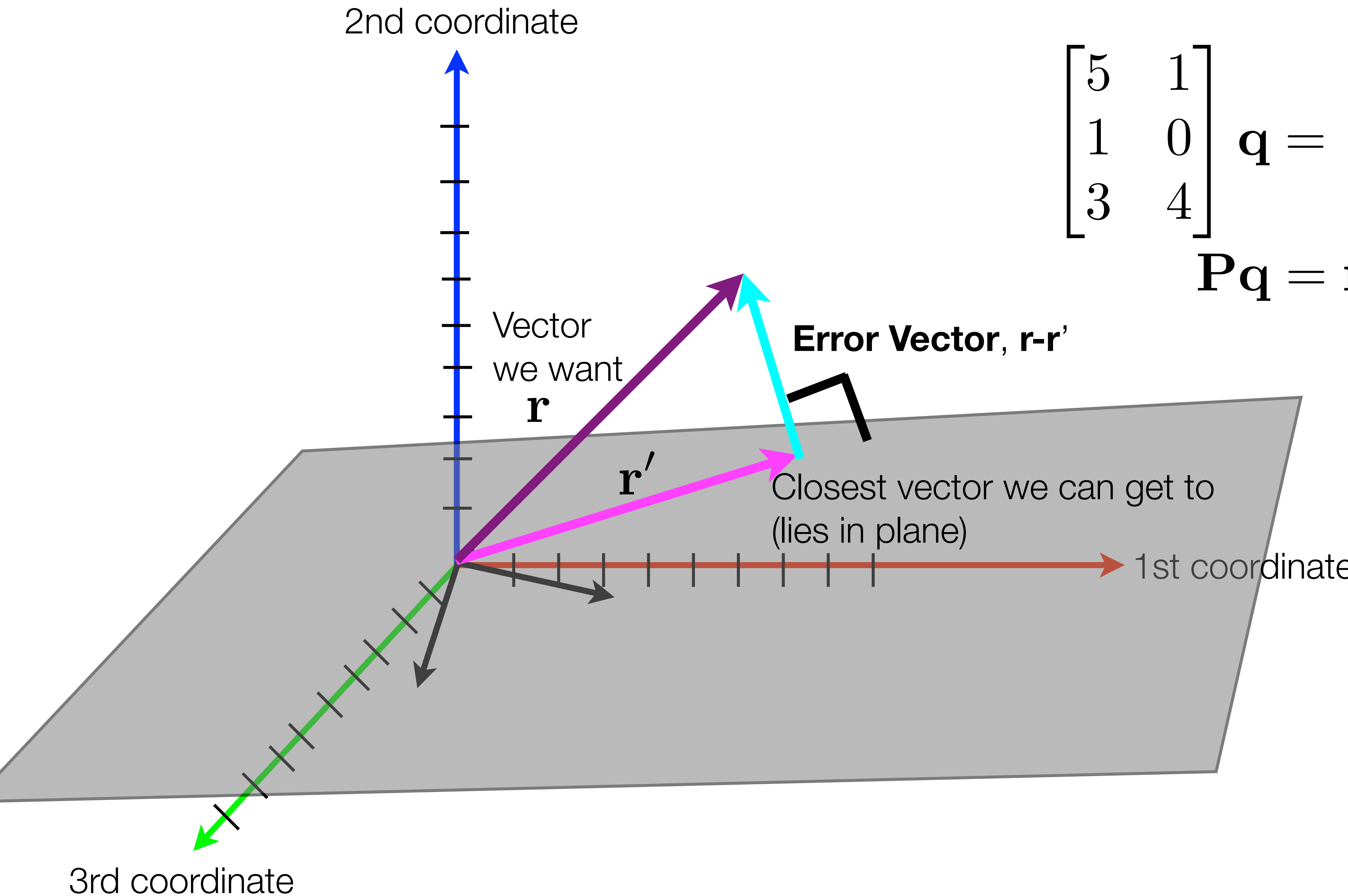
How to Compute **q**?

Use Perpendicularity to Obtain $\mathbf{q}$

Error Vector is Perpendicular to All Vectors in Plane

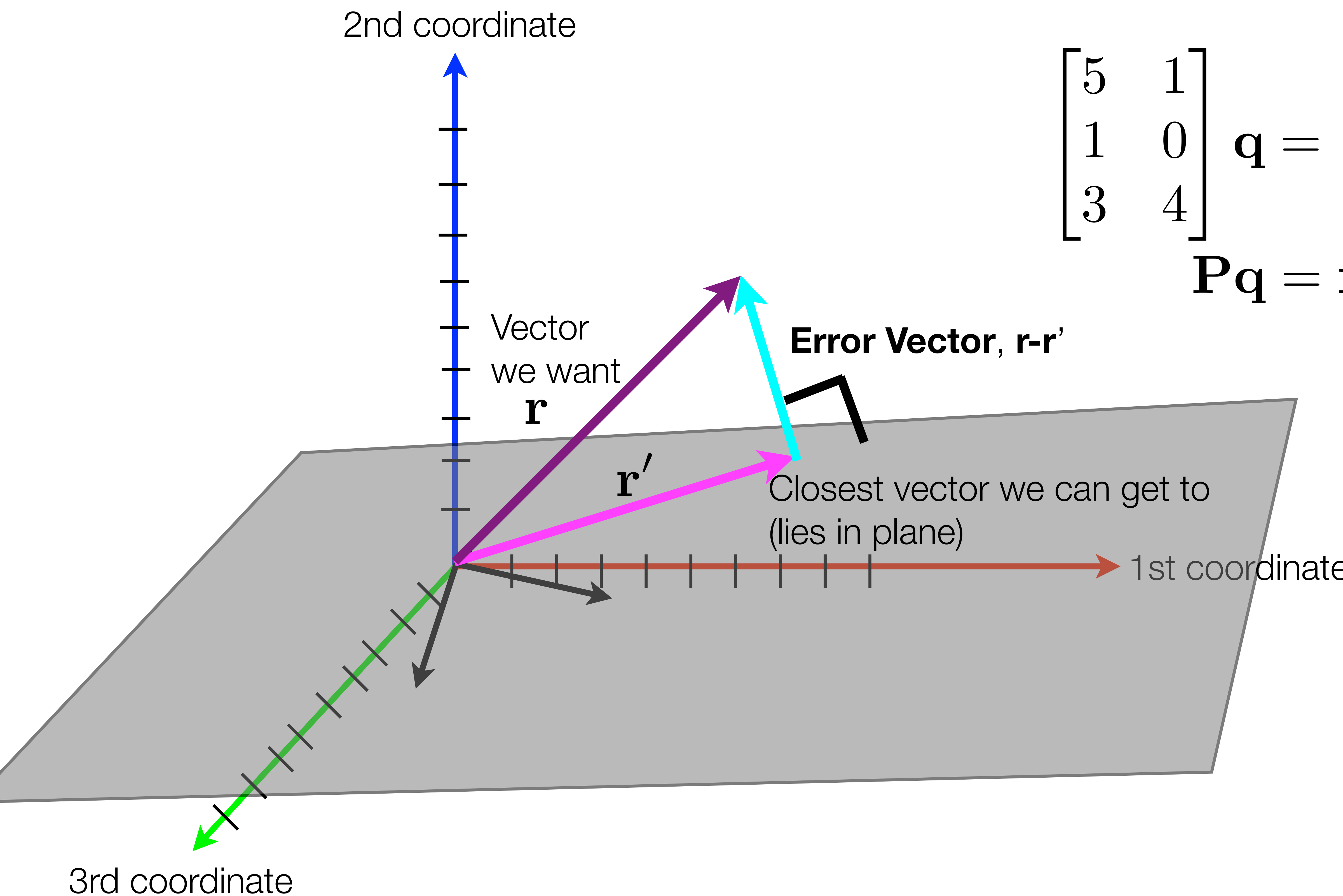
$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

$$\mathbf{P}\mathbf{q} = \mathbf{r}$$



Use Orthogonality to Obtain **q**

Error Vector is Orthogonal to All Vectors in Plane



$$\begin{bmatrix} 5 & 1 \\ 1 & 0 \\ 3 & 4 \end{bmatrix} \mathbf{q} = \begin{bmatrix} 6 \\ 6 \\ 0 \end{bmatrix}$$

$$\mathbf{Pq} = \mathbf{r}$$

Using Pseudoinverse of **P** to Obtain **q**

We can solve the system of equations

$$\mathbf{P}\mathbf{q} = \mathbf{r}$$

as follows:

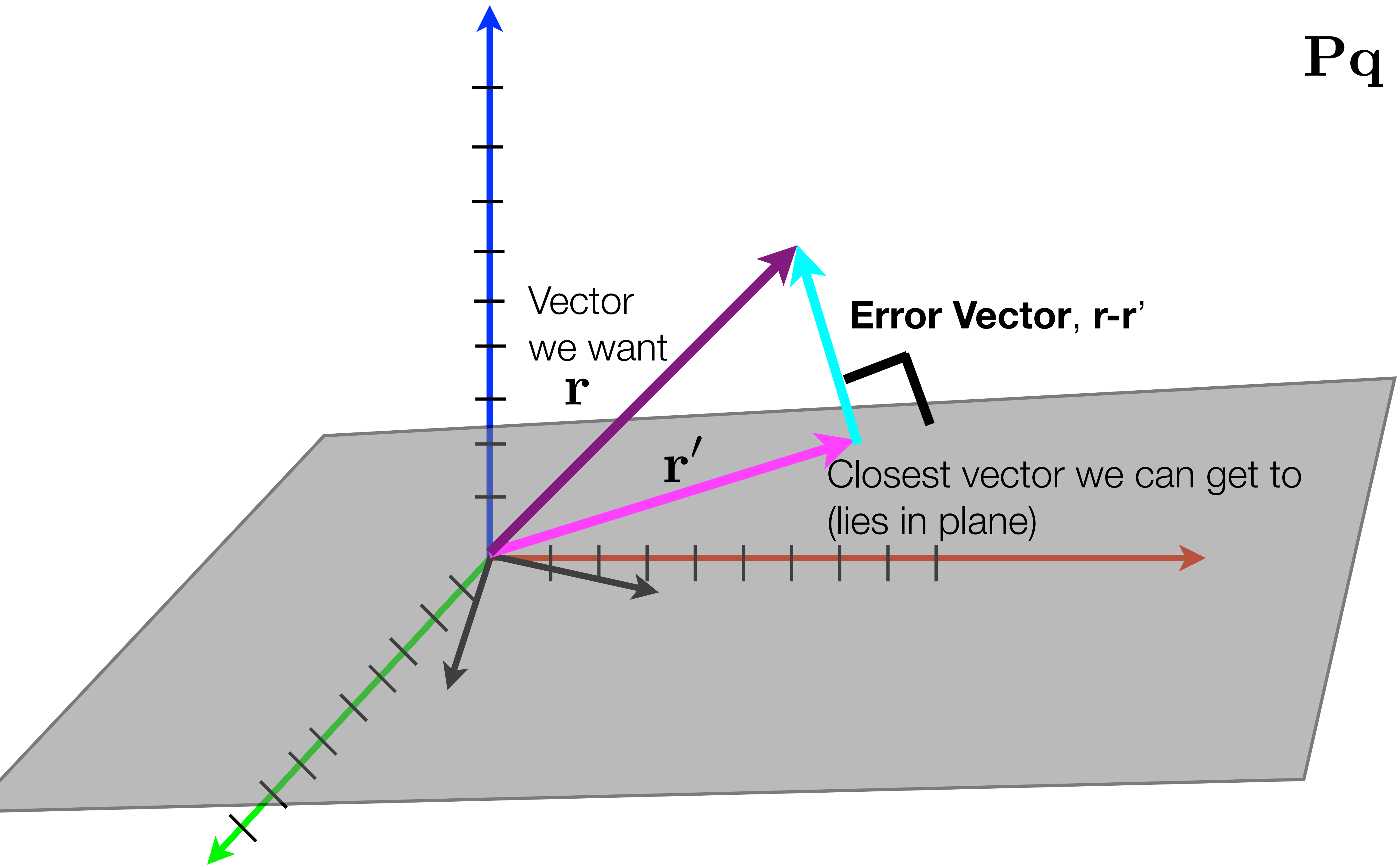
$\mathbf{q}$ = (Pseudoinverse of $\mathbf{P}$) $\mathbf{r}$.

Pseudoinverse of $\mathbf{P}$ = $(\mathbf{P}^T\mathbf{P})^{-1}$ $(\mathbf{P}^T)$

Same Principle Applies to Larger System

Error Vector is Orthogonal to All Vectors in Plane

$$\mathbf{Pq} = \mathbf{r}$$



Issue with pseudoinverse

$P = \begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & \boxed{x'x} & x'y \end{bmatrix}$

$1000 \times 1000 \approx 10^6$

1080 1920

Need to compute inverse of $\mathbf{P}^T\mathbf{P}$.
The computation is usually numerically unstable.

A better method is to use
homogeneous least squares
with a technique called
singular value decomposition (SVD).

2) Homogeneous Least Squares with SVD

We will look at two methods

2) Use **homogeneous least squares** with the following system:

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, x' = \frac{\hat{x}'}{\hat{w}'}, y' = \frac{\hat{y}'}{\hat{w}'}$$

General form of homography matrix

$\hat{x}' = ax + by + c$

$\hat{y}' = dx + ey + f$

$\hat{w}' = gx + hy + i$

$$x' = \frac{ax + by + c}{gx + hy + i}$$

$$y' = \frac{dx + ey + f}{gx + hy + i}$$

DLT

$x'(gx + hy + i) = ax + by + c$

$x'gx + x'hy + x'i = ax + by + c$

$-ax - by - c + x'gx + x'hy + x'i = 0$

$(-x)a + (-y)b + (-1)c + 0 \cdot d + 0 \cdot e + 0 \cdot f + (x'x)g + (x'y)h + (x')i = 0$

$Pq = 0$

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{array}{c} \\ 1 \\ \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \overbrace{\begin{bmatrix} -x & -y & -1 & 0 & 0 & 0 & x'x & x'y & x' \\ 0 & 0 & 0 & -x & -y & -1 & y'x & y'y & y' \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \\ & & & & & & & & \end{bmatrix}}^{P} \overbrace{\begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \end{bmatrix}}^{q} = \overbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}^{0}$$

homogeneous equs.

$\cancel{x' = \frac{0}{0}} \quad \cancel{y' = \frac{0}{0}}$

Written as matrix times vector

Convert

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, x' = \frac{\hat{x}'}{\hat{w}'}, y' = \frac{\hat{y}'}{\hat{w}'}$$

to the form

$$\begin{bmatrix} \qquad\qquad\qquad\qquad \\ \\ \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \end{bmatrix} = \begin{bmatrix} \ \\ \\ \end{bmatrix}$$

Design Matrix

Written as matrix times vector

Convert

$$\begin{bmatrix} \hat{x}' \\ \hat{y}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, x' = \frac{\hat{x}'}{\hat{w}'}, y' = \frac{\hat{y}'}{\hat{w}'}$$

to the form

$$\begin{bmatrix} \quad\quad\quad\quad\quad\quad \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \end{bmatrix} = \begin{bmatrix} \ \end{bmatrix}$$

Design Matrix

Notice that the output vector is all zeros!

Written as matrix times vector

Use many correspondences

$$(x_1, y_1) \rightarrow (x'_1, y'_1), (x_2, y_2) \rightarrow (x'_2, y'_2), \ldots$$

to set up

$$\begin{bmatrix} & & & & & & & & \\ \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \end{bmatrix} = \begin{bmatrix} \\ \end{bmatrix}$$

Notice that the output vector is all zeros!

Written as matrix times vector

How many correspondences do we need?

Use many correspondences

$$(x_1, y_1) \rightarrow (x'_1, y'_1), (x_2, y_2) \rightarrow (x'_2, y'_2), ...$$

to set up

$$\begin{bmatrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{bmatrix}$$

Written as matrix times vector

Use many correspondences

$$(x_1, y_1) \rightarrow (x_1', y_1'), (x_2, y_2) \rightarrow (x_2', y_2'), \ldots$$

to set up

$$\begin{bmatrix} \quad & \quad & \quad \\ \\ \\ \\ \\ \\ \\ \\ \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{bmatrix}$$

How many correspondences do we need?

At least 9 equations needed.

But we can only have even number of equations.

~~So we need~~
~~10 equations~~ or
5 correspondences.

Homogeneous least squares

We have the following system:

$$\mathbf{P}\,\mathbf{q} = \mathbf{0}$$

where **P** is a tall matrix,
q is a column vector of elements we want,
and **0** is a column vector full of zeros.

Regular least squares (with the pseudoinverse of **P**)
gives us $\mathbf{q} = (\mathbf{P}^T\mathbf{P})^{-1}(\mathbf{P}^T\mathbf{0}) = \mathbf{0}$.

But **q** should be non-zero!

Solution for homogeneous least squares

Solving $\mathbf{P}\,\mathbf{q} = \mathbf{0}$:

You get $\mathbf{q}$ as follows:

1) Take the singular value decomposition (SVD) of $\mathbf{P}$.

Solution for homogeneous least squares

Solving $\mathbf{P}\,\mathbf{q} = \mathbf{0}$:

You get $\mathbf{q}$ as follows:

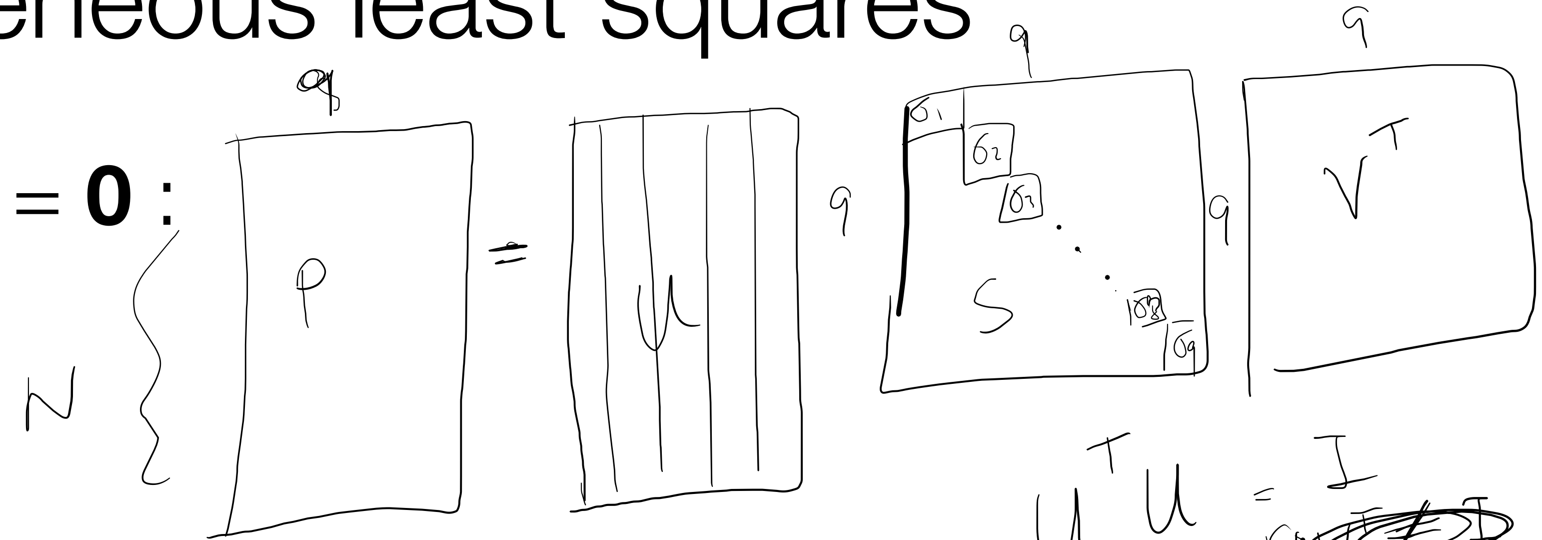
1) Take the singular value decomposition (SVD) of $\mathbf{P}$.

The SVD of $\mathbf{P}$ gives three matrices $\mathbf{U}, \mathbf{S}, \mathbf{V}$ such that

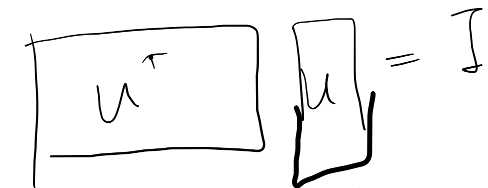
$$\mathbf{P} = \mathbf{U}\,\mathbf{S}\,\mathbf{V}^T$$

Solution for homogeneous least squares

Solving **P q** = **0** :



$U^T U = I$



You get **q** as follows:

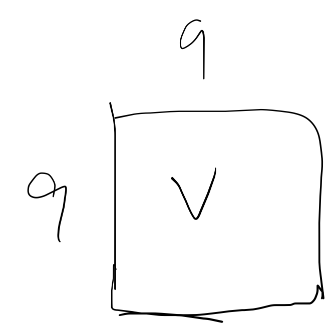
1) Take the singular value decomposition (SVD) of **P**.

The SVD of **P** gives three matrices **U, S, V** such that

$$\mathbf{P} = \mathbf{U}\,\mathbf{S}\,\mathbf{V}^T$$

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

where **U** and **V** are orthogonal matrices, while **S** is a diagonal matrix of **singular values**.

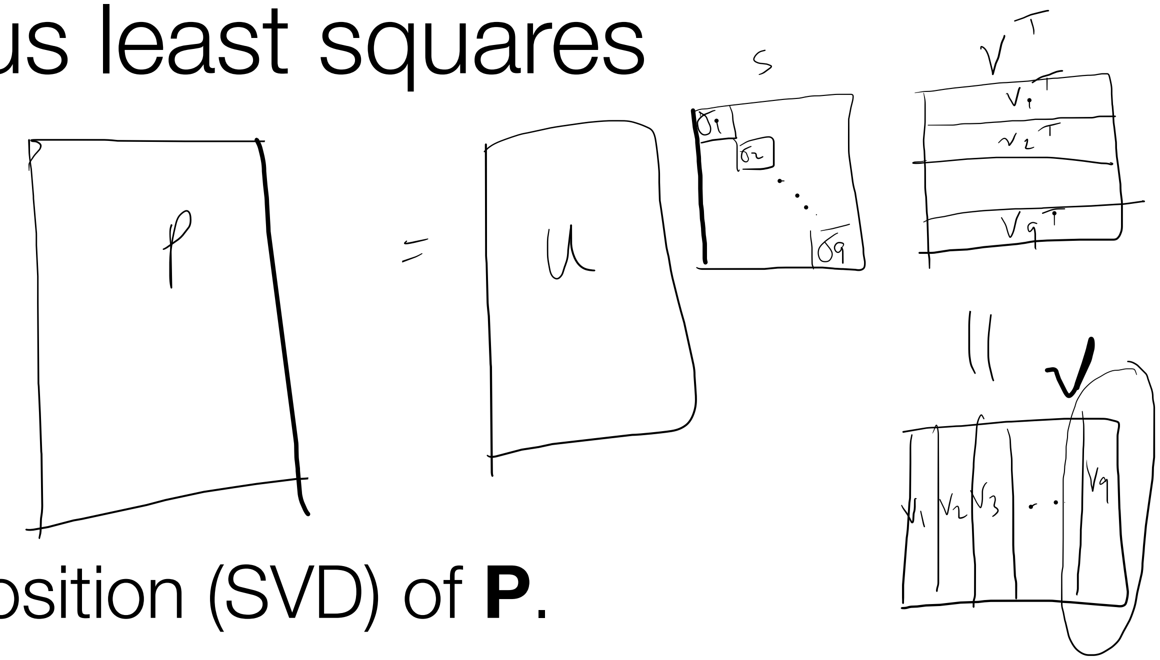


Solution for homogeneous least squares

$[U, S, V] = svd(P, 'econ')$

$q = V(:, end)$

Solving **P q** = **0** :



You get **q** as follows:

1) Take the singular value decomposition (SVD) of **P**.

The SVD of **P** gives three matrices **U, S, V** such that

$$q = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \end{bmatrix} \rightarrow \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

$$\mathbf{P} = \mathbf{U}\,\mathbf{S}\,\mathbf{V}^T$$

where **U** and **V** are orthogonal matrices, while **S** is a diagonal matrix of **singular values**.

2) Get **q** as the column vector in **V** corresponding to the location of the **smallest singular value** in **S**.

Singular Value Decomposition (SVD)

We will look at the effect of SVD for a 2 x 2 matrix **A**.

P is a matrix of size m x n
where m & n >= 10, and definitely m & n > 2.
8

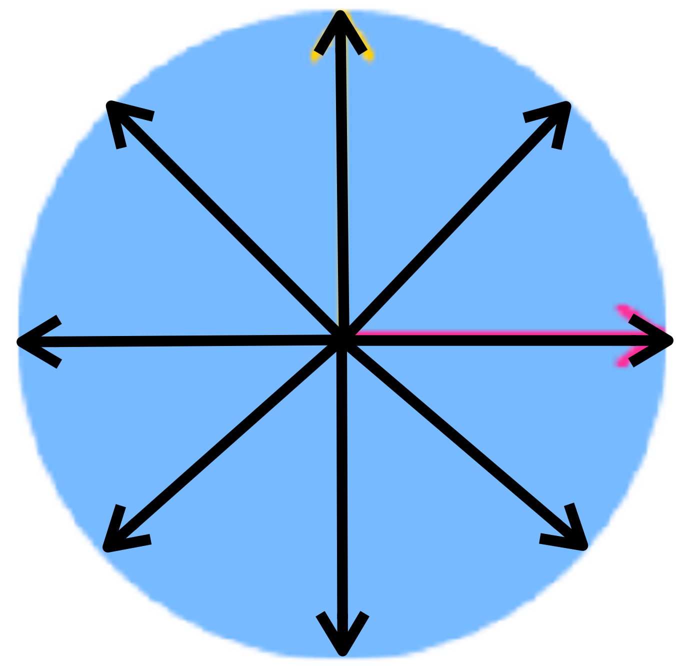
But it is hard to visualize so many dimensions,
so we will get intuitions for 2 x 2.
These intuitions extend to any number of dimensions.

Singular Value Decomposition (SVD)

We will look at the effect of SVD for a 2 x 2 matrix **A**.

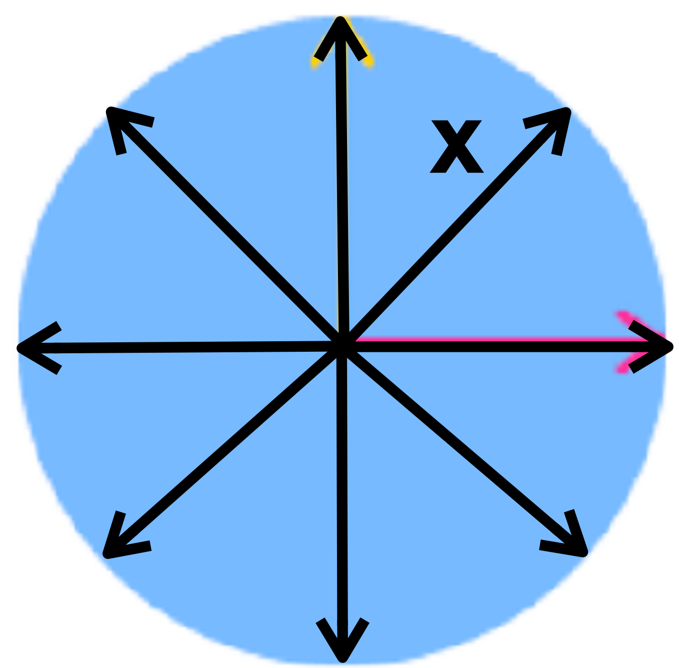
Singular Value Decomposition (SVD)

Let us see the change **A** induces on a set of vectors in the unit circle.



Singular Value Decomposition (SVD)

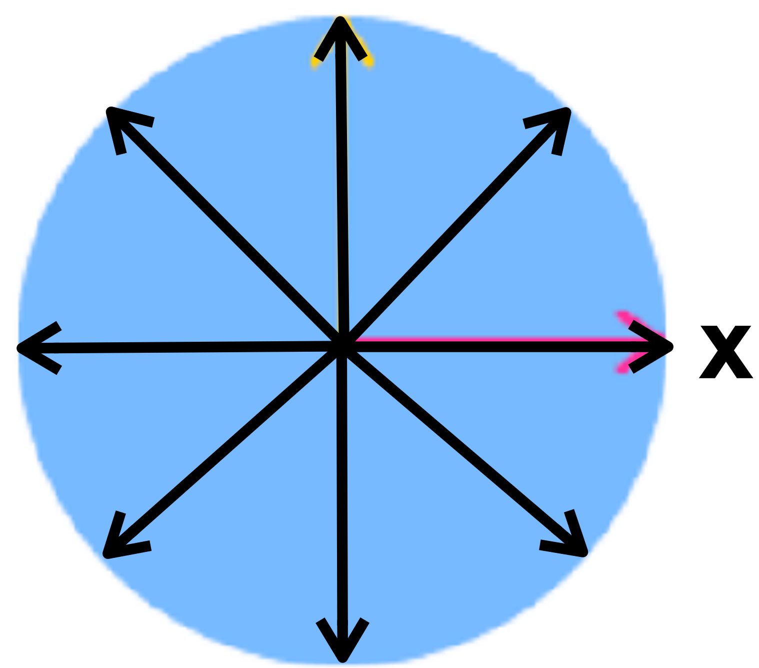
Let us see the change **A** induces on
a set of vectors in the unit circle.



How can we study this change for a vector **x**?

Singular Value Decomposition (SVD)

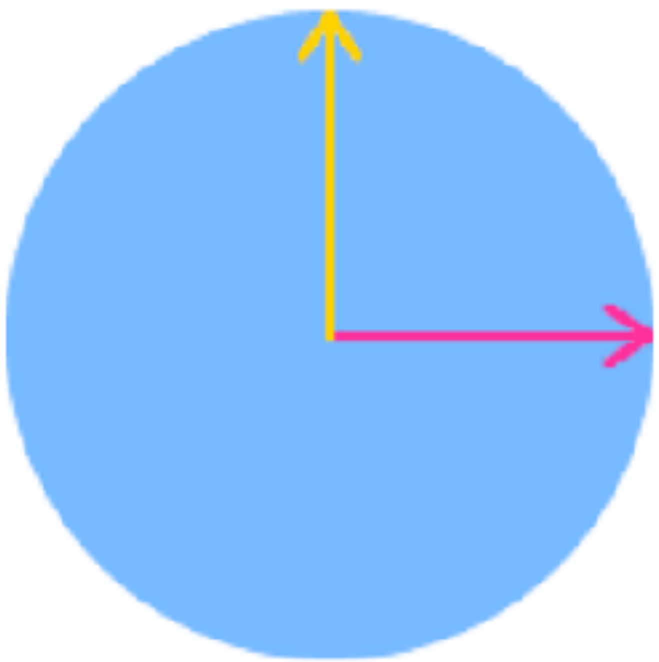
Let us see the change **A** induces on a set of vectors in the unit circle.



How can we study this change for a vector **x**?

Look at vector $\mathbf{y} = \mathbf{A}\mathbf{x}$

Singular Value Decomposition (SVD)

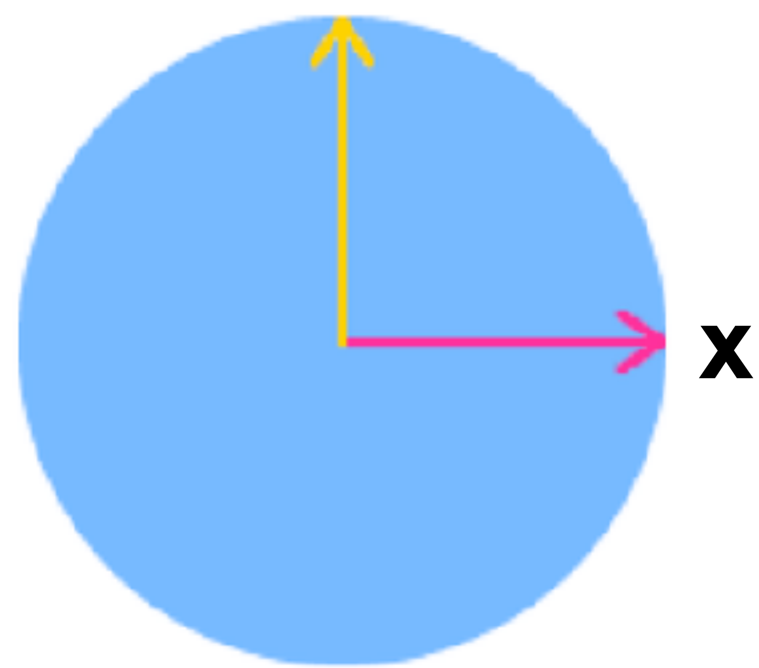


A **x** **U** **S** **V**^{T} **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad\qquad\qquad\qquad \mathbf{x}$$

Singular Value Decomposition (SVD)

$$\mathbf{V} = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}$$

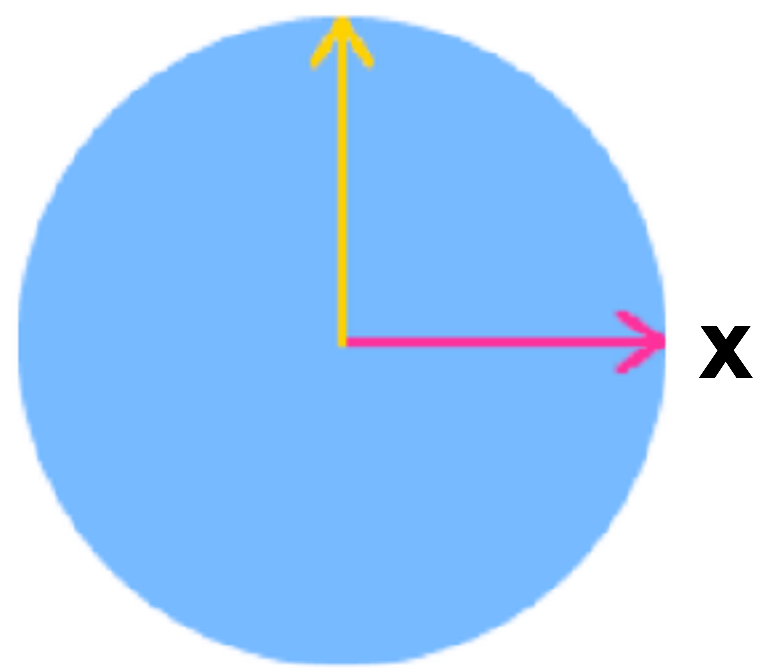


A **x** **U** **S** **V**^{T} **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

$$\mathbf{V} = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}$$



V is an **orthogonal matrix**, i.e., it represents a **rotation**.

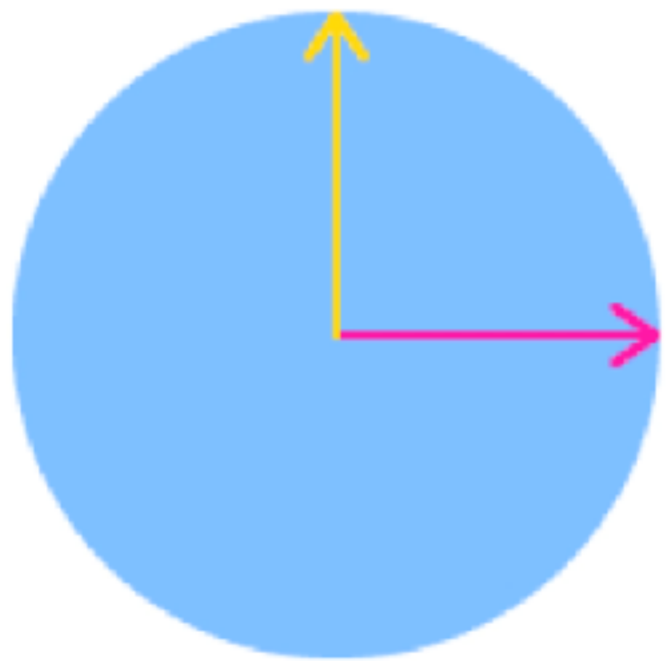
VV*ᵀ* = V*ᵀ*V = I

A **x** **U** **S** **V*ᵀ*** **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

$$\mathbf{V} = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}$$



V is an **orthogonal matrix**, i.e., it represents a **rotation**.

VV^T = V^{T}V = I

A **x** **U** **S** **V^T** **x**

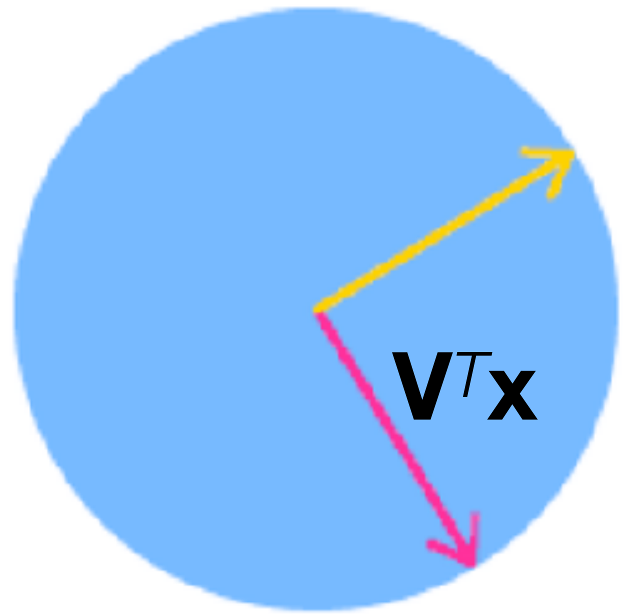
$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

$$V = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

$$V^T = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

$$\mathbf{V} = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix}$$



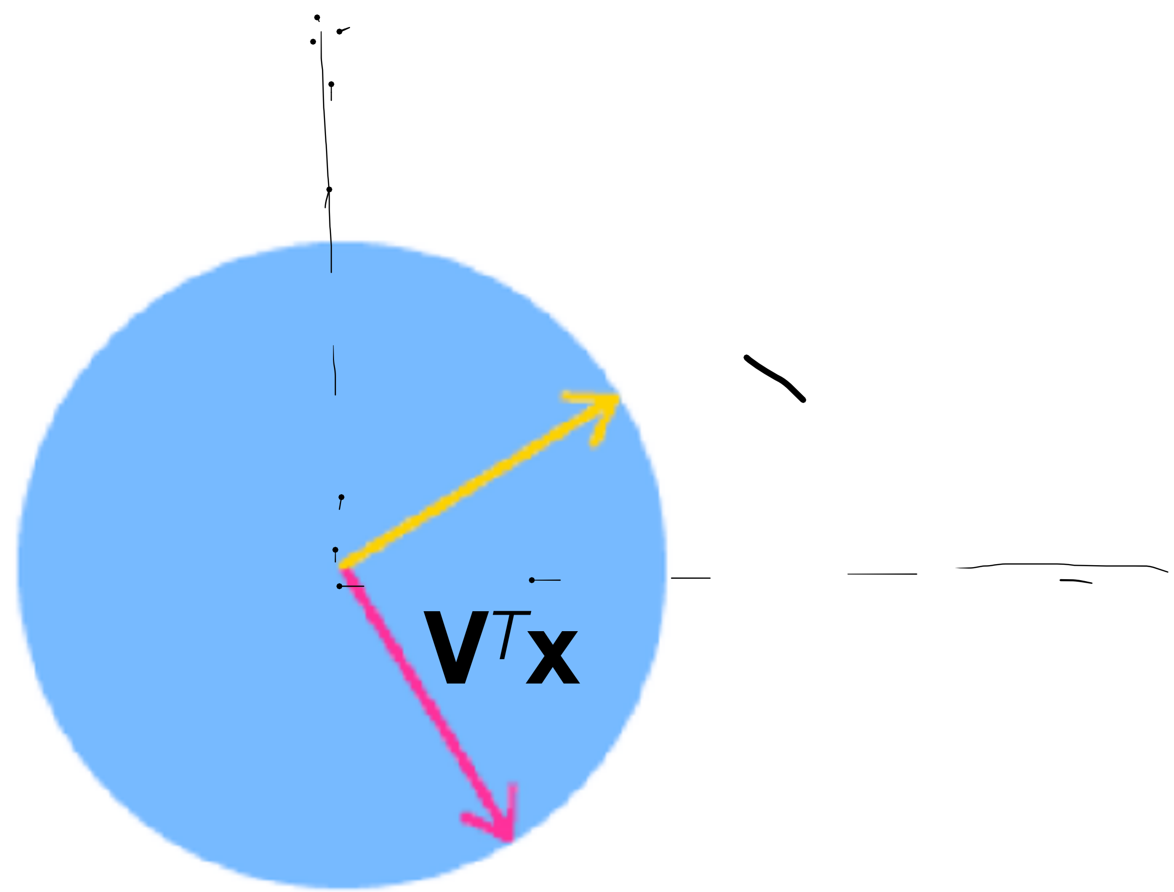
V is an **orthogonal matrix**, i.e., it represents a **rotation**.

VV^T **= V**^{T}**V = I**

A **x** **U** **S** **V**^{T} **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

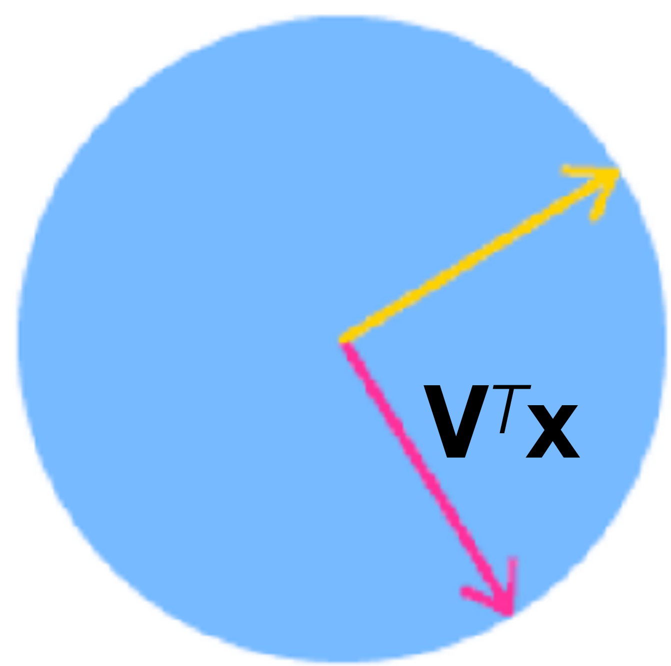


A **x** **U** **S** $\mathbf{V}^T$ **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

S is an **diagonal matrix**, i.e., it represents a **scaling**.

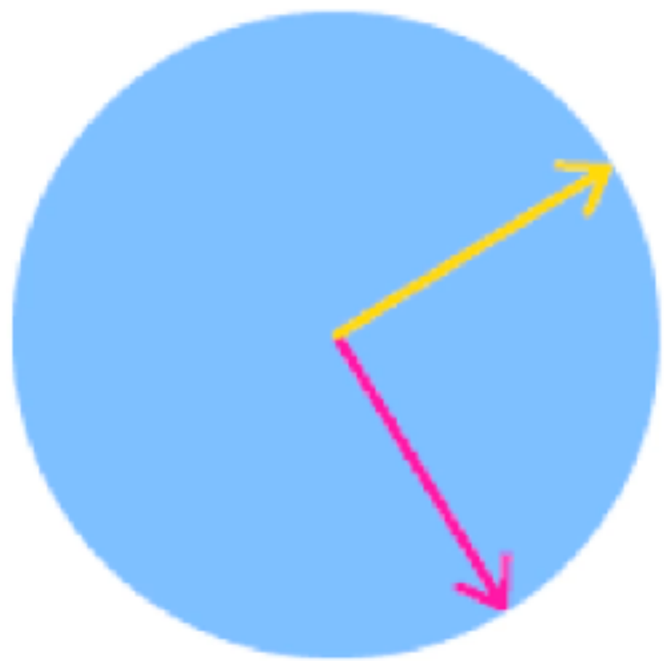


A **x** **U** **S** **V**^{T} **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

S is an **diagonal matrix**, i.e., it represents a **scaling**.

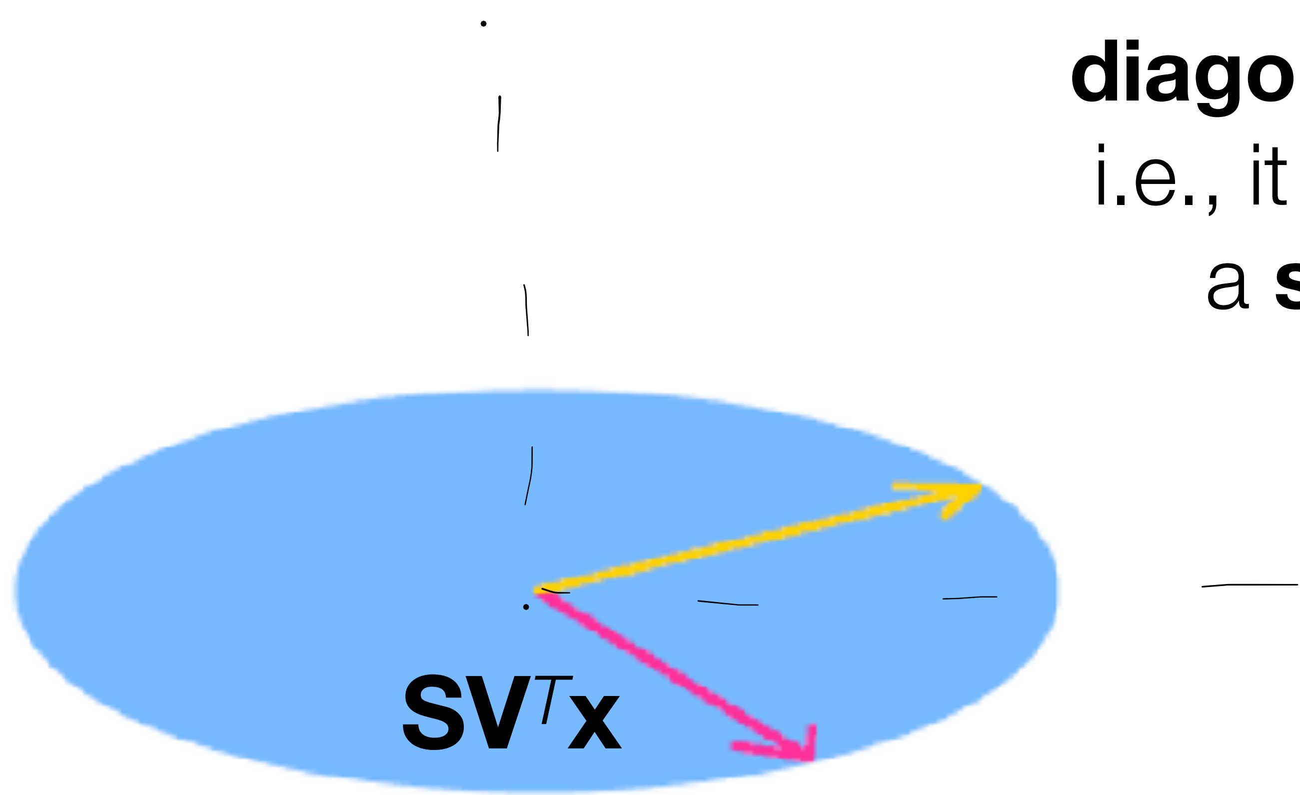


A **x** **U** **S** **V**^{T} **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

S is an **diagonal matrix**, i.e., it represents a **scaling**.

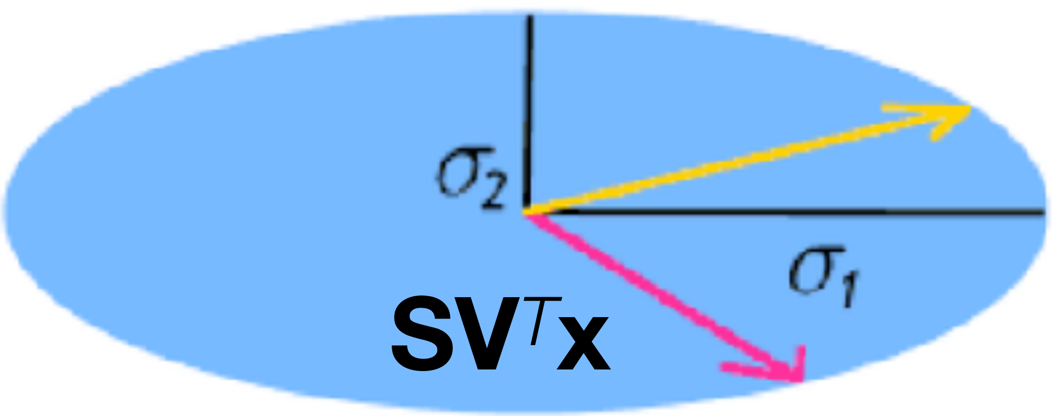


A **x** **U** **S** **V**^{T} **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

S is an **diagonal matrix**, i.e., it represents a **scaling**.



σ_1 and σ_2 are called **singular values**.

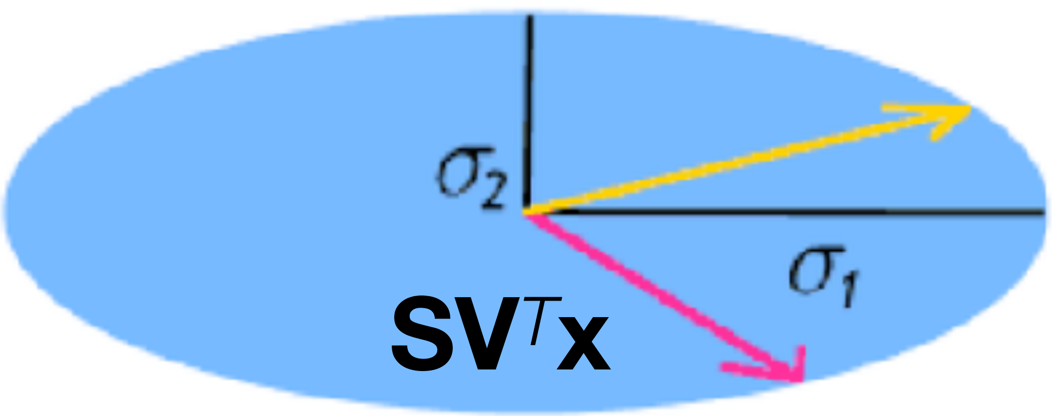
A **x** **U** **S** **V**^{T} **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \qquad \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

U is another **orthogonal matrix**, (different from **V**) i.e., it represents a **second rotation**.

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



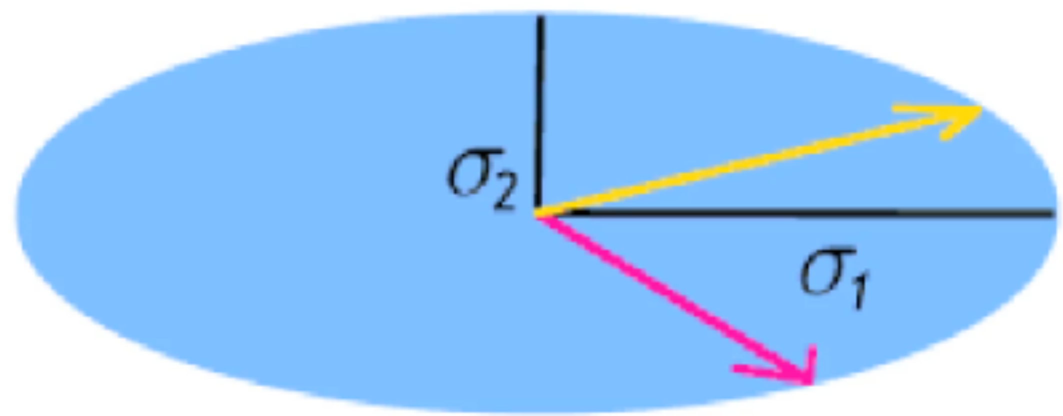
$$\begin{array}{cccccc} \mathbf{A} & \mathbf{x} & \mathbf{U} & \mathbf{S} & \mathbf{V}^T & \mathbf{x} \end{array}$$

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

U is another **orthogonal matrix**, (different from **V**) i.e., it represents a **second rotation**.

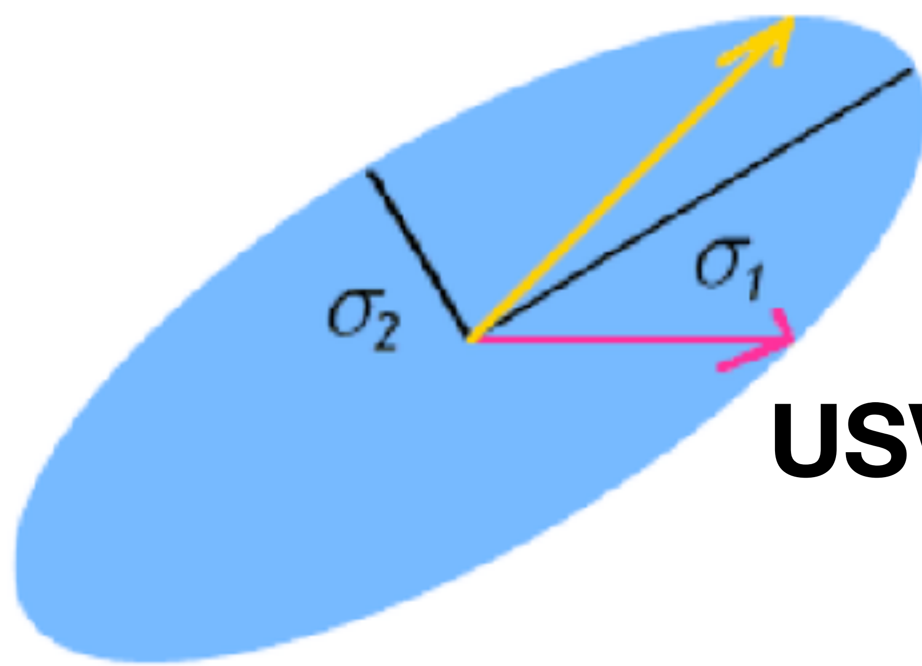
$\mathbf{U}\mathbf{U}^T = \mathbf{U}^T\mathbf{U} = \mathbf{I}$



$$\underset{\mathbf{A}}{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}} \underset{\mathbf{x}}{\mathbf{x}} = \underset{\mathbf{U}}{\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}} \underset{\mathbf{S}}{\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}} \underset{\mathbf{V}^T}{\begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix}} \underset{\mathbf{x}}{\mathbf{x}}$$

Singular Value Decomposition (SVD)

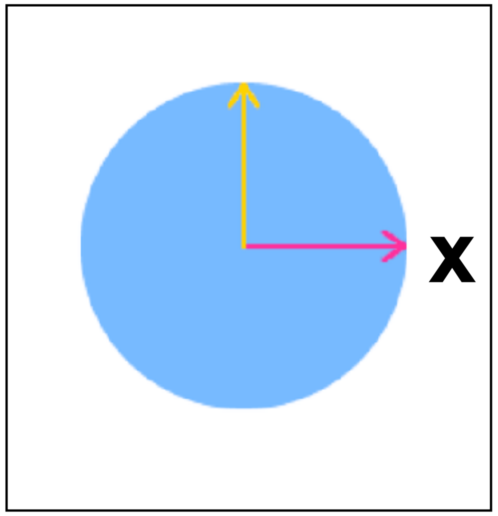
U is another **orthogonal matrix**, (different from **V**) i.e., it represents a **second rotation**.
$\mathbf{U}\mathbf{U}^T = \mathbf{U}^T\mathbf{U} = \mathbf{I}$



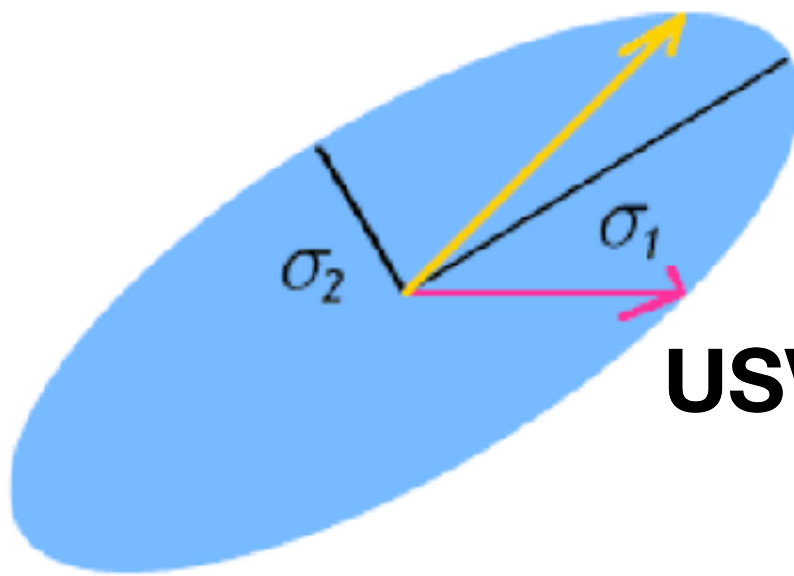
$\mathbf{USV}^T\mathbf{x}$ or $\mathbf{Ax}$

$$\underset{\mathbf{A}}{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}} \underset{\mathbf{x}}{\mathbf{x}} = \underset{\mathbf{U}}{\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}} \underset{\mathbf{S}}{\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}} \underset{\mathbf{V}^T}{\begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix}} \underset{\mathbf{x}}{\mathbf{x}}$$

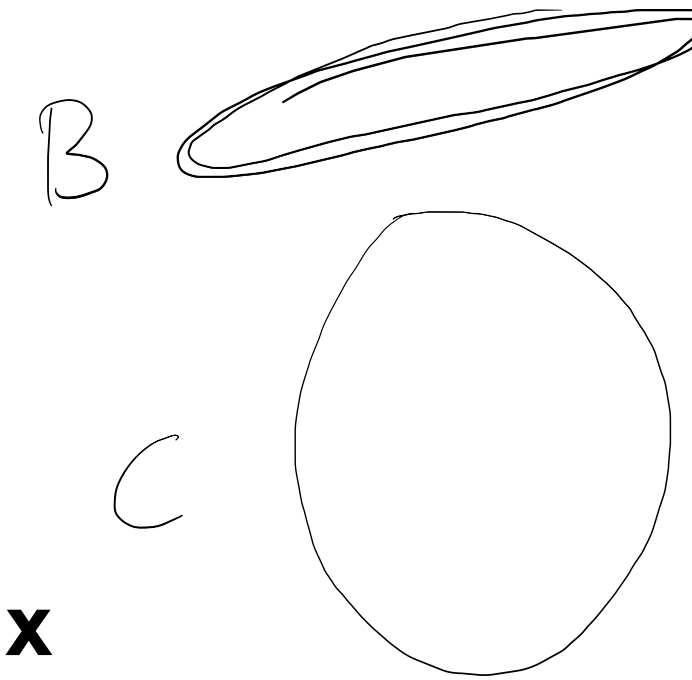
Singular Value Decomposition (SVD)



Notice how much
the vector changes
under the influence of **A**.



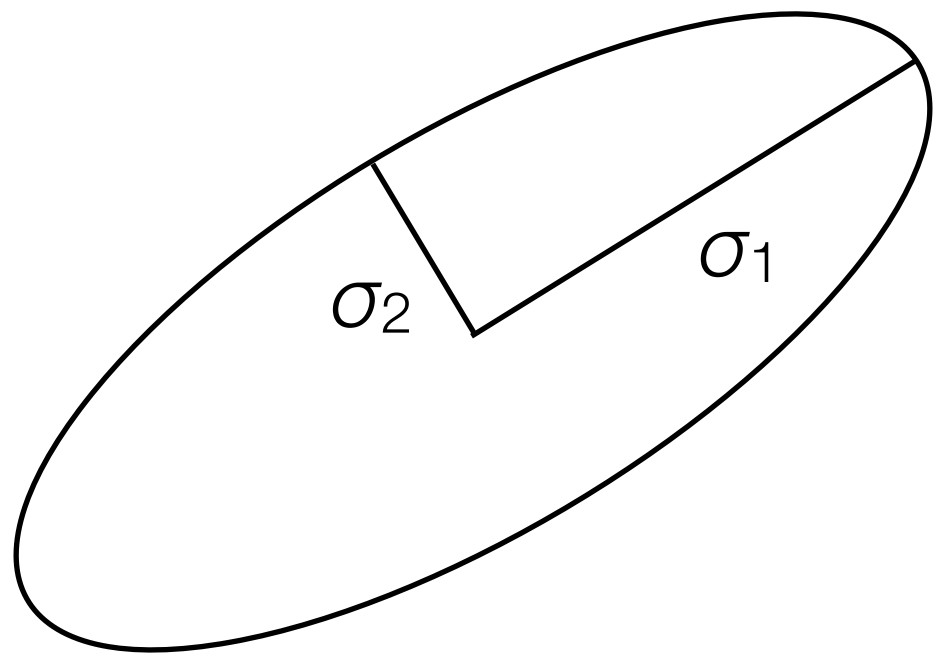
USV^T**x** or **Ax**



$$\underbrace{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}}_{\mathbf{A}} \underbrace{\mathbf{x}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}}_{\mathbf{S}} \underbrace{\begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix}}_{\mathbf{V}^T} \underbrace{\mathbf{x}}_{\mathbf{x}}$$

Singular Value Decomposition (SVD)

Circle is converted to a rotated ellipse.

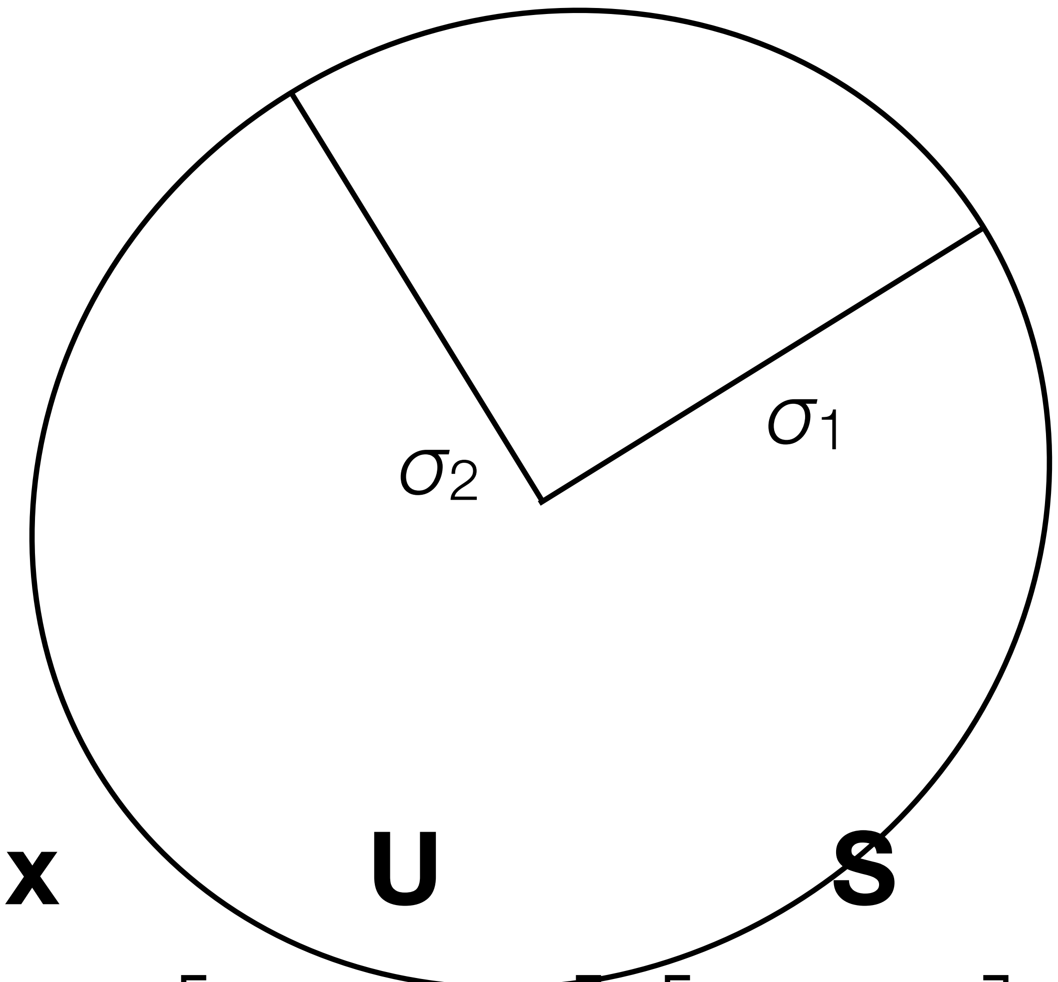


A **x** **U** **S** $\mathbf{V}^T$ **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

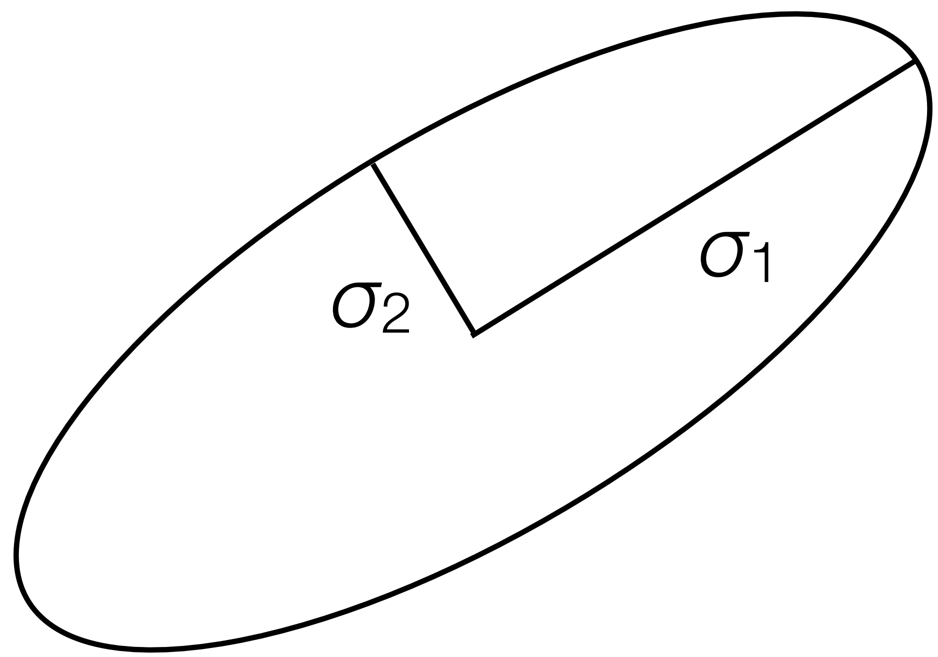
If $\sigma_1 \sim= \sigma_2$, both singular values are large, ellipse is nearly a circle.



$$\underbrace{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}}_{\mathbf{A}} \underbrace{\mathbf{x}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}}_{\mathbf{S}} \underbrace{\begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix}}_{\mathbf{V}^T} \underbrace{\mathbf{x}}_{\mathbf{x}}$$

Singular Value Decomposition (SVD)

The more interesting thing happens if $\sigma_2 = 0$...

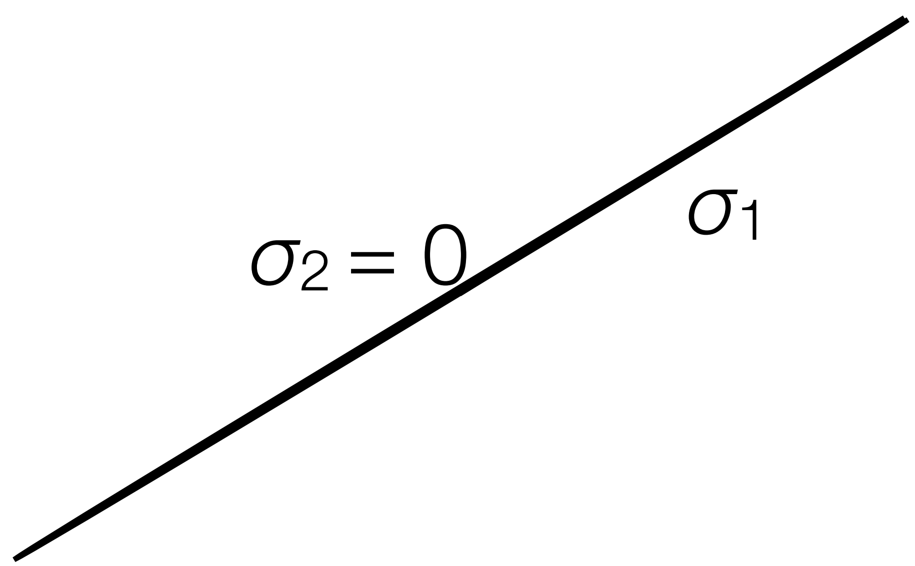


$$\underset{\mathbf{A}}{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}} \underset{\mathbf{x}}{\mathbf{x}} = \underset{\mathbf{U}}{\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}} \underset{\mathbf{S}}{\begin{bmatrix} \sigma_1 & 0 \\ 0 & 0 \end{bmatrix}} \underset{\mathbf{V}^T}{\begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix}} \underset{\mathbf{x}}{\mathbf{x}}$$

Singular Value Decomposition (SVD)

The more interesting thing happens if $\sigma_2 = 0$...

The ellipse collapses to a line.



$\mathbf{A}$ $\mathbf{x}$ $\mathbf{U}$ $\mathbf{S}$ $\mathbf{V}^T$ $\mathbf{x}$

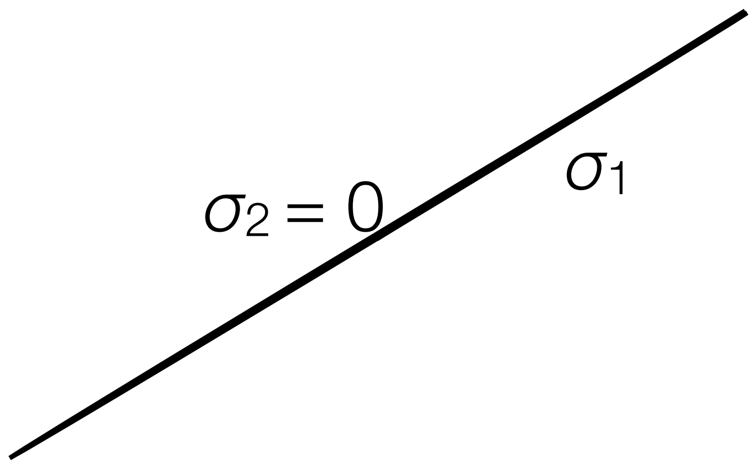
$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ v_{12} & v_{22} \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

The more interesting thing happens if $\sigma_1 = 0$...

The ellipse collapses to a line.

The multiplication can be re-written as:-



$$\begin{array}{cccccc} \mathbf{A} & \mathbf{x} & \mathbf{U} & \mathbf{S} & \mathbf{V}^T & \mathbf{x} \\ \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & \mathbf{x} = & \begin{bmatrix} u_{11} & \quad \\ u_{21} & \quad \end{bmatrix} & \begin{bmatrix} \sigma_1 & \quad \\ & \end{bmatrix} & \begin{bmatrix} v_{11} & v_{21} \\ & \end{bmatrix} & \mathbf{x} \end{array}$$

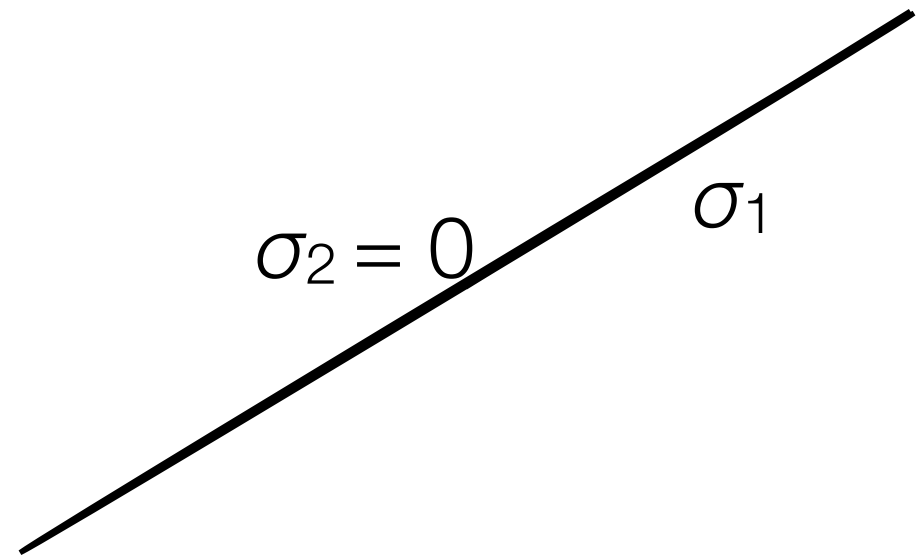
$$\boxed{\begin{bmatrix} u_{11} \\ u_{21} \end{bmatrix} \begin{bmatrix} \sigma_1 \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \end{bmatrix} \begin{bmatrix} x \end{bmatrix}}$$

Singular Value Decomposition (SVD)

The more interesting thing happens if $\sigma_1 = 0$...

The ellipse collapses to a line.

The multiplication can be re-written as:-



$$\begin{array}{cccccc} \mathbf{A} & \mathbf{x} & \mathbf{U} & \mathbf{S} & \mathbf{V}^T & \mathbf{x} \\ & & \text{(first column)} & \text{(first value)} & \text{(first row)} & \\ \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & \mathbf{x} = & \begin{bmatrix} u_{11} & \\ u_{21} & \end{bmatrix} & \begin{bmatrix} \sigma_1 & \\ & \end{bmatrix} & \begin{bmatrix} v_{11} & v_{21} \\ & \end{bmatrix} & \mathbf{x} \end{array}$$

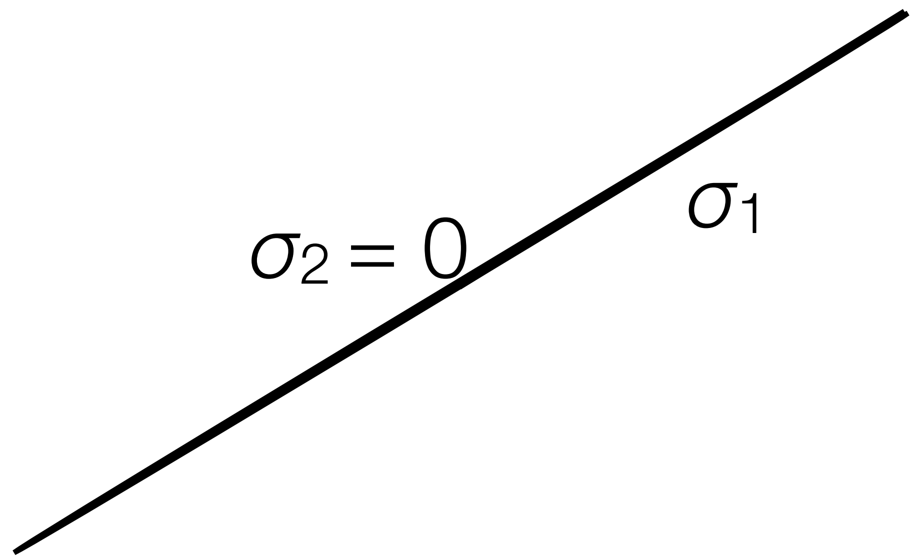
(or first column of **V**)^T

Singular Value Decomposition (SVD)

The more interesting thing happens if $\sigma_1 = 0$...

The ellipse collapses to a line.

The multiplication can be re-written as:-



A **x** **U** **S** $\mathbf{V}^T$ **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \begin{bmatrix} u_{11} & \phantom{u_{12}} \\ u_{21} & \end{bmatrix} \begin{bmatrix} \sigma_1 & \\ & \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ & \end{bmatrix} \mathbf{x}$$

Singular Value Decomposition (SVD)

If you set **x** = the second column of **V**,

$$\begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix}^T \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix} = 0$$

$$\begin{bmatrix} v_{11} & v_{21} \end{bmatrix} \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix} = 0$$

$\sigma_2 = 0$

σ_1

$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}} \\ v_{21} & \end{bmatrix}$$

a

$v_{11} v_{12} + v_{21} v_{22} = 0$

$A x = 0$

if x = last col. of V

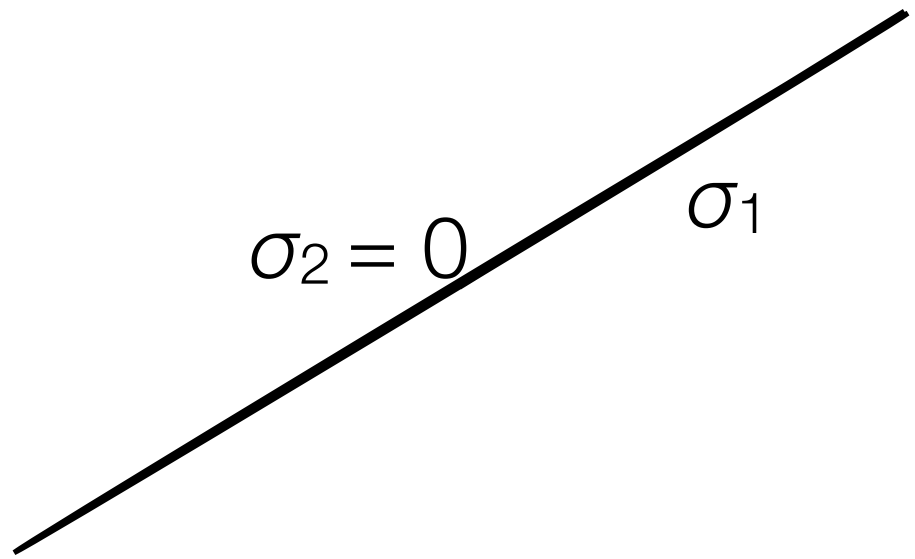
A **x** **U** **S** $\mathbf{V}^T$ **x**

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \begin{bmatrix} u_{11} & \\ u_{21} & \end{bmatrix} \begin{bmatrix} \sigma_1 & \\ & \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ & \end{bmatrix} \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix}$$

0 0 0

Singular Value Decomposition (SVD)

If you set **x** = the second column of **V**, the entire equation goes to 0, since (first column of **V**)^T(second column of **V**) = 0 due to orthogonality of **V**.

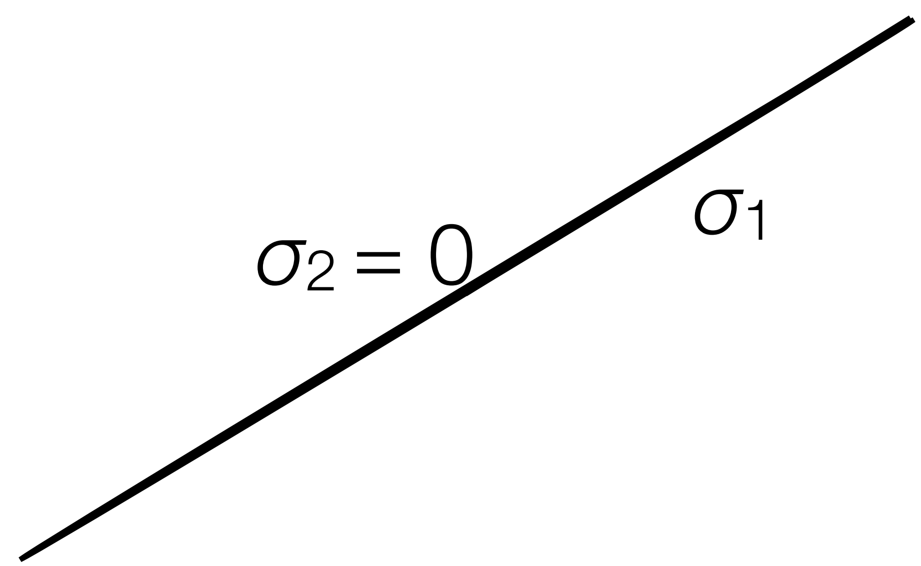


$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}} \\ v_{21} & \end{bmatrix}$$

$$\begin{array}{cccccc} \mathbf{A} & \mathbf{x} & \mathbf{U} & \mathbf{S} & \mathbf{V}^T & \mathbf{x} \\ \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & \mathbf{x} = & \begin{bmatrix} u_{11} & \\ u_{21} & \end{bmatrix} & \begin{bmatrix} \sigma_1 & \\ & \end{bmatrix} & \begin{bmatrix} v_{11} & v_{21} \\ & \end{bmatrix} & \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix} = \mathbf{0} \end{array}$$

Singular Value Decomposition (SVD)

If you set **x** = the second column of **V**, the entire equation goes to 0, since (first column of **V**)^T(second column of **V**) = 0 due to orthogonality of **V**.



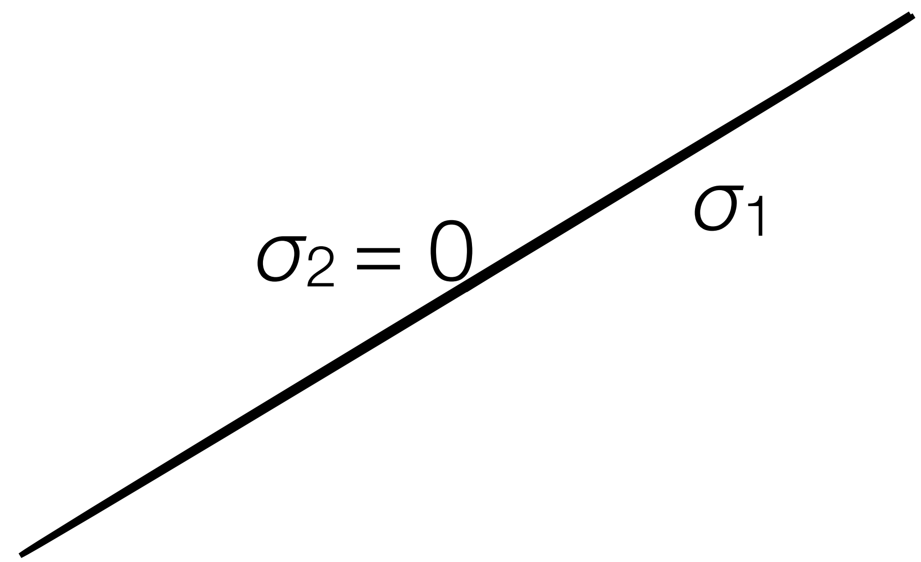
$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}} \\ v_{21} & \end{bmatrix}$$

$$\begin{array}{cccccc} \mathbf{A} & \mathbf{x} & \mathbf{U} & \mathbf{S} & \mathbf{V}^T & \mathbf{x} \\ \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & \mathbf{x} = & \begin{bmatrix} u_{11} & \\ u_{21} & \end{bmatrix} & \begin{bmatrix} \sigma_1 & \\ & \end{bmatrix} & \begin{bmatrix} v_{11} & v_{21} \\ & \end{bmatrix} & \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix} = \mathbf{0} \end{array}$$

$$v_{11}v_{12} + v_{21}v_{22} = 0$$

Singular Value Decomposition (SVD)

If you set **x** = the second column of **V**, the entire equation goes to 0, since (first column of **V**)^T(second column of **V**) = 0 due to orthogonality of **V**.

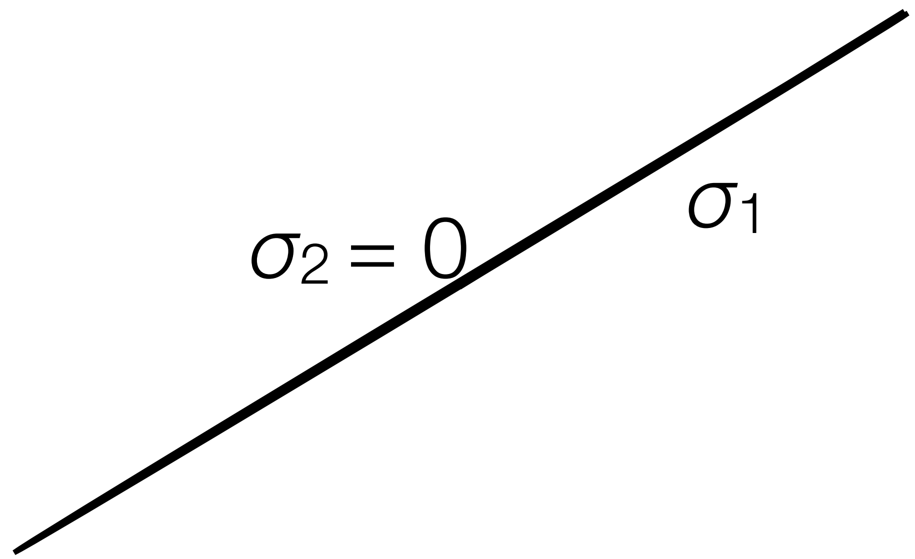


$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{v_{12}} \\ v_{21} & \boxed{v_{22}} \end{bmatrix}$$

$$\begin{array}{cccccc} \mathbf{A} & \mathbf{x} & \mathbf{U} & \mathbf{S} & \mathbf{V}^T & \mathbf{x} \\ \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & \mathbf{x} = & \begin{bmatrix} u_{11} & \\ u_{21} & \end{bmatrix} & \begin{bmatrix} \sigma_1 & \\ & \end{bmatrix} & \begin{bmatrix} v_{11} & v_{21} \\ & \end{bmatrix} & \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix} = \mathbf{0} \end{array}$$

Singular Value Decomposition (SVD)

If you set $\mathbf{x}$ = the second column of $\mathbf{V}$, the entire equation goes to 0, since (first column of $\mathbf{V}$)^T(second column of $\mathbf{V}$) = 0 due to orthogonality of $\mathbf{V}$.



$$\mathbf{V} = \begin{bmatrix} v_{11} & \underbrace{\boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}}}_{\mathbf{v}_2} \\ v_{21} & \end{bmatrix}$$

$$\overset{\mathbf{A}}{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}} \overset{\mathbf{x}}{\mathbf{x}} = \overset{\mathbf{U}}{\begin{bmatrix} u_{11} & \\ u_{21} & \end{bmatrix}} \overset{\mathbf{S}}{\begin{bmatrix} \sigma_1 & \\ & \end{bmatrix}} \overset{\mathbf{V}^T}{\begin{bmatrix} v_{11} & v_{21} \\ & \end{bmatrix}} \overset{\mathbf{x}}{\underbrace{\begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix}}_{\mathbf{v}_2}} = \mathbf{0}$$

$\mathbf{Ax} = \mathbf{0}$ for $\mathbf{x} = \mathbf{v}_2$

Singular Value Decomposition (SVD)

What this means is, to solve **A x** = **0**,

you can set **x** to the vector in **V** whose singular value is 0.

σ_2 is 0, so use $\mathbf{v}_2$.

$\sigma_2 = 0$ σ_1

$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}} \\ v_{21} & \end{bmatrix}$$

$\mathbf{v}_2$

$$\begin{array}{cccccc} \mathbf{A} & \mathbf{x} & \mathbf{U} & \mathbf{S} & \mathbf{V}^T & \mathbf{x} \\ \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} & \mathbf{x} = & \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} & \begin{bmatrix} \sigma_1 & 0 \\ 0 & \boxed{0} \end{bmatrix} & \begin{bmatrix} v_{11} & v_{21} \\ \boxed{v_{12} \quad v_{22}} \end{bmatrix} & \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix} = \mathbf{0} \\ & & & & & \mathbf{v}_2 \end{array}$$

Ax = **0** for **x** = $\mathbf{v}_2$

Singular Value Decomposition (SVD)

In practice, you may not have a matrix **A** with a 0 singular value.

$\sigma_2 = 0$ σ_1

$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}} \\ v_{21} & \end{bmatrix}$$

$\mathbf{v}_2$

$$\underbrace{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}}_{\mathbf{A}} \underbrace{\mathbf{x}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}}_{\mathbf{U}} \underbrace{\begin{bmatrix} \sigma_1 & 0 \\ 0 & \boxed{0} \end{bmatrix}}_{\mathbf{S}} \underbrace{\begin{bmatrix} v_{11} & v_{21} \\ \boxed{v_{12} \quad v_{22}} \end{bmatrix}}_{\mathbf{V}^T} \underbrace{\begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix}}_{\mathbf{x}} = \mathbf{0}$$

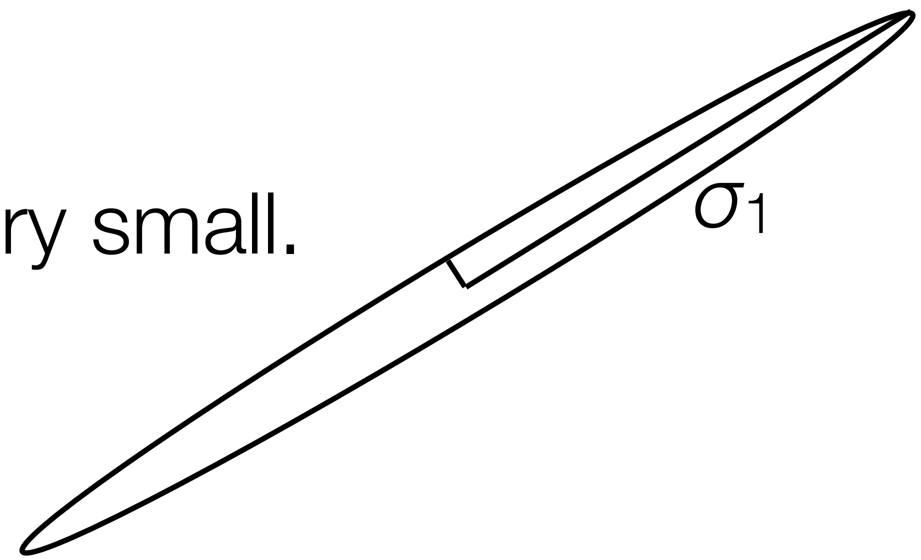
$\mathbf{v}_2$

$\mathbf{Ax} = \mathbf{0}$ for $\mathbf{x} = \mathbf{v}_2$

Singular Value Decomposition (SVD)

In practice, you may not have a matrix **A** with a 0 singular value.
But you may have **A** with a fairly small singular value.

σ_2 is very small.



$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}} \\ v_{21} & \end{bmatrix}$$

$\mathbf{v}_2$

$$\underset{\mathbf{A}}{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}} \underset{\mathbf{x}}{\mathbf{x}} = \underset{\mathbf{U}}{\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}} \underset{\mathbf{S}}{\begin{bmatrix} \sigma_1 & 0 \\ 0 & \boxed{\sigma_2} \end{bmatrix}} \underset{\mathbf{V}^T}{\begin{bmatrix} v_{11} & v_{21} \\ \boxed{v_{12} \quad v_{22}} \end{bmatrix}} \underset{\mathbf{x}}{\begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix}}$$

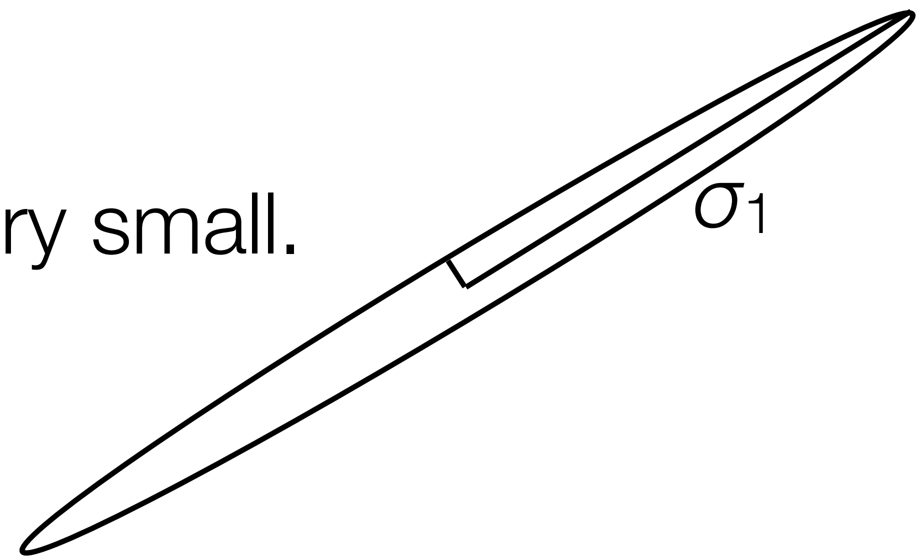
$\mathbf{v}_2$

Ax = **vector with very small length** for **x** = $\mathbf{v}_2$

Singular Value Decomposition (SVD)

So you can still solve **A x** = **0**,
approximately, if not exactly.

σ_2 is very small.



$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}} \\ v_{21} & \end{bmatrix}$$

$\mathbf{v}_2$

$$\begin{array}{cccccc} \textbf{A} & \textbf{x} & \textbf{U} & \textbf{S} & \textbf{V}^T & \textbf{x} \end{array}$$

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \mathbf{x} = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & \boxed{\sigma_2} \end{bmatrix} \begin{bmatrix} v_{11} & v_{21} \\ \boxed{v_{12} \quad v_{22}} \end{bmatrix} \begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix}$$

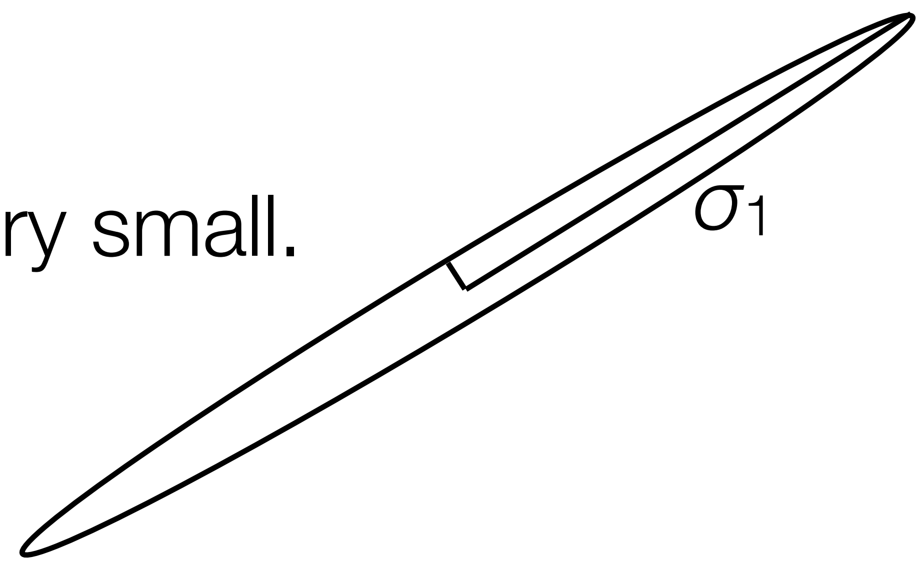
$\mathbf{v}_2$

Ax = **vector with very small length** for **x** = $\mathbf{v}_2$

Singular Value Decomposition (SVD)

So you can still solve $\mathbf{A}\,\mathbf{x} = \mathbf{0}$
approximately, if not exactly.

σ_2 is very small.



$$\mathbf{V} = \begin{bmatrix} v_{11} & \boxed{\begin{matrix} v_{12} \\ v_{22} \end{matrix}} \\ v_{21} & \end{bmatrix}$$

$\mathbf{v}_2$

$$\underset{\mathbf{A}}{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}} \underset{\mathbf{x}}{\mathbf{x}} = \underset{\mathbf{U}}{\begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix}} \underset{\mathbf{S}}{\begin{bmatrix} \sigma_1 & 0 \\ 0 & \boxed{\sigma_2} \end{bmatrix}} \underset{\mathbf{V}^T}{\begin{bmatrix} v_{11} & v_{21} \\ \boxed{v_{12} \quad v_{22}} \end{bmatrix}} \underset{\mathbf{x}}{\begin{bmatrix} v_{12} \\ v_{22} \end{bmatrix}}$$

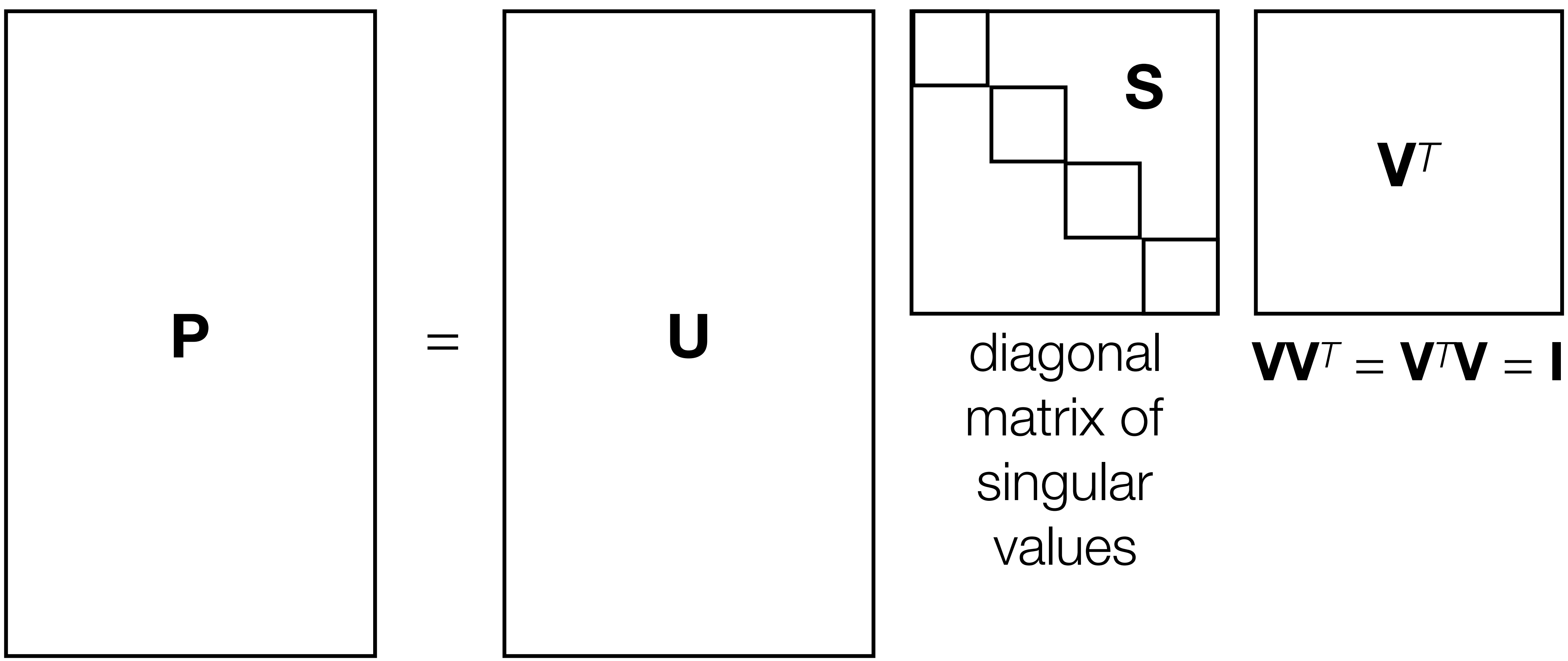
$\mathbf{v}_2$

Ax = **vector with very small length** of $\underline{\sigma_2}$ for $\mathbf{x} = \mathbf{v}_2$

The minimum possible value!

Singular Value Decomposition (SVD)

Similarly you can solve $\mathbf{P}\,\mathbf{q} = \mathbf{0}$.
The same approach works for dimensions more than 2.
Also works for a tall matrix!



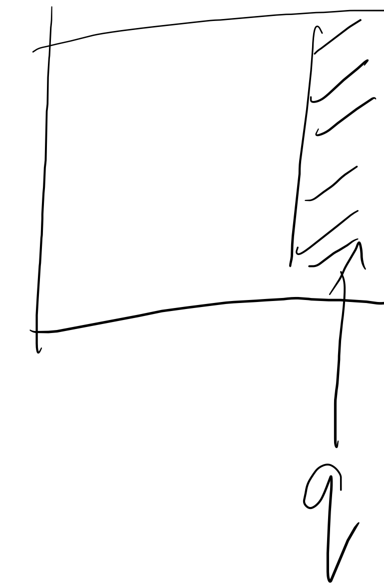
diagonal matrix of singular values

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

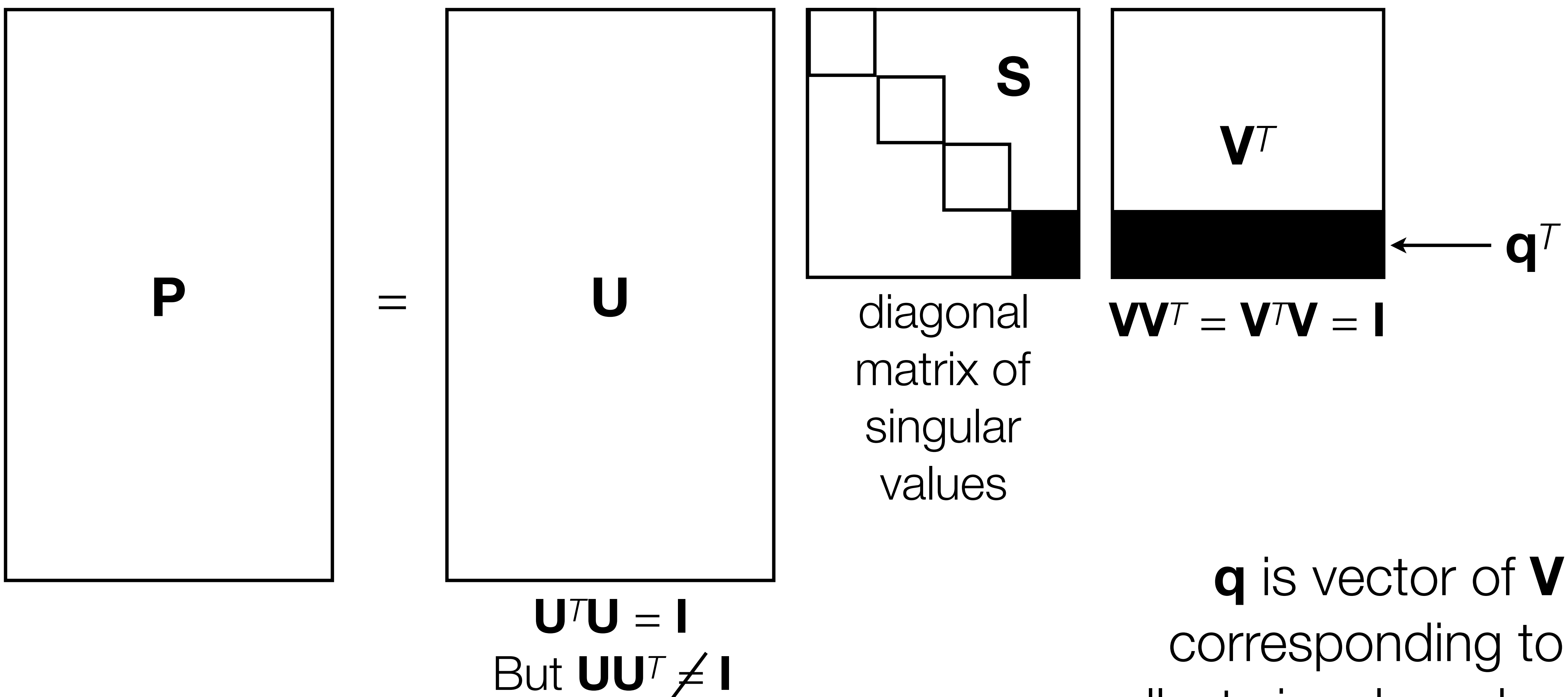
$\mathbf{U}^T\mathbf{U} = \mathbf{I}$

But $\mathbf{U}\mathbf{U}^T \neq \mathbf{I}$

Singular Value Decomposition (SVD)



Similarly you can solve $\mathbf{P}\,\mathbf{q} = \mathbf{0}$.
The same approach works for dimensions more than 2.
Also works for a tall matrix!



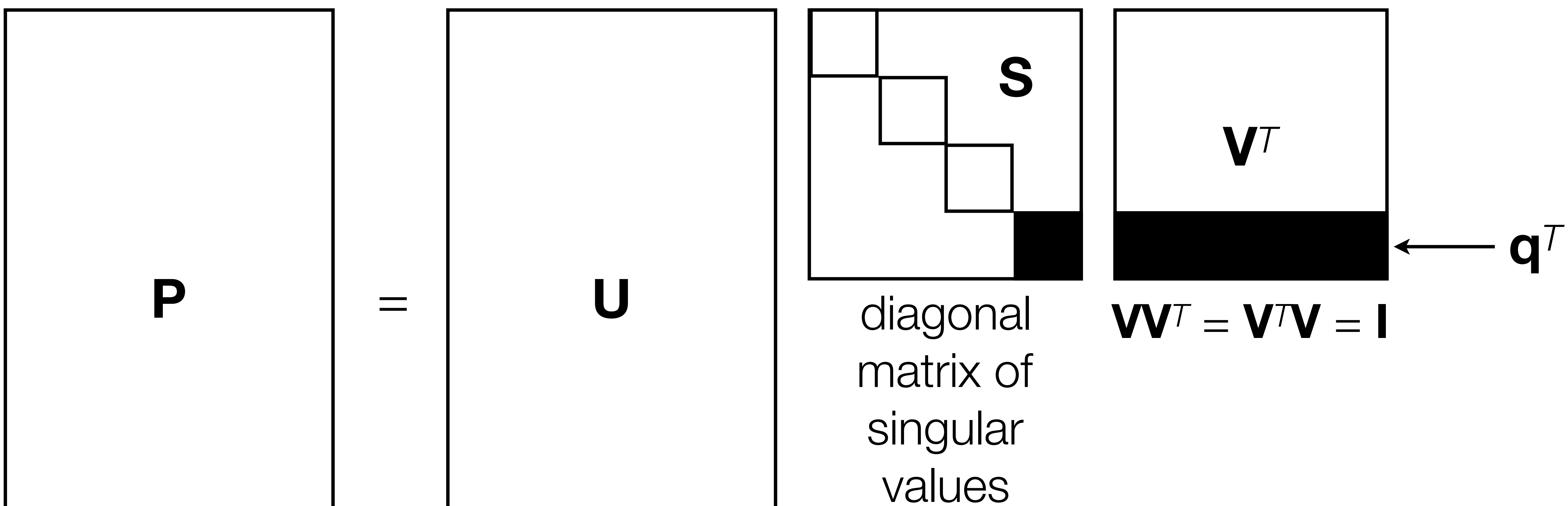
q is vector of **V** corresponding to smallest singular value

Singular Value Decomposition (SVD)

In MATLAB, you can use the `svd()` function:

```
[U, S, V] = svd( P, ‘econ’ );
```

Usually, the last column of **V** corresponds to the smallest singular value (or the last diagonal element in **S**).



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

But $\mathbf{U}\mathbf{U}^T \neq \mathbf{I}$

q is vector of **V** corresponding to smallest singular value

Solution for homogeneous least squares

Solving $\mathbf{P}\,\mathbf{q} = \mathbf{0}$:

You get **q** as follows:

1) Take the singular value decomposition (SVD) of **P**.

The SVD of **P** gives three matrices **U, S, V** such that

$$\mathbf{P} = \mathbf{U}\,\mathbf{S}\,\mathbf{V}^{T}$$

where **U** and **V** are orthogonal matrices, while **S** is a diagonal matrix of **singular values**.

2) Get **q** as the column vector in **V** corresponding to the location of the **smallest singular value** in **S**.

Why is it called homogeneous least squares?

$$\mathbf{P}\,\mathbf{q} = \mathbf{0}$$

length of Ax was the smallest possible

is a **homogeneous linear system**.

Pq

Solving it provides you a **least squares** solution, because you are finding the smallest sum of squares of the elements in vector **Pq,**

i.e., you are minimizing $||\mathbf{Pq}||^2$

Nature of solution **q**

q is a vector with length 1.

The length of the vector serves as a constraint to reduce the number of dimensions in the final transform from 9 to 8.

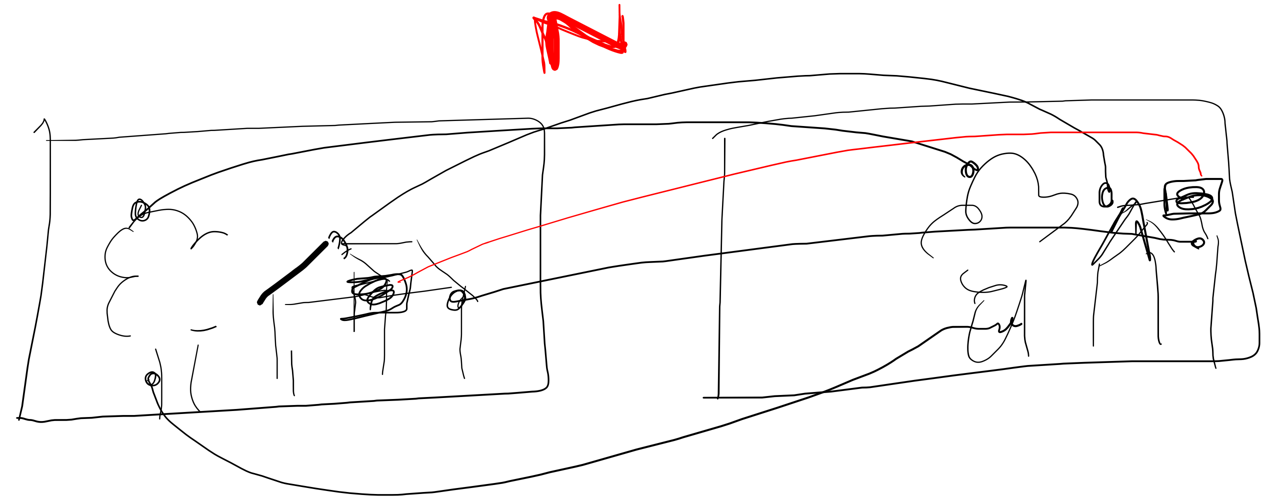
On Assignment 2,

You will need to use `svd()` to solve for the values in $\mathbf{q}$, i.e., you will need to use **homogeneous least squares.**

Remember to re-structure the elements in $\mathbf{q}$ to form the homography matrix!

$$\mathbf{q} = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \\ i \end{bmatrix} \longrightarrow \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

$$\begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix}$$



RANSAC algorithm

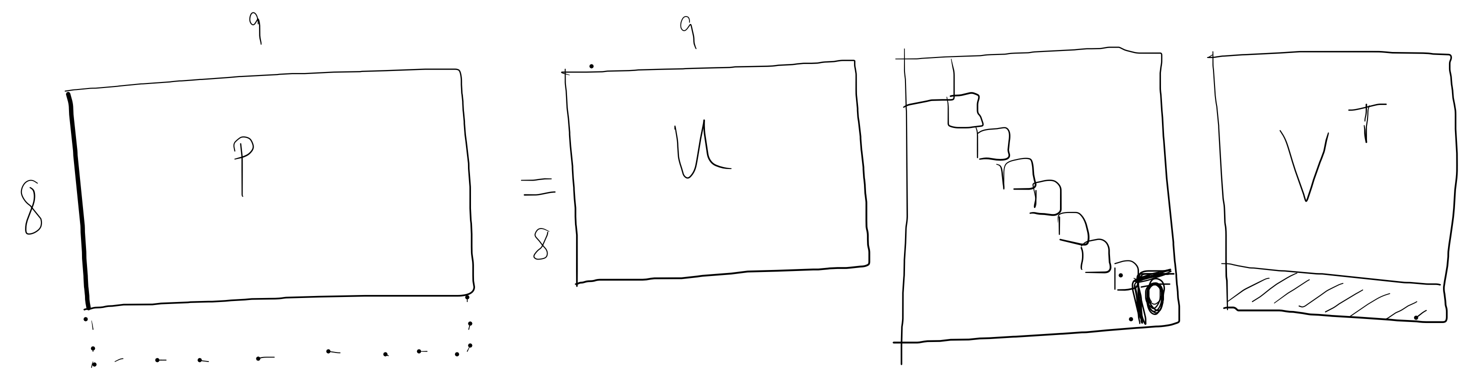
$$\begin{bmatrix}\hat{x}'\\ \hat{y}'\\ \hat{w}'\end{bmatrix} = \begin{bmatrix} a & b & c\\ d & e & f\\ g & h & i\end{bmatrix}\begin{bmatrix}x\\ y\\ 1\end{bmatrix}, \quad x' = \frac{\hat{x}'}{\hat{w}'}, \quad y' = \frac{\hat{y}'}{\hat{w}'}$$

$$[U, S, V] = \mathrm{svd}(P)$$

$$\underbrace{\begin{bmatrix} \\ P \\ \\ \end{bmatrix}}_{9} \begin{bmatrix} a\\ b\\ c\\ d\\ e\\ f\\ g\\ h\\ i\end{bmatrix} = \begin{bmatrix}0\\ \vdots\\ \vdots \\ 0\end{bmatrix}$$

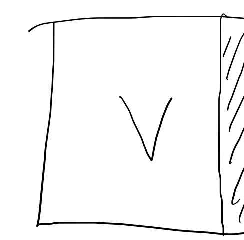
8

9×1



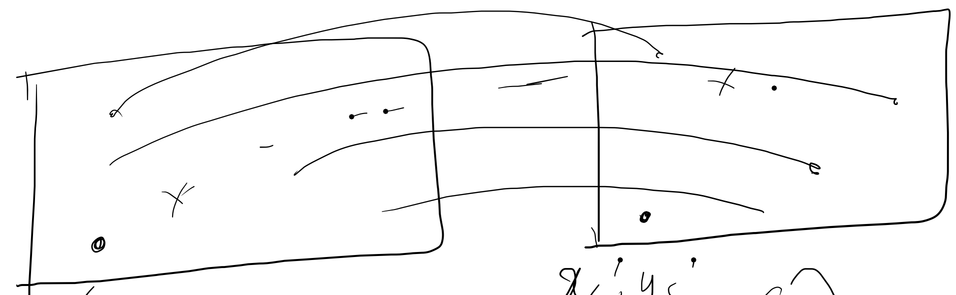
$$\underline{\underline{Pq = 0}}$$

q = last col of V



$^{10}C_4$

$N = 10$



x_s, y_s $\quad$ x_s', y_s'

$$\left.\begin{matrix}1\\2\\3\\4\end{matrix}\right\} T_{1,2,3,4}$$

$$(x_{5e}', y_{5e}') = T_{1,2,3,4} \to x_s, y_s$$

$$d_5 = \sqrt{(x_{se}' - x_s')^2 + (y_{se}' - y_{s'})^2}$$

9

$T_{5,7,8,10}$

5	d_5	$< th$
6	d_6	$< th$
7	d_7	$< th$
8	d_8	$< th$
9	d_9	$< th$
10	d_{10}	$< th.$

P

The least squares approaches discussed till now
use n >= 4~~/5~~ 2 correspondences to estimate
the transformation.

But often, not all n points are good. Some of them
may be outliers.

The RANSAC algorithm helps deal with outliers.

Exhaustive Approach to Eliminate Outliers

4

Repeat the following for all (*n choose k*) combinations of correspondences:

Get the current combination of *k* correspondences.

$$\underset{8\times 9}{P} = U \, S \, V^T$$

Estimate the transform **A** from the *k* correspondences.

Compute the error of estimation for all *n* points, i.e., compute:

$$e = (x'_{actual} - x'_{estim})^2 + (y'_{actual} - y'_{estim})^2$$

$${}_{10}C_4 = \frac{10 \times 9 \times 8 \times 7}{4 \times 3 \times 2 \times 1} = 30$$

$${}_{100}C_4 = \frac{100 \times 99 \times 98 \times 97}{4 \times 3 \times 2 \times 1}$$

Count the number of points whose error *e* falls below a threshold *t*.

Retain the transform corresponding to the maximum point count.

Exhaustive Approach to Eliminate Outliers

Repeat the following for all (*n choose k)* combinations of correspondences:

Get the current combination of *k* correspondences.

Estimate the transform **A** from the *k* correspondences.

Compute the error of estimation for all *n* points, i.e., compute:

$$e = (x'_{actual} - x'_{estim})^2 + (y'_{actual} - y'_{estim})^2$$

Count the number of points whose error *e* falls below a threshold *t*.

Retain the transform corresponding to the maximum point count.

Problem: *n choose k* can be too large to make the computation practical.

RANSAC: RANdom SAmple Consensus

101 025

Repeat the following for N_{ransac} steps:

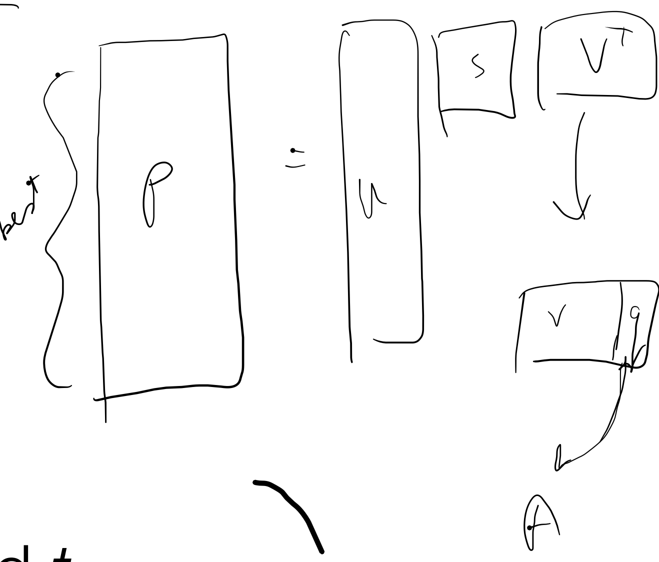
Choose k correspondences at random, where k is the minimum number of correspondences needed to estimate the transform.

Estimate the transform **A** from the k correspondences.

Compute the error of estimation for all n points, i.e., compute:

$$e = (x'_{actual} - x'_{estim})^2 + (y'_{actual} - y'_{estim})^2$$

indices best



Count the number of points whose error e falls below a threshold t.

If the count exceeds the maximum count up till now, update the best transform $\mathbf{A}_{\text{best}}$ to be **A**. indices$_{\text{best}}$ to be indices.

$$\begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ y_1 & y_2 & y_3 & \cdots & y_n \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}$$

$$pts2e = \begin{bmatrix} \hat{x}_1' & \hat{x}_2' & \cdots \\ \hat{y}_1' & \hat{y}_2' & \cdots \\ \hat{w}_1' & \hat{w}_2' & \cdots \end{bmatrix}$$

RANSAC: RANdom SAmple Consensus

Repeat the following for N_{ransac} steps:

Choose k correspondences at random, where k is the minimum number of correspondences needed to estimate the transform.

Estimate the transform **A** from the k correspondences.

Compute the error of estimation for all n points, i.e., compute:

$$e = (x'_{actual} - x'_{estim})^2 + (y'_{actual} - y'_{estim})^2$$

Count the number of points whose error e falls below a threshold t.

If the count exceeds the maximum count up till now, update the best transform $\mathbf{A}_{\text{best}}$ to be **A**.

Incredibly simple algorithm!!

RANSAC: RANdom SAmple Consensus

Repeat the following for N_{ransac} steps:

Choose k correspondences at random, where k is the minimum number of correspondences needed to estimate the transform.

Estimate the transform **A** from the k correspondences.

Compute the error of estimation for all n points, i.e., compute:

$$e = (x'_{actual} - x'_{estim})^2 + (y'_{actual} - y'_{estim})^2$$

Count the number of points whose error e falls below a threshold t.

If the count exceeds the maximum count up till now, update the best transform $\mathbf{A}_{\text{best}}$ to be **A**.

Incredibly simple algorithm!! Can be used to remove outliers in any model estimation involving several data points.