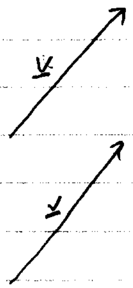
Vectors

notation: vectors are lowercase underlined



a vector

the same vector

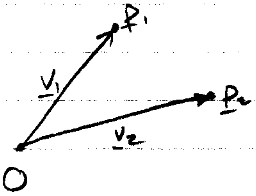
represents 2 things : length and direction
$\|\underline{v}\|$ (length) $\hat{\underline{v}}$ (unit vector) (direction)

($\nexists$ direction if length = 0)

*special vector $\underline{0}$ has zero length, no direction

a vector represents an offset from one location in $\mathbb{R}^n$ to another

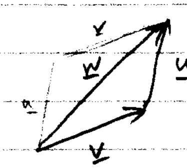
a vector only represents a position if you first choose an origin:



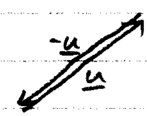
when the origin is understood we'll ignore the distinction for convenience and label the vector $\underline{p}$.

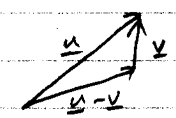
vector space operations

addition $\underline{w} = \underline{v} + \underline{u}$
"how far have I gone if I first walk $\underline{v}$, then $\underline{u}$?"
- commutative $(\underline{v} + \underline{u} = \underline{u} + \underline{v})$ — from parallelogram



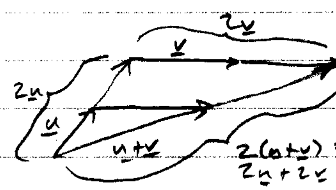
negation $\underline{u} + (-\underline{u}) = 0$
- notation $\underline{u} - \underline{v} \equiv \underline{u} + (-\underline{v})$





scaling (scalar multiplication) $\alpha \underline{u}$ ← changes length preserves direction
- distributive $\alpha(\underline{u} + \underline{v}) = \alpha\underline{u} + \alpha\underline{v}$

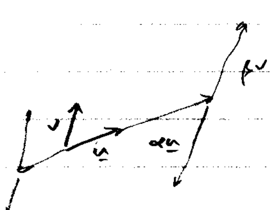
scalar – always $\in \mathbb{R}$ for us.



↑ whole diagram magnified by α

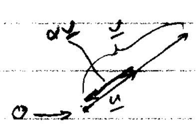
Coordinates & bases

def: 2 vectors $\underline{u}, \underline{v}$ are linearly dependent if $\alpha\underline{u} + \beta\underline{v} = 0$ for some α, β.



← not going to get back to zero

can get back to zero →

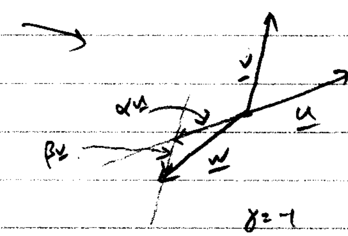


Coordinates & bases, cont.

for 2 vecs, linearly dependent $\Leftrightarrow$ parallel.

def: 3 vectors $\underline{u}, \underline{v}, \underline{w}$ are linearly dependent if $\alpha\underline{u} + \beta\underline{v} + \gamma\underline{w} = 0$ for some α, β, γ.

- any 3 vectors in the plane are dependent →
- for 3 vectors linearly dependent $\Leftrightarrow$ coplanar



useful property: For 2 lin. indep. vectors $\underline{u}, \underline{v}$ in 2D, you can write any vector as a linear combination of $\underline{u}$ and $\underline{v}$:

$$\underline{w} = \alpha\underline{u} + \beta\underline{v}$$

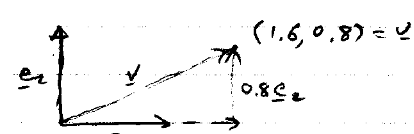
(really this is the same stmt as just above)

- same property holds for 3 vectors in 3D.

def: span of vectors $\underline{u}, \underline{v}$ $\quad \text{span}\{\underline{u}, \underline{v}\} = \{\alpha\underline{u} + \beta\underline{v} \mid \alpha, \beta \in \mathbb{R}\}$.
- that is, all the vectors you can get to using $\underline{u}$ and $\underline{v}$.
- property above is, for lin indep $\underline{u}, \underline{v} \in \mathbb{R}^2$, $\text{span}\{\underline{u}\ \underline{v}\} = \mathbb{R}^2$
- trivial generalization to 3D . . .

Q: How do we represent vectors in code?
- can't really store a vector in memory – only numbers.

→ So pick some basis vectors and store the coefficients!
I'll call these vectors the "canonical basis" and name them $\underline{e}_1, \underline{e}_2$, (and $\underline{e}_3$).
We do not store $\underline{e}_1, \underline{e}_2, \underline{e}_3$ – we just agree upon them and store the coefficients.



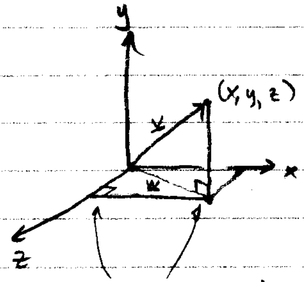
Note I drew those vectors perpendicular →

def: orthogonal = perpendicular
orthonormal = orthogonal and length = 1

The canonical basis will always be orthonormal.
- this is a cartesian coordinate system.
- one nice property: can use Pythagorean thm to compute lengths:

$$\|(x, y)\| = \sqrt{x^2 + y^2} \qquad \|(x, y, z)\| = \sqrt{x^2 + y^2 + z^2}$$

Coordinates & bases cont.



use pythag. twice $\|w\|^2 = x^2 + z^2$

$\|v\|^2 = \|w\|^2 + y^2$

Notation $(x, y) = \begin{bmatrix} x \\ y \end{bmatrix} = [x \ y]^T$ often $\underline{u} = (u_1, u_2) = u_1 e_1 + u_2 e_2$

Vector products

Dot product of two vectors: (also inner prod. or scalar prod)

$a = \underline{u} \cdot \underline{v} = \langle \underline{u}, \underline{v} \rangle$

scalar vectors

· has some properties:

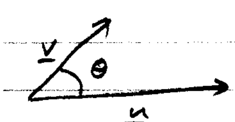
$\underline{u} \cdot \underline{v} = \underline{v} \cdot \underline{u}$ ← commutative

$u \cdot (\underline{v} + \underline{w}) = \underline{u} \cdot \underline{v} + \underline{v} \cdot \underline{w}$ ← distributes over addition

$(\alpha \underline{u}) \cdot \underline{v} = \underline{u} \cdot (\alpha \underline{v}) = \alpha (u \cdot v)$ ← scalars factor out

interpretations

$\underline{u} \cdot \underline{v} = \|u\| \, \|v\| \cos\theta$



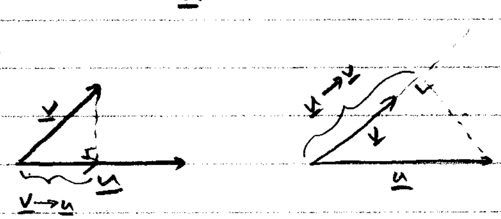
special case $\underline{u} \cdot \underline{u} = \|u\|^2$

special case if $\underline{u} \perp \underline{v}$ $\underline{u} \cdot \underline{v} = 0$ $(\cos\theta = 0)$

$\underline{u} \cdot \underline{v} = \|v\| (\underline{u} \to \underline{v}) = \|u\| (\underline{v} \to \underline{u})$

special case if $\|a\| = 1$

$u \cdot a = \underline{u} \to \underline{a}$



$\underline{a} \to \underline{b}$ is a projected onto b.

Vector products cont.

So far dot prod. is in terms of abstract vectors (no coordinates).

How to compute a dot product? We store u_1, u_2, v_1, v_2

$$\underline{u} \cdot \underline{v} = (u_1 \underline{e}_1 + u_2 \underline{e}_2)(v_1 \underline{e}_1 + v_2 \underline{e}_2)$$
$$= u_1 v_1 \cancel{\underline{e}_1 \cdot \underline{e}_1}^{1} + u_1 v_2 \cancel{\underline{e}_1 \cdot \underline{e}_2}^{0} + u_2 v_1 \cancel{\underline{e}_2 \cdot \underline{e}_1}^{0} + u_2 v_2 \cancel{\underline{e}_2 \cdot \underline{e}_2}^{1}$$
$$= u_1 v_1 + u_2 v_2$$

In $\mathbb{R}^3$, $\underline{u} \cdot \underline{v} = u_1 v_1 + u_2 v_2 + u_3 v_3$.

Cross product of two vectors: (only for 3D!)

$$\underbrace{\underline{w}}_{\text{vector}} = \underbrace{\underline{u} \times \underline{v}}_{\text{vectors}}$$

- has some properties:

$\underline{u} \times \underline{v} = -\underline{v} \times \underline{u}$ ← anticommutative

$\underline{u} \times (\underline{v} + \underline{w}) = \underline{u} \times \underline{v} + \underline{u} \times \underline{w}$ ← distributes over +

$(\alpha \underline{u}) \times \underline{v} = \underline{u} \times (\alpha \underline{v}) = \alpha (\underline{u} \times \underline{v})$ ← scalars factor out

$\underline{u} \cdot (\underline{u} \times \underline{v}) = (\underline{u} \times \underline{v}) \cdot \underline{v} = 0$ ← orthog. to both

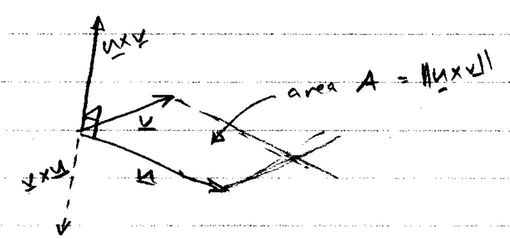
interpretations:

$\|\underline{u} \times \underline{v}\| = \|\underline{u}\| \, \|\underline{v}\| \sin\theta$

$\|\underline{u} \times \underline{v}\|$ is the area of the parallelogram spanned by $\underline{u}, \underline{v}$.

$\underline{u} \times \underline{v}$ is a kind of oriented area

special case: $\underline{u} \times \underline{u} = \underline{0}$



arbitrary choice (really a convention about ordering of coordinates):

$e_1 \times e_2 = e_3$; $e_2 \times e_3 = e_1$; $e_3 \times e_1 = e_2$ ← forward pairs

$e_2 \times e_1 = -e_3$, etc. ← backward pairs

arbitrary geometric convention — right hand rule (really discriminates for lefties!)

e_1 = thumb, e_2 = forefinger, e_3 = palmward.

Vector products cont.

So far $\times$ in terms of abstract vectors — what about coordinates?

$$
\begin{aligned}
\underline{u} \times \underline{v} &= (u_1 \underline{e}_1 + u_2 \underline{e}_2 + u_3 \underline{e}_3) \times (v_1 \underline{e}_1 + v_2 \underline{e}_2 + v_3 \underline{e}_3) \\
&= u_1 v_1 \cancel{e_1 \times e_1}^{\,0} + u_1 v_2 \cancel{(e_1 \times e_2)}\, e_3 + u_1 v_3 \cancel{(e_1 \times e_3)}(-e_2) \\
&\quad u_2 v_1 \cancel{(e_2 \times e_1)}(-e_3) + u_2 v_2 \cancel{e_2 \times e_2}^{\,0} + u_2 v_3 \cancel{(e_2 \times e_3)}(e_1) \\
&\quad u_3 v_1 \cancel{(e_3 \times e_1)}(e_2) + u_3 v_2 \cancel{(e_3 \times e_2)}(-e_1) + u_3 v_3 \cancel{e_3 \times e_3}^{\,0} \\
&= (u_2 v_3 - u_3 v_2) e_1 + (u_3 v_1 - u_1 v_3) e_2 + (u_1 v_2 - u_2 v_1) e_3 \\
&= (u_2 v_3 - u_3 v_2,\ u_3 v_1 - u_1 v_3,\ u_1 v_2 - u_2 v_1)
\end{aligned}
$$

note forward pairs are ⊕, backward pairs are ⊖, and each coordinate depends on the other two coordinates.
to me this is the best mnemonic aid — straight from the definition!

ONB's and Coordinate frames

An orthonormal basis (ONB) is a set of n mutually orthogonal unit length vectors ($n = 2$ for plane, 3 for 3-space)
(usual notation will be $\underline{u}, \underline{v}, \underline{w}$).
Convention: right-handed $\underline{u} \times \underline{v} = \underline{w}$ (same as e_1, e_2, e_3).

Useful property: coordinates of $\underline{x}$ wrt $(\underline{u}, \underline{v}, \underline{w})$ are $\underline{u} \cdot \underline{x}$, $\underline{v} \cdot \underline{x}$, $\underline{w} \cdot \underline{x}$:

$$
\begin{aligned}
(x_u \underline{u} + x_v \underline{v} + x_w \underline{w}) \cdot \underline{u} &= x_u \cancel{(\underline{u} \cdot \underline{u})}^{\,1} + x_v \cancel{(\underline{v} \cdot \underline{u})}^{\,0} + x_w \cancel{(\underline{w} \cdot \underline{u})}^{\,0} \\
&= x_u .
\end{aligned}
$$

This, in a nutshell, is why we care about ONBs: it is very easy to transform to and from canonical coordinates.

if we have the coordinates (x_u, x_v, x_w) of $\underline{x}$ in the ONB $(\underline{u}, \underline{v}, \underline{w})$ and we have the coordinates (u_1, u_2, u_3), (v_1, v_2, v_3), and (w_1, w_2, w_3) of $\underline{u}$, $\underline{v}$, and $\underline{w}$ in the canonical basis, then

$$
\left.
\begin{aligned}
x_1 &= x_u u_1 + x_v v_1 + x_w w_1 \\
x_2 &= x_u u_2 + x_v v_2 + x_w w_2 \\
x_3 &= x_u u_3 + x_v v_3 + x_w w_3
\end{aligned}
\right\}
\text{ just the def'n } \underline{x} = x_u \underline{u} + x_v \underline{v} + x_w \underline{w} \text{ expanded.}
$$

(This is the same for a non-orthonormal basis.)

If we have the coordinates (x_1, x_2, x_3) of $\underline{x}$ in the canonical basis then

$$x_u = \underline{x}\cdot\underline{u} = x_1u_1 + x_2u_2 + x_3u_3$$
$$x_v = \underline{x}\cdot\underline{v} = x_1v_1 + x_2v_2 + x_3v_3$$
$$x_w = \underline{x}\cdot\underline{w} = x_1w_1 + x_2w_2 + x_3w_3$$

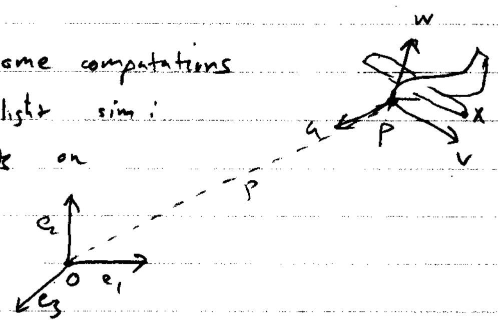
Note this (and the above) is a matrix multiplication — more on this in a couple of lectures...

(This depends on orthonormality.)

This ONB discussion has just been about vectors that represent offsets, not ones that represent points (or, we've been using the same origin).

Generally we want the freedom to change the origin also, so we associate an origin with the ONB to make a coordinate frame $(\underline{p}, \underline{u}, \underline{v}, \underline{w})$.

These are chosen to make some computations more convenient. E.g. flight sim: useful to represent points on the wings in a frame attached to the plane.



$$\underline{x} = \underline{p} + x_u\underline{u} + x_v\underline{v} + x_w\underline{w}$$
$$x_u = (\underline{x}-\underline{p})\cdot\underline{u}$$
$$x_v = (\underline{x}-\underline{p})\cdot\underline{v}$$
$$x_w = (\underline{x}-\underline{p})\cdot\underline{w}$$

Useful example: constructing ONB from one vector $\underline{a}$ (e.g. if we need to compute something that's symmetric about that vector)

We'll align $\underline{w}$ with $\underline{a}$, but it's supposed to be a unit vector so $\underline{w} = \frac{\underline{a}}{\|\underline{a}\|}$

Now if we can get our hands on a vector perpendicular to $\underline{w}$, the cross product will finish the job. The cross product solves this too: if $\underline{t}$ is not parallel to $\underline{w}$ then we can use

$$\underline{u} = \frac{\underline{t}\times\underline{w}}{\|\underline{t}\times\underline{w}\|}$$ ← that is, compute cross product and then normalize

now $\underline{u}\perp\underline{w}$. To complete the right-handed ONB:

$$\underline{v} = \underline{w}\times\underline{u}$$ ← note forward pair (convention for R.H.)

Note a hw prob is related closely to this construction.