Section - 2 - Intro to Linear Algebra

② Introduction

Linear Algebra

↓

branch of maths → focuses on the study of vectors
(vector spaces)
(also called
Linear Spaces)

provides a framework
↳ in order to understand the prop and operations
of the mathematical objects
↳ can be represented using
matrices and vectors

The above are
① foundational
Concepts
which you need to
understand

when you → are studying
ML
DL
NLP
Images (CV)

↓ some of the
core concepts in
Linear Algebra

① Scalars

② Vectors

③ Matrices

④ Mathematical
operations (matrices)

⑤ Linear Transformation
(This will be used in Principal comp Analysis)

⑥ Eigen Values

⑦ Eigen Vectors

Applications of Linear Algebra

① Data Representation and Manipulation

Say I have a

DATASET → (I have to create a MODEL which will be able to predict)

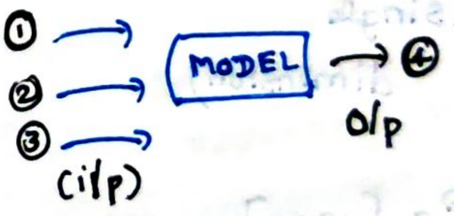
↓

Say I have a House Price Dataset (with the below columns)

① Area ② No. of Rooms ③ Location ④ Price → O/p feature (dependent one)

① ② ③ (I/p features Independent features basically)

My Model



Suppose Say I have values in my table like this

[1200 2 Bangalore 45 lakhs INR] → This will be converted to a vector (for computer to understand this)

→ By converting them to vectors we can quantify the relationship

Now converting this to a vector I will be able to know

x, y (I have 2 entities/ columns)

when x is (↑) y is (↑)/(↓) Similarly other relationships

Note

while training the model with i/p data we convert them into vectors and input it

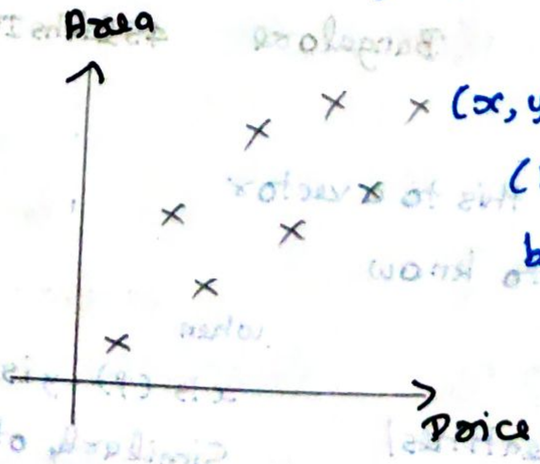
Linear Algebra

↳ provides a tool in order to manipulate the data in the form of vectors for easy understanding of computers

Vectors can also be represented in the form of dimensions

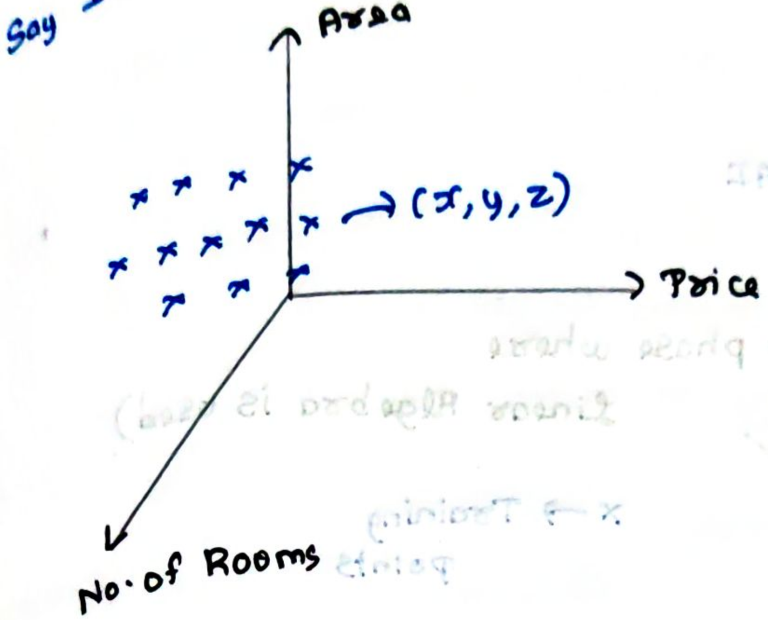
1D vector (single dimension)	2D vector (2-Dim)	Likewise we can also have (3D/4D...)
$\vec{V} = [1200]$	$\vec{V} = [1200\ 2]$	

↓ If I want to plot this in a graph



$(x, y) \Rightarrow$ (1200, 45 lakhs) becomes a 2D vector

Say I want to have a 3D Vector



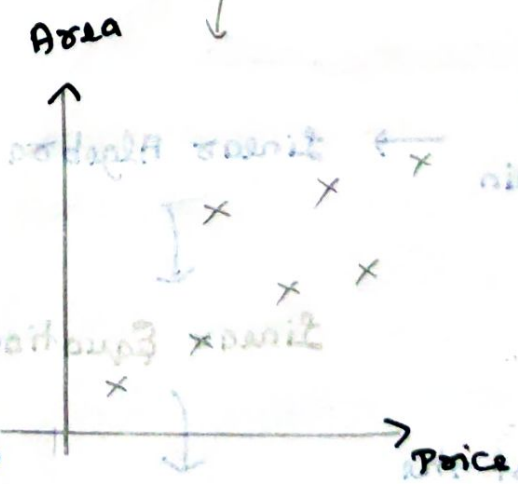
Q why we use Linear Algebra for ML?

② works with high Dimension Data

Eg Say we have a data with 500 dimensions

⇓ with the help of Dimensionality reduction (PCA)

we can convert it into a 2-D Vector



from which we can find the relationship b/w x & y

↳ x↑ y↑ x↑ y↓

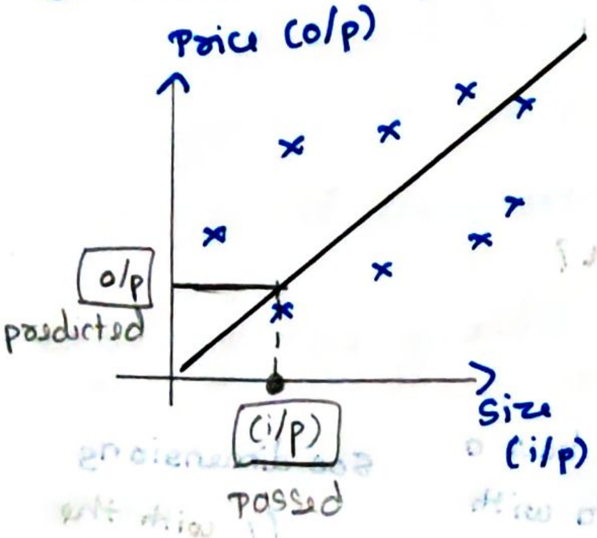
x↓ y↓ x↓ y↑

Another Case why Linear Algebra is used widely

↓

② Application in ML and AI

2. ① Model Training (imp phase where Linear Algebra is used)



x → Training points

(This is a regression problem

↓

So with the help of training points we draw a best fit line after which if so any points of size are given we will be able to predict them)

So,

Model Train → Linear Algebra → In order to train the model we use a lot of Matrix multiplication

↓ Matrix operations basically

Linear Algebra

↓

Linear Equation

↓

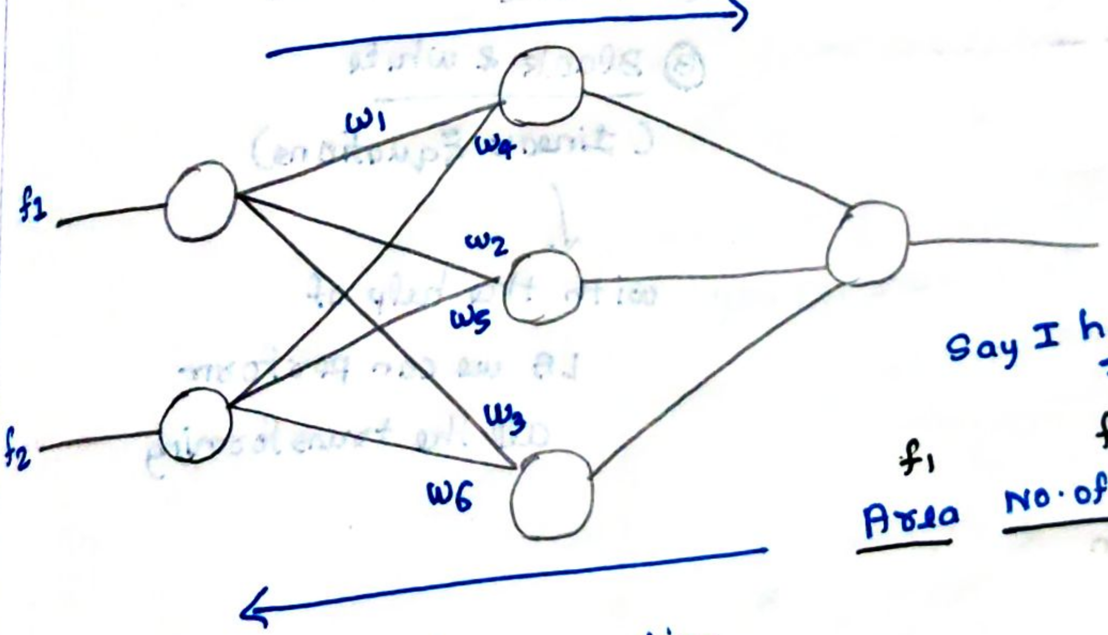
one of the most common LE → Equation of a st. line

$$\boxed{ax+by+c=0} \Rightarrow \boxed{y=mx+c}$$

② Dimensionality Reduction ⇒ PCA → Linear Algebra → Eigen Values and Eigen Vectors

Reduce from higher Dimension → Lower Dimension

③ Neural N/W : Forward propagation and Backward Propagation (we use chain rule of derivative)



Say I have 2 i/p's

f1	f2	o/p
Area	No. of Rooms	Price

for the forward propagation (for the 1st layer)

$\boxed{2 \times 3}$ (It is a Matrix multiplication)

$$\begin{bmatrix} f_1 \\ f_2 \end{bmatrix} \begin{bmatrix} w_1 & w_2 & w_3 \\ w_4 & w_5 & w_6 \end{bmatrix}$$ ⇒ Matrix multiplication

we use Tensorflow
↓
convert data into tensors → progs in GPU → has Cores → $||^{ele}$ execution

we train a NNet with the help of GPU's

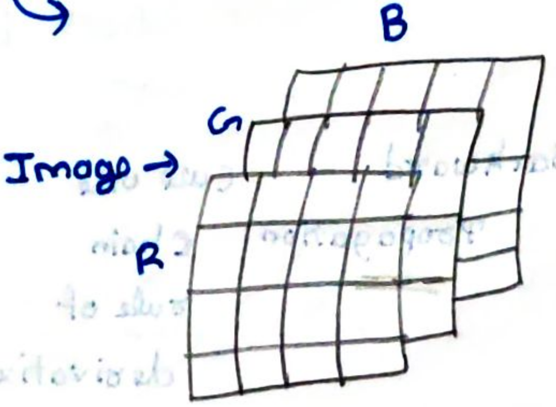
③ Computer Graphics

Suppose say

I have an

RGB image

↳



we can perform

① Scaling

② Rotate

③ Black & white

(Linear Equations)

↓

with the help of LA we can perform all the transforming

④ Optimisation

① Solving Equations

↳ Say I have a Linear Equation

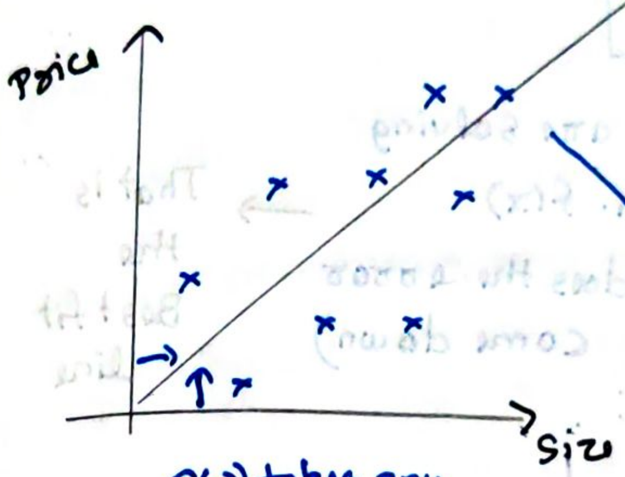
$$\boxed{y=mx+c} \longrightarrow$$ Speaking about Regression (algo)

slope / coeff

intercept

we need to find the right kind of m and c value

Eg → If I have a dataset



Aim is to draw a best fit line

x → Data points

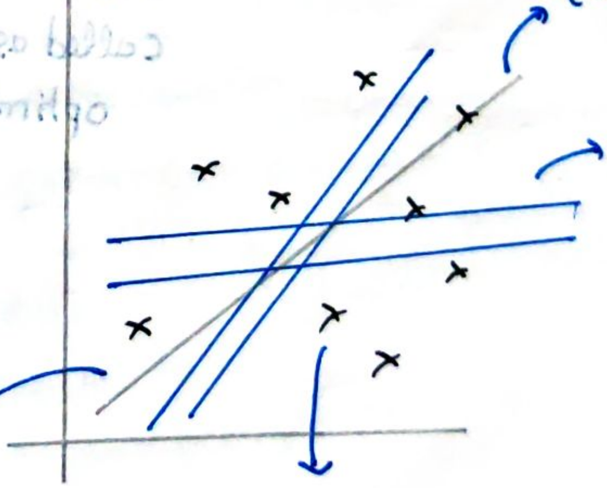
↑(→) takes any size as i/p

→ (→) should predict the prize of the house

if you want to really recreate this line you must have the proper $\boxed{\text{m and c value}}$

you can draw a lot of lines

(but why this?)



you can draw many lines

how to select this best fit line? (where you come up with ① Right Slope ② Intercept)

we need to solve the fn. $\boxed{y = mx + c}$

why? The diff b/w the data points and the best fit line (Summation of diff/errors w.r.t data points → minimal)

We take

$\boxed{y=mx+c}$ as a function $\boxed{f(x)}$

(when we are solving this fn. f(x) when does the error come down) → That is the Best fit line

Our aim

f(x) → Maximize the fn.

Minimize the error

(The above process is called as Optimization)

a concept of gradient descent will be applied here

This is basically called as an optimizer

Side Doubts on Revision

(Introduction video)

Regression

↳ Supervised ML Alg → goal is to predict a continuous (numerical) value based on i/p data

Also
you are trying to find a relationship b/w i/p variable (feature) target variable (target)

(why Supervised)

Set of Numerical independent values based on which → you try to find the dependent values

Size (house)
No. of bedrooms
Location
→ Predict the price of house

Cts numerical values are no's that can take any values within a given range (not restricted to any specific/separate values)

Eg
height → 170.5 cm, 180.2 cm

weight ↳ 65.3 kg, 70.8 kg

(What is the diff b/w Discrete and continuous values)

Specific and countable (represent items which can be counted indiv)

Counting

Eg → No. of Items
23

No. of apples
10

measurement
① length
② weight
③ temp
④ time

Can take any values within the given range

can represent a measurement (more precise)

rope → 23m long (could be any thing including 23.1/23.2 . . .)