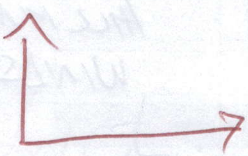
Vectors – basis application

- so we have seen that we not only need to use natural number ~~of~~ basis vectors of $\underline{1}$, $\Rightarrow (1,0)$ or $(0,1)$

$\Rightarrow$ which is natural basis

eg. $e_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$



$e_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

$\therefore$ we can have different basis vectors, that redefine how we move about in space

~~How~~ <u>Now</u>, we going to define what we mean by a basis (of vector space) <u>and</u> by term linear independence

$\Rightarrow$ which will allow us to understand how many dimension our vector space actually has.

Lets define what is a $\boxed{\text{basis:}}$

- (1) Not linear combination of eachother
  ie. they linearly independent.
- (2) Span the space
- (3) The space is then n-dimensional.

Or in other words,...

its a set of n vectors that are
not linear combinations of each other.
they linearly independent
and they span the space they describe
⇒ that means the space is then n-dimensional.

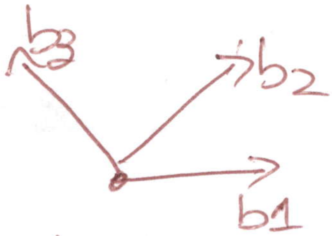
hence

Lineanly independent?

eg b_3 it will be lineanly dependent on basis vectors b_1 and b_2, if I can take some combination of them (a_1 or a_2), and find b_3

$$b_3 = a_1 b_1 + a_2 b_2$$

∴ b_3 is lineanly dependent on b_1 and b_2 (it not independent)



B_2 is found on some combination of b_1 or b_3.

∴ it lies in same plane as b_1/b_2

But if b_3 lies outside plane, I cannot use b_1 or b_2

then b_3 would be lineanly independent

∴ B $b_3 \neq a_1 b_1 + a_2 b_2$

∴ then we will have 3 basis vectors

⇒ which will then define a 3 dimensional vector space

and I can get anywhere in 3D space with them.

(or 4 ⇒ 4 Dim, etc, etc) — as many Dimension

But notice what my Basis vectors DONT Have to be:

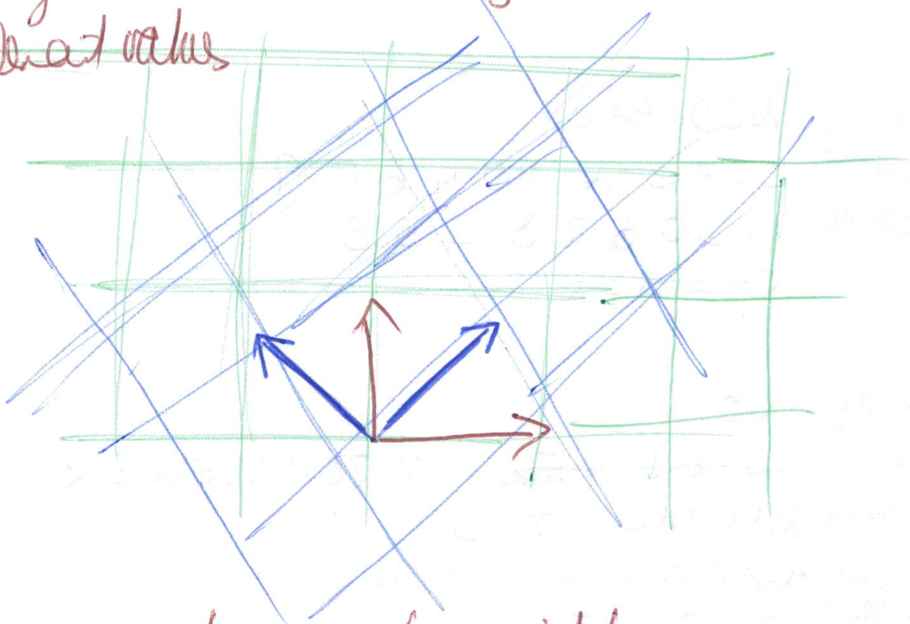
⇒ Dont have to be unit vectors
i.e. vectors of length 1

⇒ Dont have to be orthogonal ⊥ (90°)
(90° to each other)
(thought will be easier if they are....)
↳ so if possible construct a
orthonormal basis vector set
↳ 90°
↳ unit length } IDeal

Lets see what happens when we map from one basis to another....

Basis
- original
- New

⇒ Old project onto the new grid
- with different values



But projection keep the grid been evenly spaced

∴ therefore any mapping we do, from one set (in one coordinate system) to another (in another coordinate system)

Keeps vector space regularly spaced.

- where original rules of multiplication with scalar still works

(it does not warp space

⇒ this is the linearity bit of linearity algebra)

∴ may be stretched or inverted,

But everything remains evenly spaced.

∴ linear combinations still work.

6

But if new Basis vectors are **NOT** orthogonal, then ~~d~~ to do change from one basis to another, we won't be able to use the dot product any more, **we** will have to use Matrices **instead**