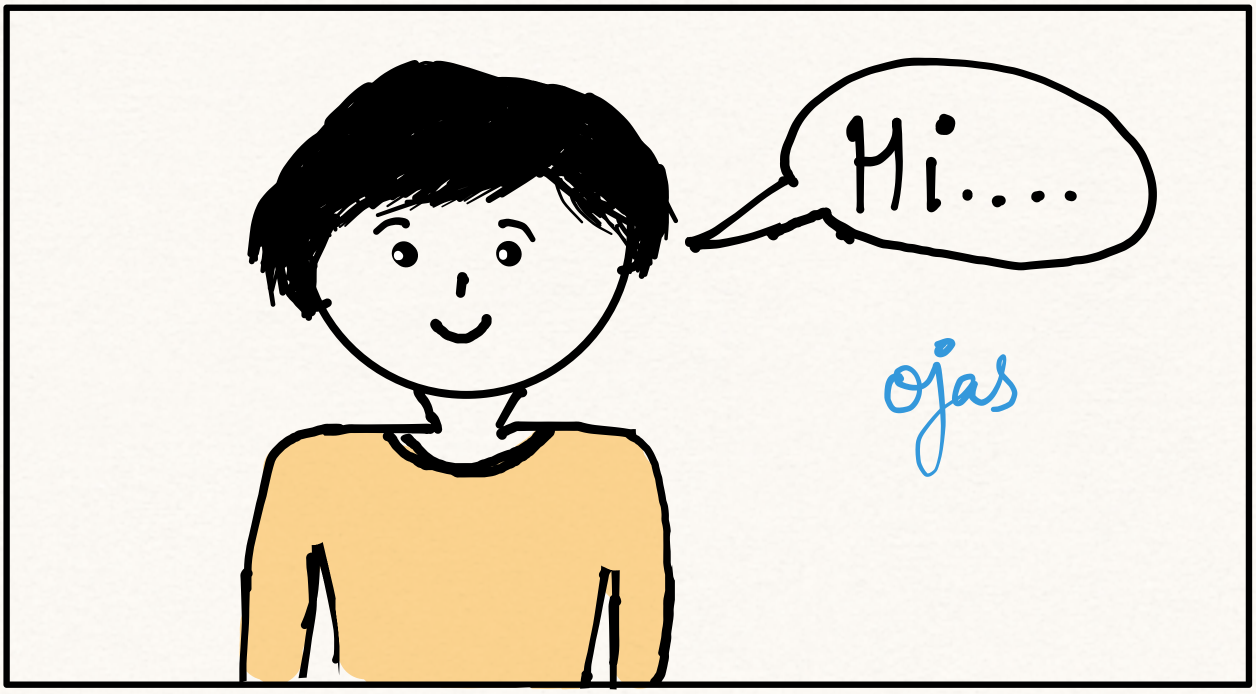
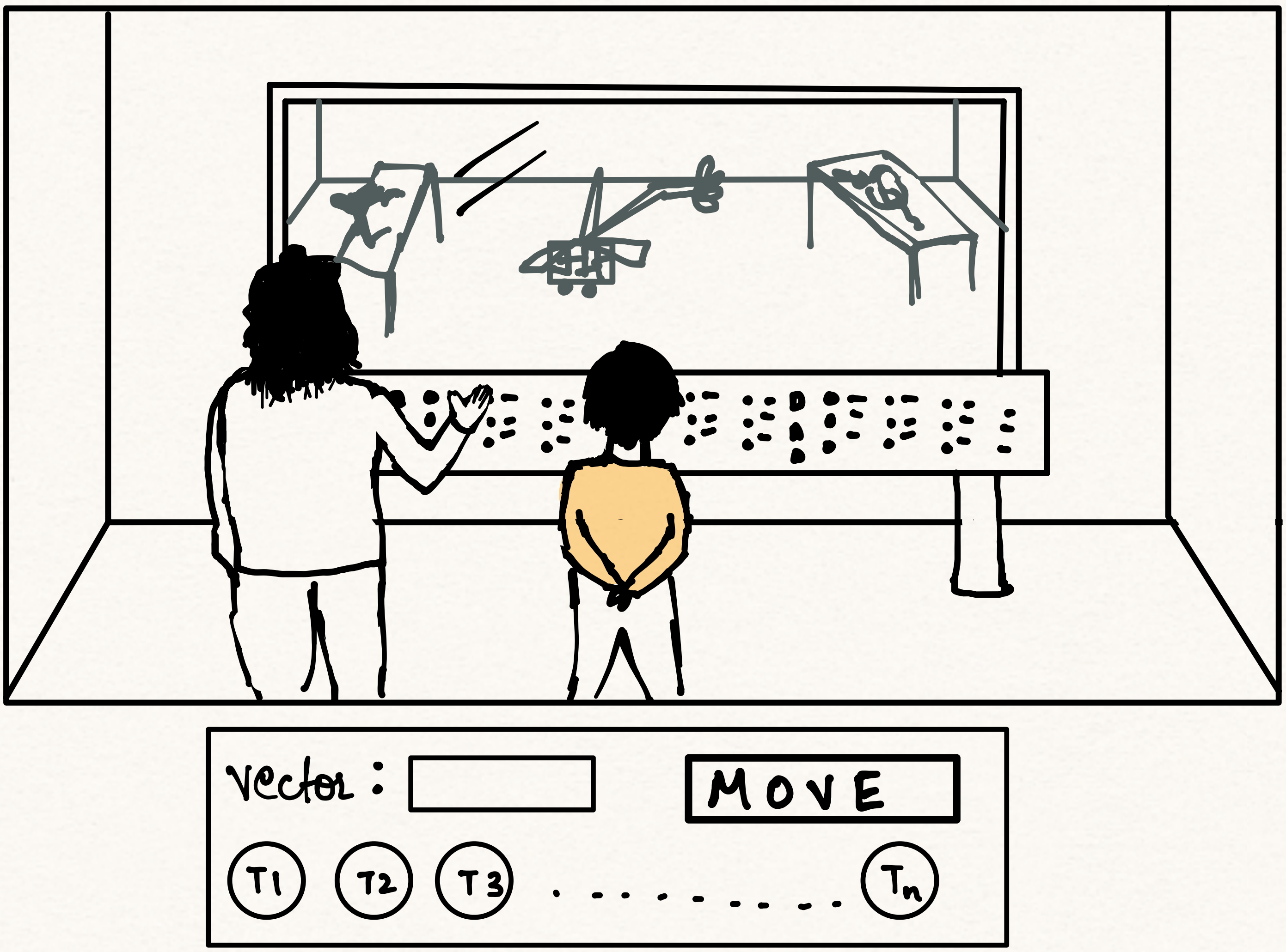
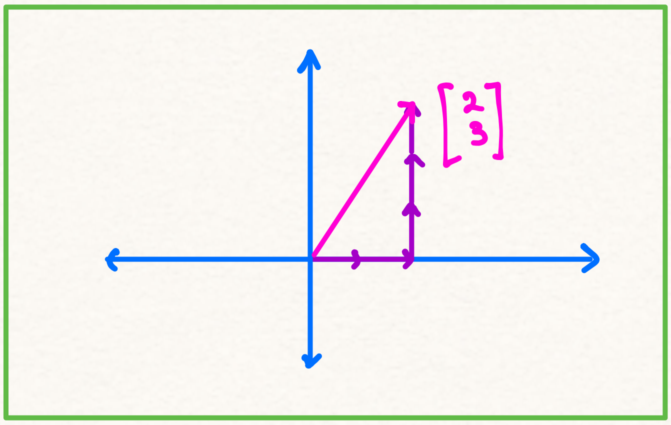
video : https://youtu.be/5bLgpFBx74A

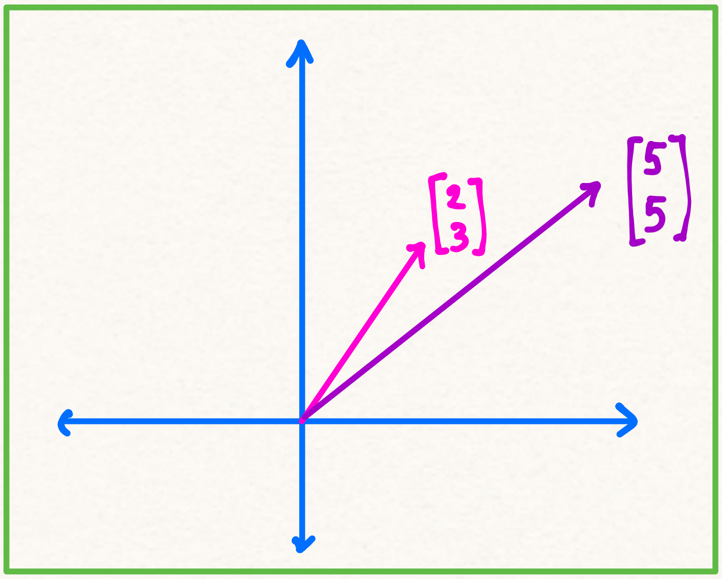


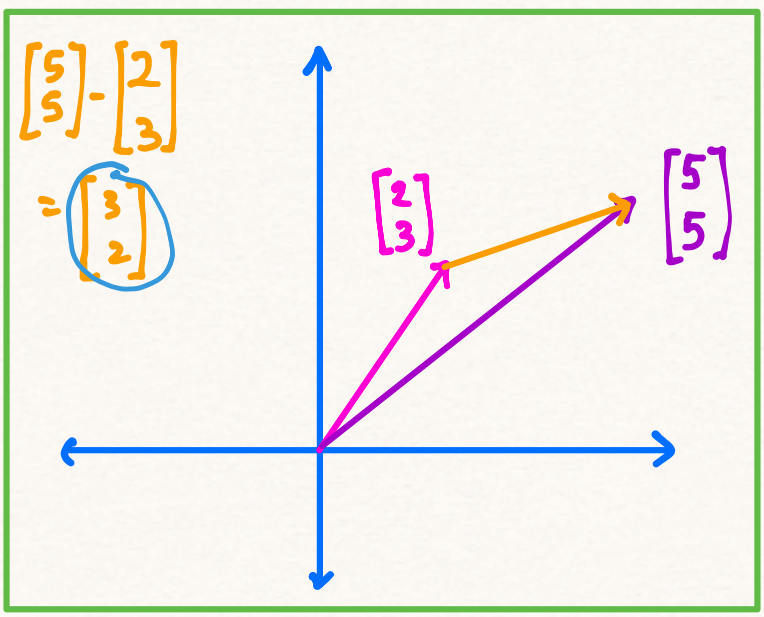


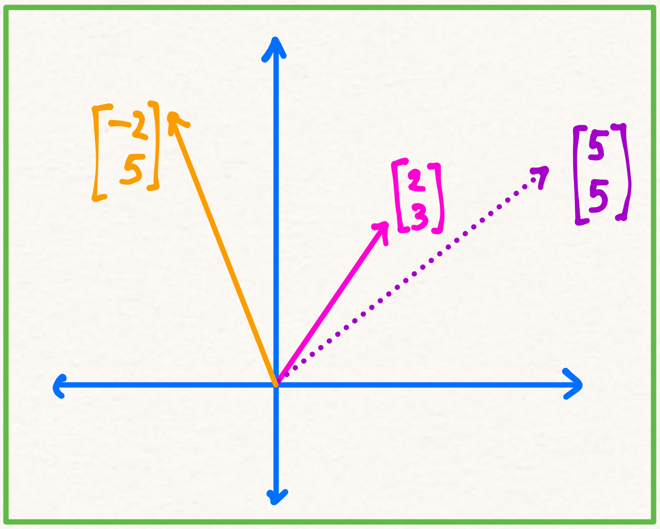
Robot's understanding



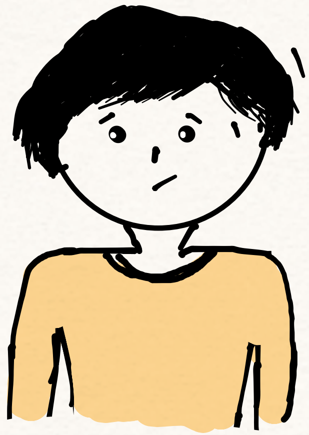
Task

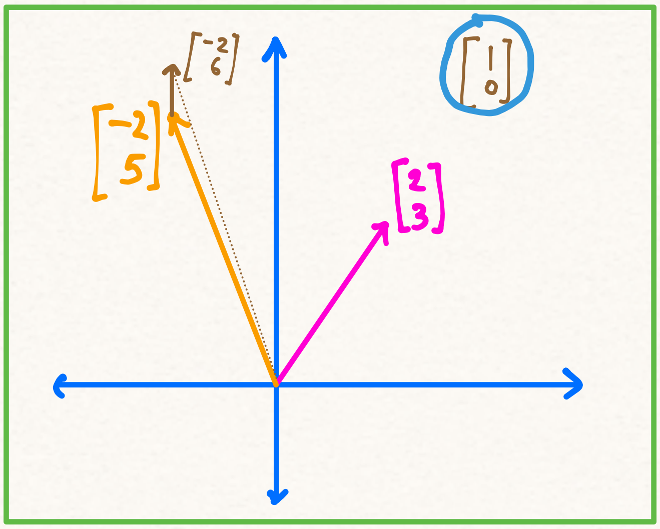


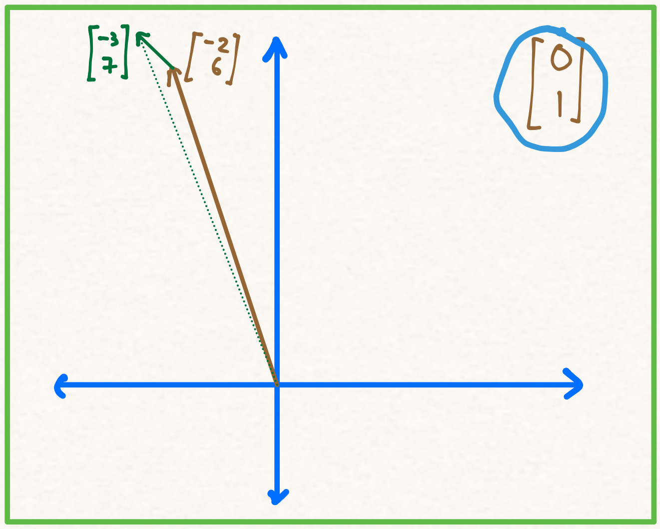




$\begin{bmatrix}-2\\5\end{bmatrix}$ instead of $\begin{bmatrix}5\\5\end{bmatrix}$







$\begin{bmatrix}1\\0\end{bmatrix} \longrightarrow \begin{bmatrix}0\\1\end{bmatrix}$

$\begin{bmatrix}0\\1\end{bmatrix} \longrightarrow \begin{bmatrix}-1\\1\end{bmatrix}$

$$c_1\begin{bmatrix}1\\0\end{bmatrix} + c_2\begin{bmatrix}0\\1\end{bmatrix} = \begin{bmatrix}x\\y\end{bmatrix}$$

$$c_1\begin{bmatrix}0\\1\end{bmatrix} + c_2\begin{bmatrix}-1\\1\end{bmatrix} = \begin{bmatrix}x\\y\end{bmatrix}$$ known

unknown

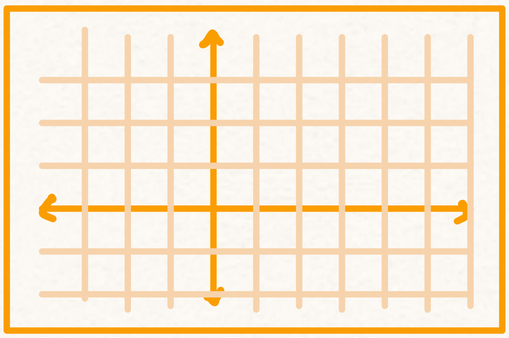
2 equ. 2 variables

$$\boxed{\begin{bmatrix} 0 & -1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix}}$$

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 1 \end{bmatrix}^{-1}\begin{bmatrix} x \\ y \end{bmatrix}$$

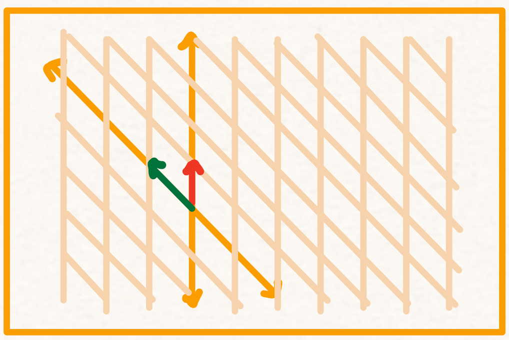
$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$$

Ojas' view



Robot's view

Robot's view from ojas' view

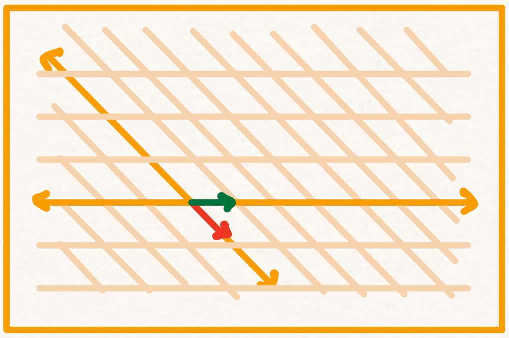


$$\begin{bmatrix} 0 & -1 \\ 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 \\ 1 & 1 \end{bmatrix}\underbrace{\begin{bmatrix} x \\ y \end{bmatrix}}_{\text{Robot's view}} = \underbrace{\begin{bmatrix} w \\ z \end{bmatrix}}_{\text{ojas' view}}$$

But, ojas' grid to Robot's grid

Ojas' view from Robot's view



$$\begin{bmatrix} 1 & 1 \\ -1 & 0 \end{bmatrix}$$ used for directing robot

$$\begin{bmatrix} 1 & 1 \\ -1 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} x \\ y \end{bmatrix}}_{\text{ojas' view}} = \underbrace{\begin{bmatrix} w \\ z \end{bmatrix}}_{\text{Robot's view}}$$ But, Robot's grid to ojas' grid

Robot moving $= \boxed{\begin{bmatrix} b_1 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}}$

$$\Downarrow$$

$$\begin{bmatrix} b_1 & b_2 \end{bmatrix} \begin{bmatrix} b_1 & b_2 \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \end{bmatrix} = \boxed{\begin{bmatrix} x \\ y \end{bmatrix}}$$

Room Shift

90° rotation to left $\Rightarrow \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = R$

ojas' view rotated by 90°

How will this look to Robot?

from whose view?

$$\begin{bmatrix} b_1 & b_2 \end{bmatrix} \begin{bmatrix} b_1 & b_2 \end{bmatrix}^{-1} R^{-1} \begin{bmatrix} x \\ y \end{bmatrix} = R^{-1} \underbrace{\begin{bmatrix} x \\ y \end{bmatrix}}_{\text{ojas' view}}$$

What does 90° rotation in Robot's view look like?

$$\begin{bmatrix} b_1 & b_2 \end{bmatrix} \underbrace{\begin{bmatrix} x \\ y \end{bmatrix}}_{\text{vector in Robot's view}} \longrightarrow \text{ojas' view}$$

$$R \begin{bmatrix} b_1 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \longrightarrow$$ Transformed vector in ojas' view

$$\underbrace{\begin{bmatrix} b_1 & b_2 \end{bmatrix}^{-1} R \begin{bmatrix} b_1 & b_2 \end{bmatrix}}_{\Downarrow \text{ R in robot's view}} \begin{bmatrix} x \\ y \end{bmatrix} \longrightarrow$$ Transformed vector in Robot's view as seen by ojas

$\Rightarrow$ **Any transformation T in ojas' view is $B^{-1} . T . B$ in Robot's view**

B $\rightarrow$ Robot's basis vector matrix from ojas' view

Now she can manage with:-

→ Malicious co-worker (basis change)

→ Room change (space transformation)

Observation:

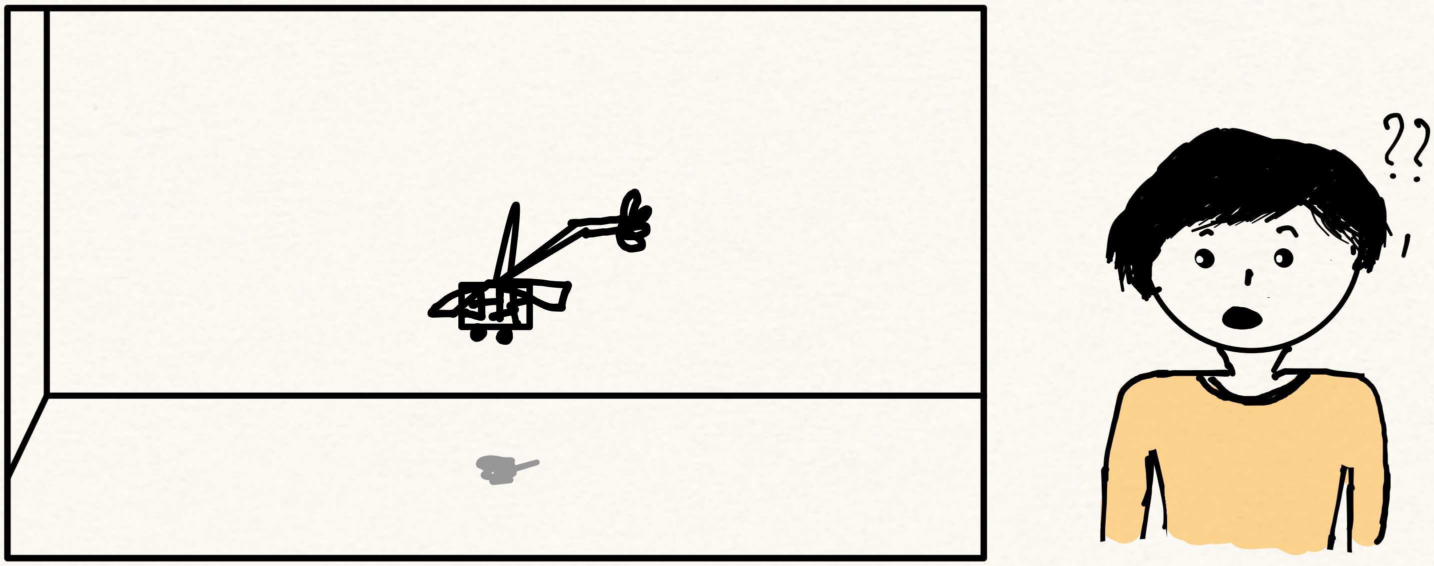
$$\begin{bmatrix} b_1 & b_2 \end{bmatrix}^{-1} \underbrace{\begin{bmatrix} x \\ y \end{bmatrix}}_{\text{eigen vectors}} = \underbrace{c}_{\text{eigen value}} \begin{bmatrix} x \\ y \end{bmatrix}$$

common Robot's and her view

What if these $\begin{bmatrix} x \\ y \end{bmatrix}$ were Rickie's basis?

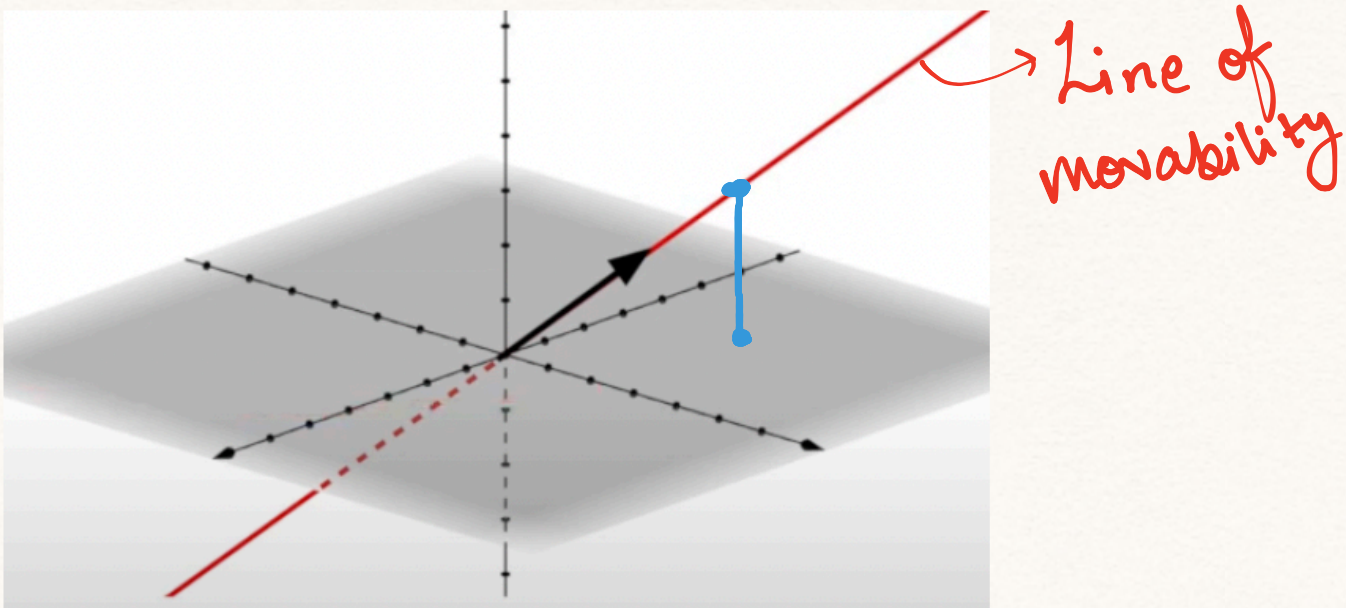
$$\text{Rickie's Matrix} = \underbrace{\begin{bmatrix} e_1 & e_2 \end{bmatrix}^{-1}}_{\text{eigen basis}} \cdot \underbrace{\begin{bmatrix} c_1 & 0 \\ 0 & c_2 \end{bmatrix}}_{\text{just scaling for eigen basis' view}} \cdot \underbrace{\begin{bmatrix} e_1 & e_2 \end{bmatrix}}$$

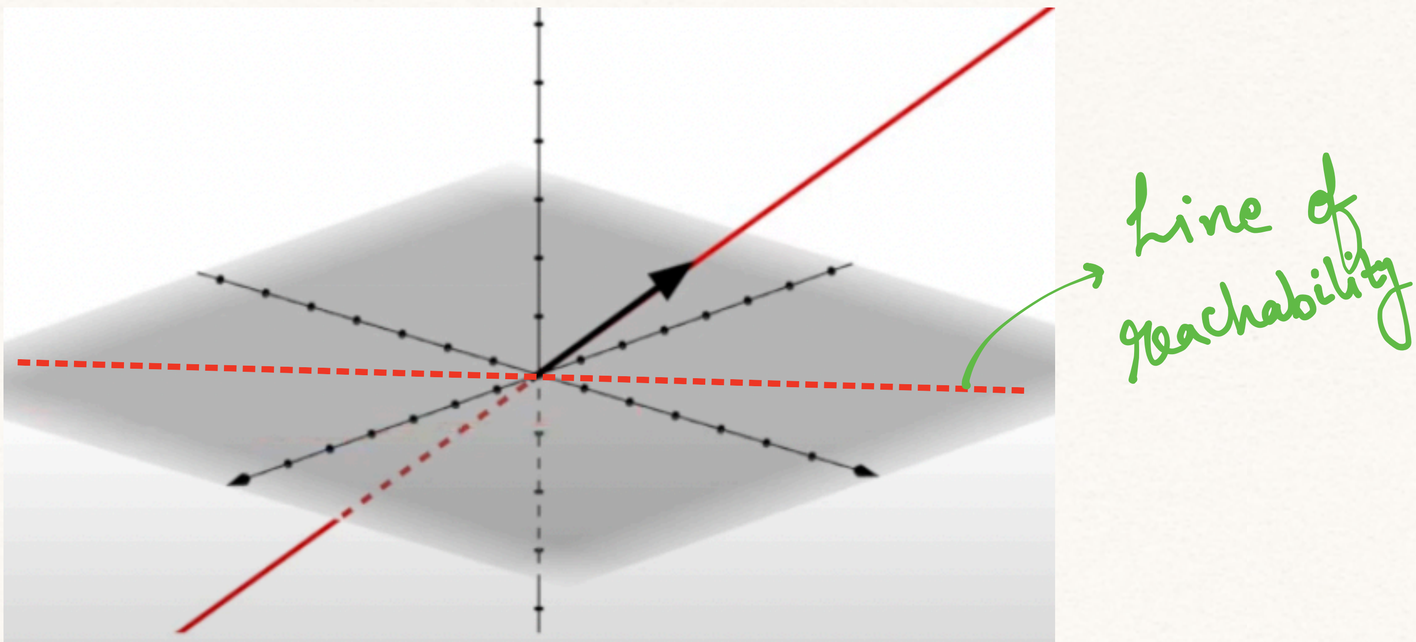
$$= \begin{bmatrix} b_1 & b_2 \end{bmatrix}$$

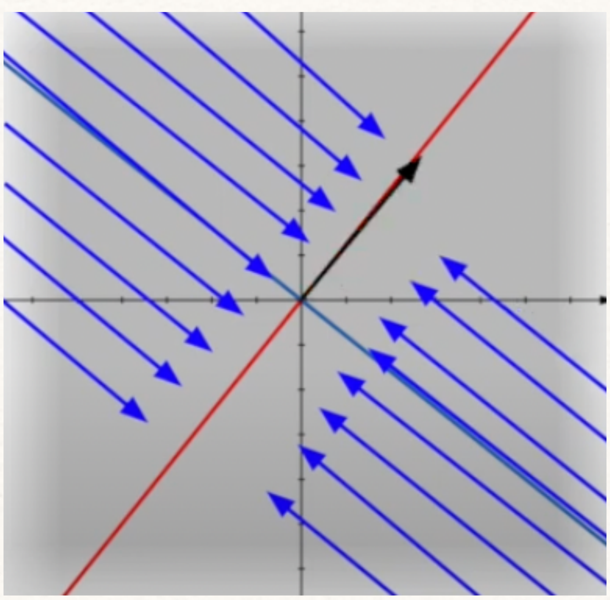

??

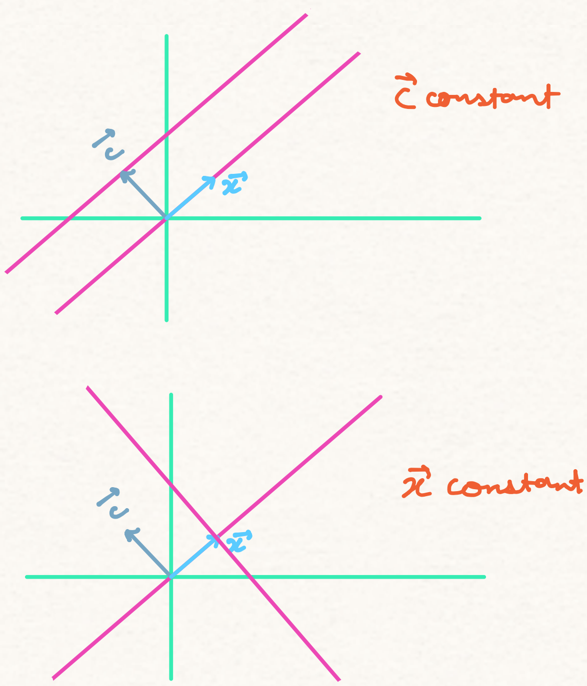

Basis?
Inverse?
Transformations?
eigenvectors?

illustrations taken from https: / / youtu. be / yfj8uMwAgrI

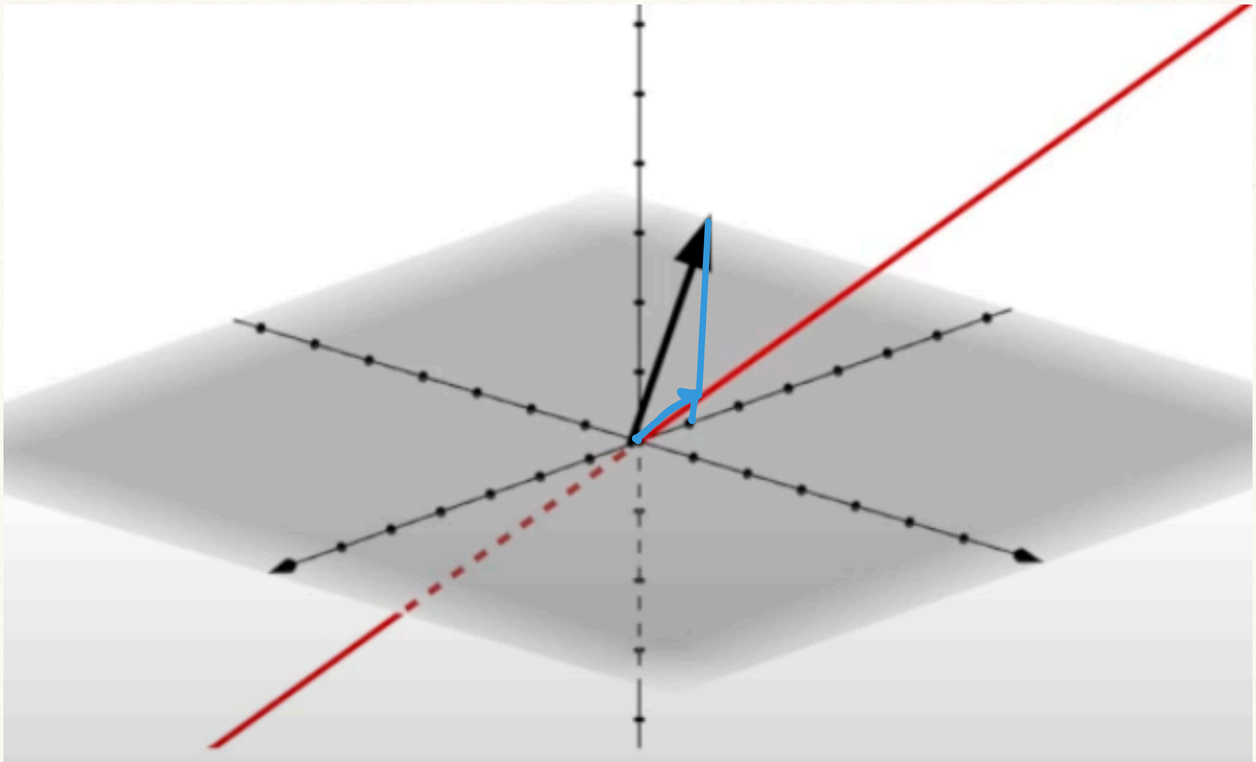


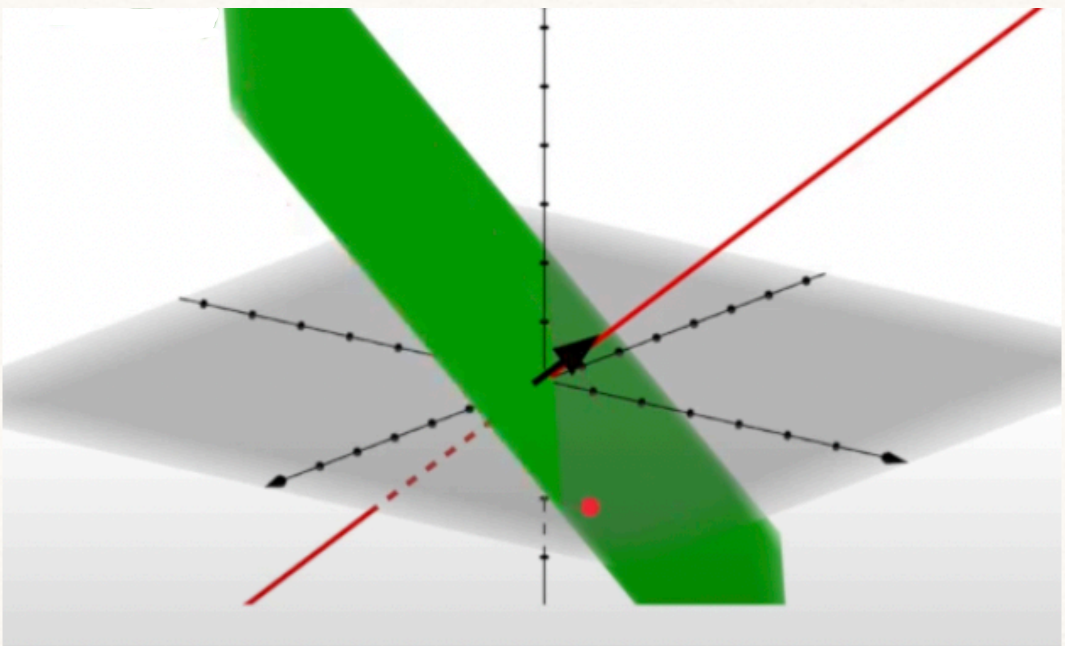






find inverse



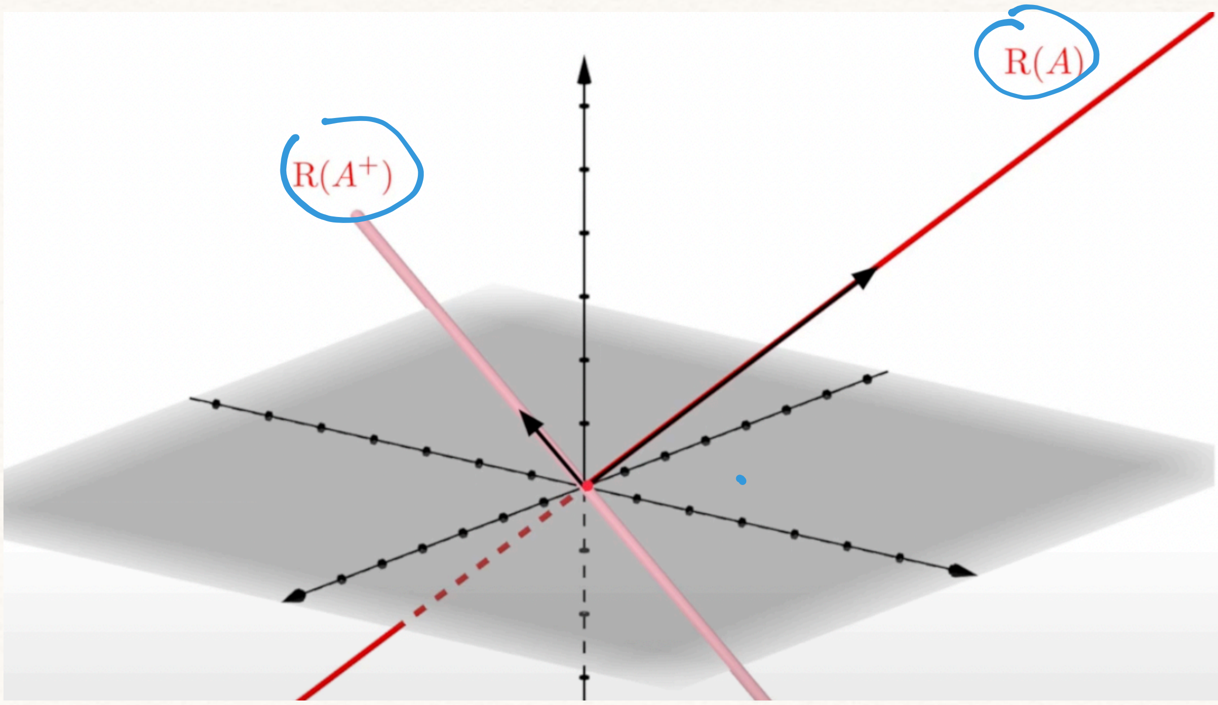


$A \longrightarrow$ basis vector matrix of Rickie

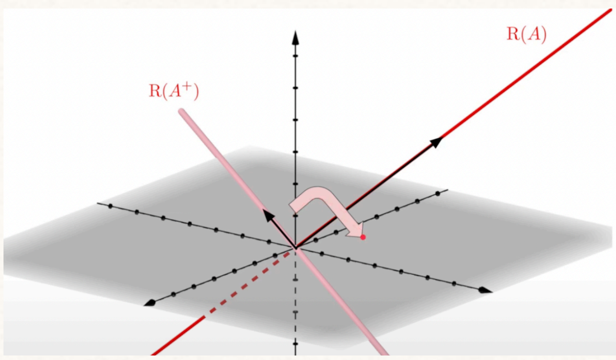
$A^+ \longrightarrow$ pseudo inverse of A

$R(A) \rightarrow$ Line of movability

$R(A^+) \rightarrow$ Range of A^+



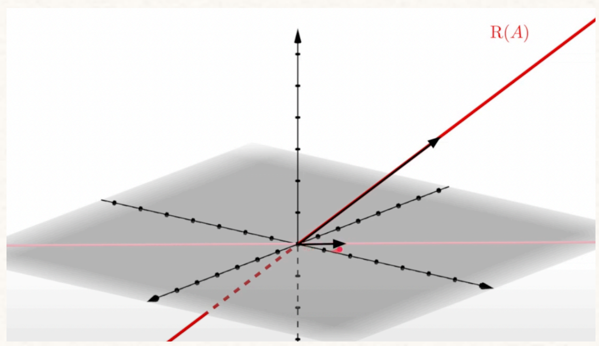
$A^+ \rightarrow A$

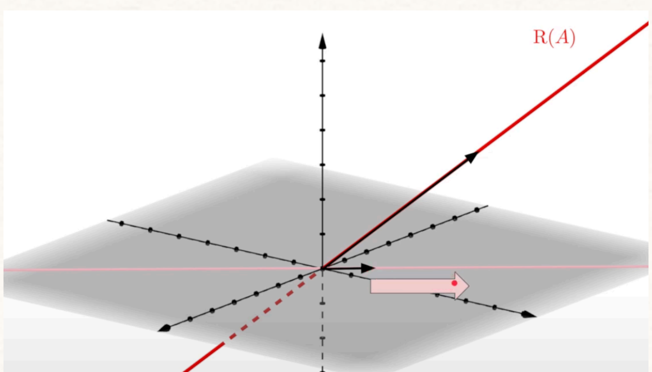


Rotating

$V^T_{2\times2} \rightarrow$ orthogonal

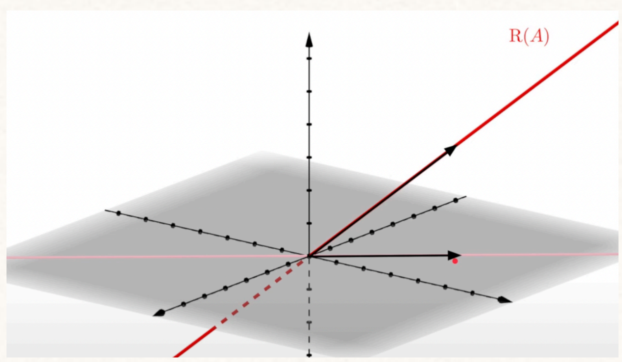
$V^{-1} = V^T$

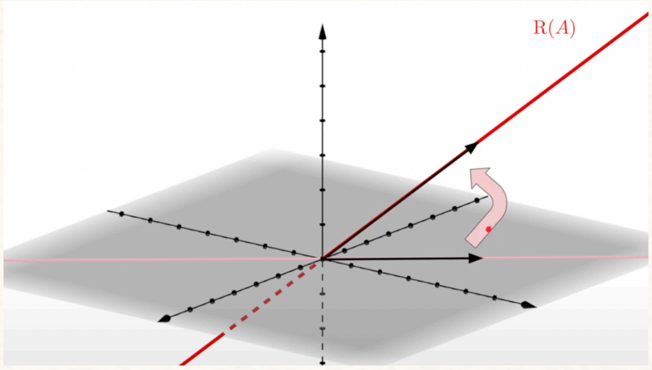




Scaling

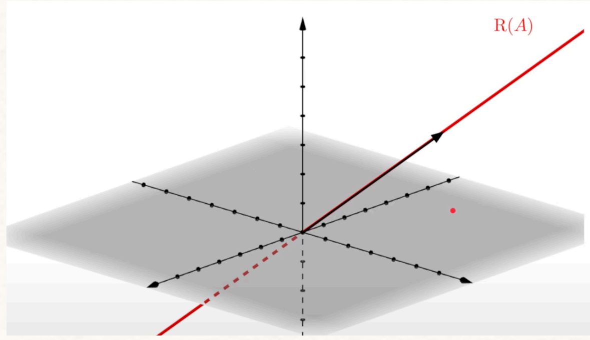
$$\Sigma_{3\times2} \rightarrow \begin{bmatrix} \sigma & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$





Rotating

$U_{3\times 3} \rightarrow$ orthogonal



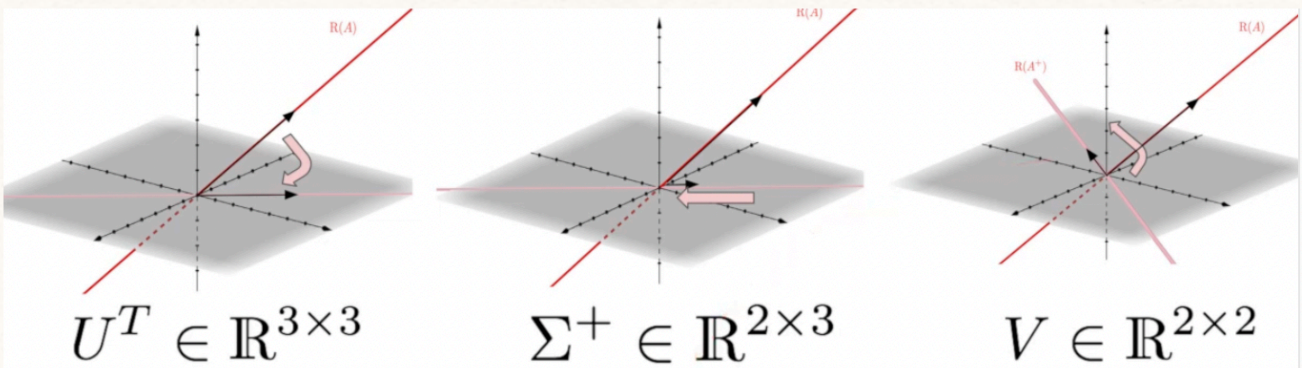
$$\boxed{A = U \Sigma V^T}$$

Singular Value Decomposition of A

$\downarrow$

σ

pseudo Inverse (A^+)

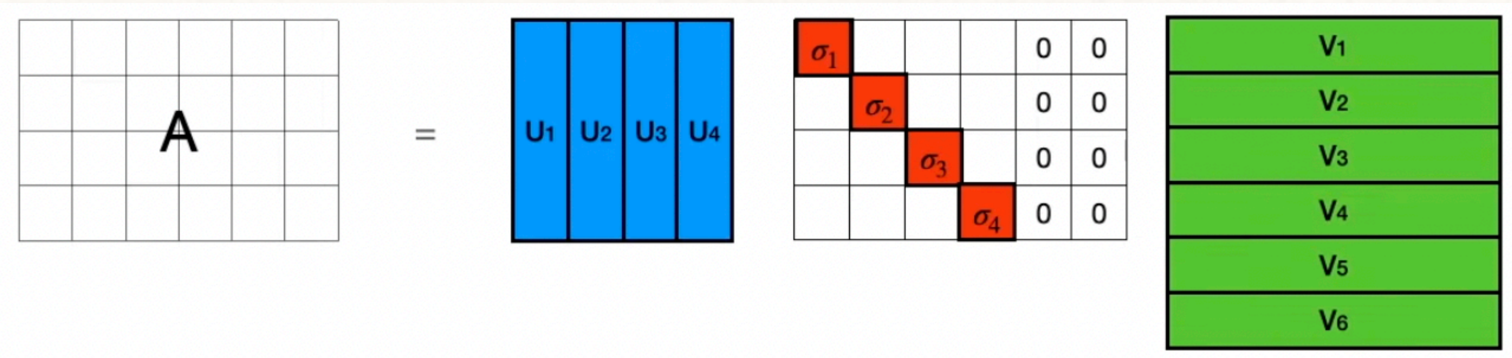


$$\boxed{A^+ = V \Sigma^+ U^T}$$

$$\Sigma^+ = \begin{bmatrix} 1/\sigma & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- → Image compression
- → video
- → Genomics
- → Movie recom.
- → PageRank
- → fluid flow
- → Any Kind of prediction systems
- → Statistics (PCA)
- → Robotics

illustrations taken from https://youtu.be/DG7YTlGnCEo



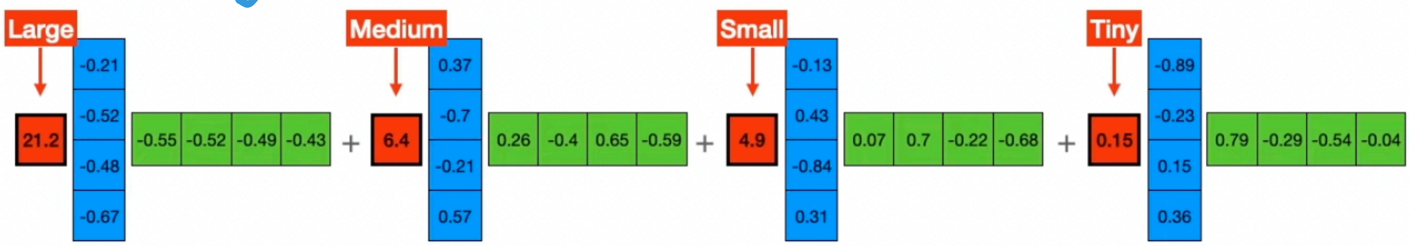
3	1	4	1
5	9	2	6
5	3	5	8
9	7	9	3

=

-0.21	0.37	-0.13	-0.89
-0.52	-0.7	0.43	-0.23
-0.48	-0.21	-0.84	0.15
-0.67	0.57	0.31	0.36

21.2			
	6.4		
		4.9	
			0.15

-0.55	-0.52	-0.49	-0.43
0.26	-0.4	0.65	-0.59
0.07	0.7	-0.22	-0.68
0.79	-0.29	-0.54	-0.04



2.51	2.37	2.22	1.97
6.07	5,72	5.37	4.77
5.63	5,31	4.99	4.43
7.88	7.43	6.98	6.19

3.15	1.41	3.79	0.56
4.87	7.53	2.44	7.43
5.28	5.85	4.12	5.22
8.85	5.96	9.36	4.03

3.1	0.96	3.93	0.99
5.03	8.99	1.98	6
4.98	3.01	5.01	8
8.96	7.02	9.03	3

3	1	4	1
5	9	2	6
5	3	5	8
9	7	9	3

GitHub

www.github.com/luisguiserrano/singular_value_decomposition

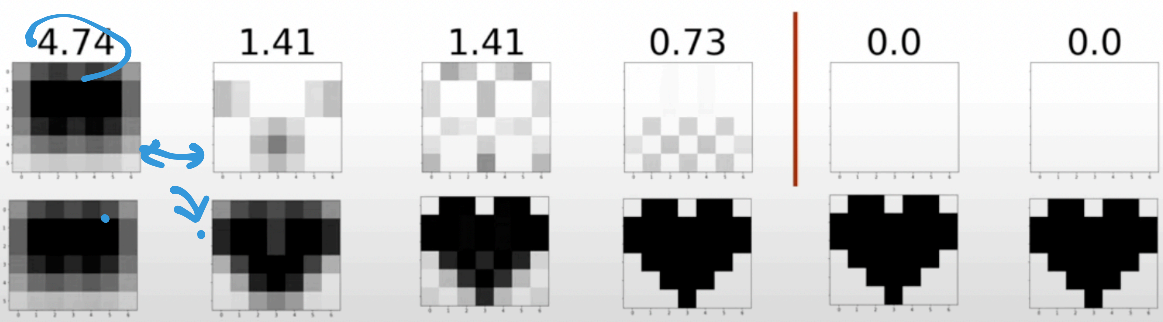
0	1	1	0	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1	1	0	0
0	0	0	1	0	0	0

F

```
[[-0.36 -0.   -0.73 -0.05  0.56  0.13]
 [-0.54  0.35  0.27 -0.08 -0.16  0.69]
 [-0.54  0.35  0.27 -0.08  0.16 -0.69]
 [-0.45 -0.35 -0.27  0.52 -0.56 -0.13]
 [-0.28 -0.71  0.18 -0.62 -0.   -0.  ]
 [-0.08 -0.35  0.46  0.57  0.56  0.13]]
```

```
[[4.74 0.   0.   0.   0.   0.   0.  ]
 [0.   1.41 0.   0.   0.   0.   0.  ]
 [0.   0.   1.41 0.   0.   0.   0.  ]
 [0.   0.   0.   0.73 0.   0.   0.  ]
 [0.   0.   0.   0.   0.   0.   0.  ]
 [0.   0.   0.   0.   0.   0.   0.  ]]
```

```
[[-0.23 -0.4  -0.46 -0.4  -0.46 -0.4  -0.23]
 [ 0.5   0.25 -0.25 -0.5  -0.25  0.25  0.5 ]
 [ 0.39 -0.32 -0.19  0.65 -0.19 -0.32  0.39]
 [-0.22  0.42 -0.44  0.42 -0.44  0.42 -0.22]
 [ 0.56 -0.43  0.03  0.   -0.03  0.43 -0.56]
 [-0.42 -0.55 -0.16  0.    0.16  0.55  0.42]
 [-0.12 -0.11  0.69 -0.   -0.69  0.11  0.12]]
```



Thanks to:

- Peers at coder's high and Sudarshan sir
- 3Blue1Brown LA playlist
- Prof. Gilbert Strang's LA Lectures
- Ben Newman : Video on Transpose
- Steve Brunton : SVD lectures
- Serrano.Academy : video on SVD

Email : 21f2000232@student.onlinedegree.iitm.ac.in