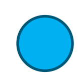
Vector and Affine Math I

Points in Euclidean Space $\mathbb{R}^n$

Location in space

Tuple of n **coordinates** x, y, z, etc

$$p = (p_x, p_y, p_z)$$

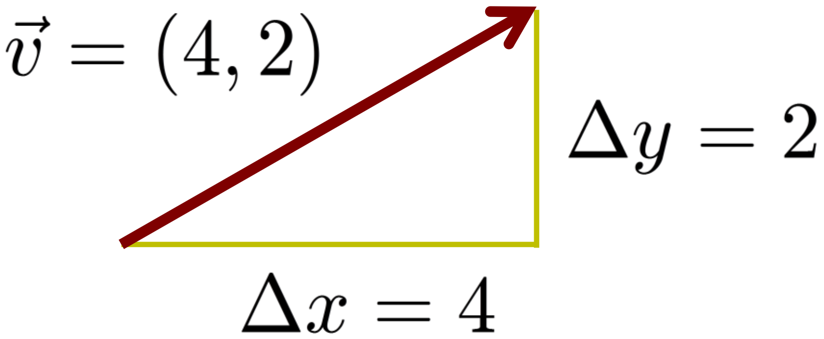


Cannot be added or multiplied together

Vectors: “Arrows in Space”

Vectors are **point changes**

Also number tuple: coordinate changes

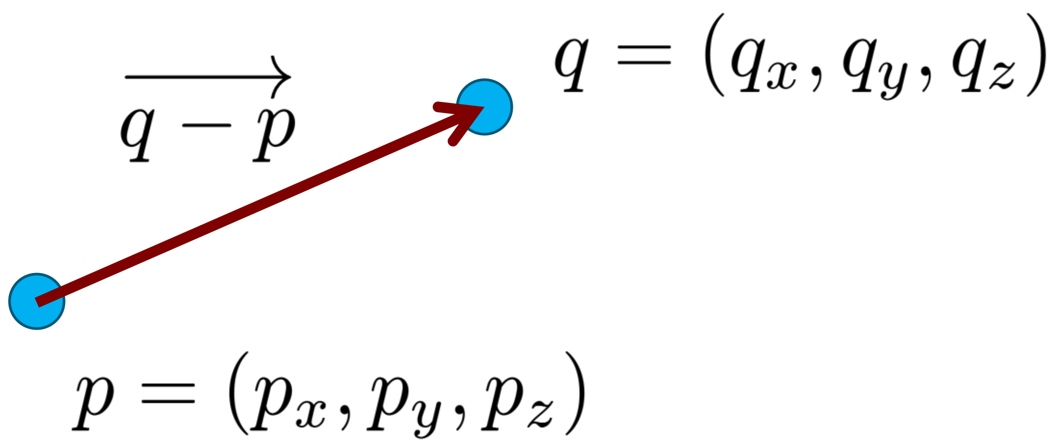


Exist **independent** of any reference point

Vector Arithmetic

Subtracting points gives vectors

- Vector between p and q: q - p

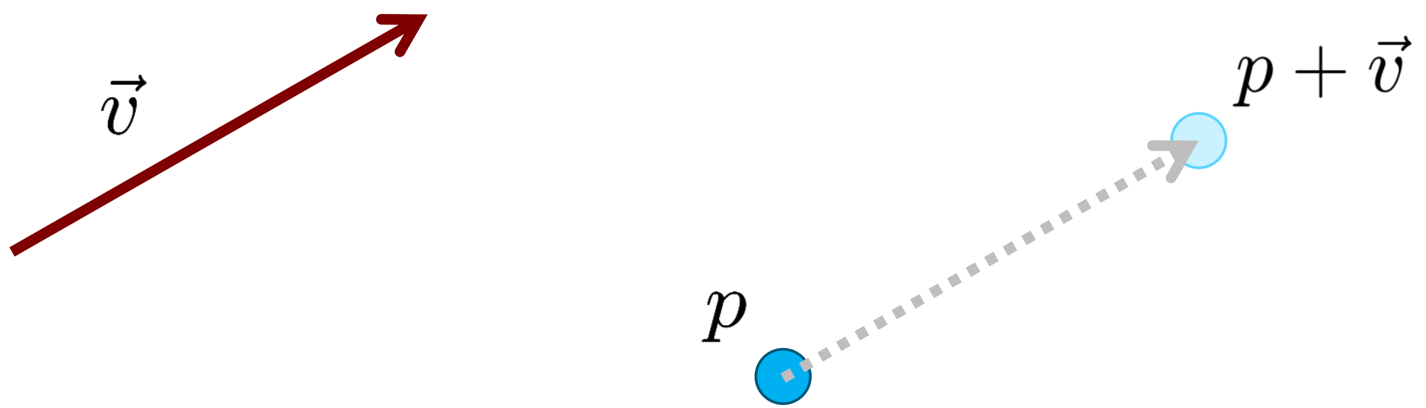


Vector Arithmetic

Subtracting points gives vectors

- Vector between p and q: q – p

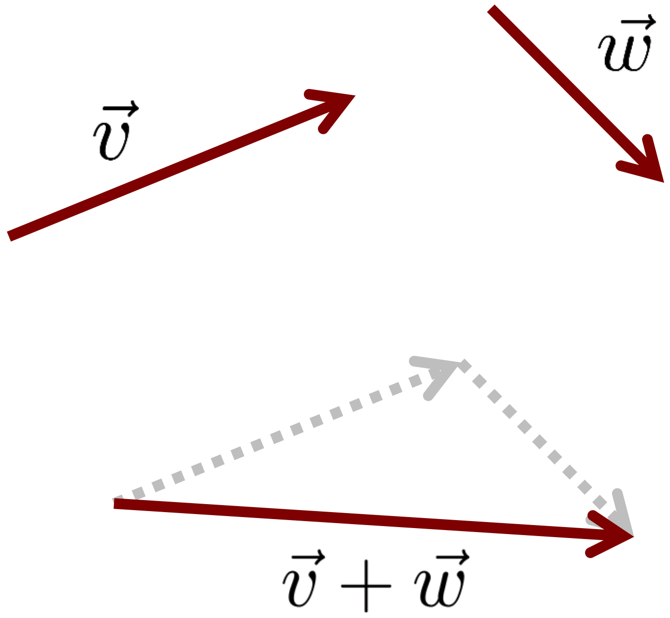
Add vector to point to get new point



Vector Arithmetic

Vectors can be

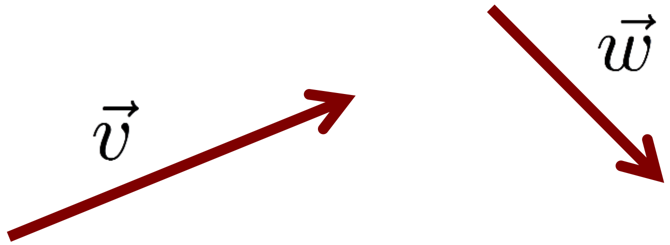
- added (tip to tail)



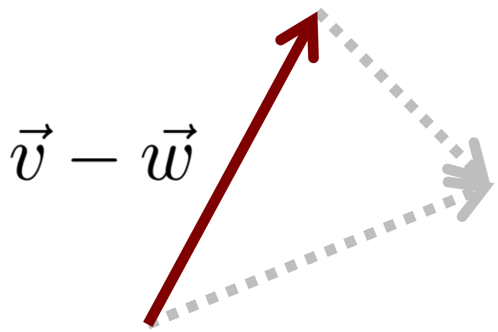
Vector Arithmetic

Vectors can be

- added (tip to tail)



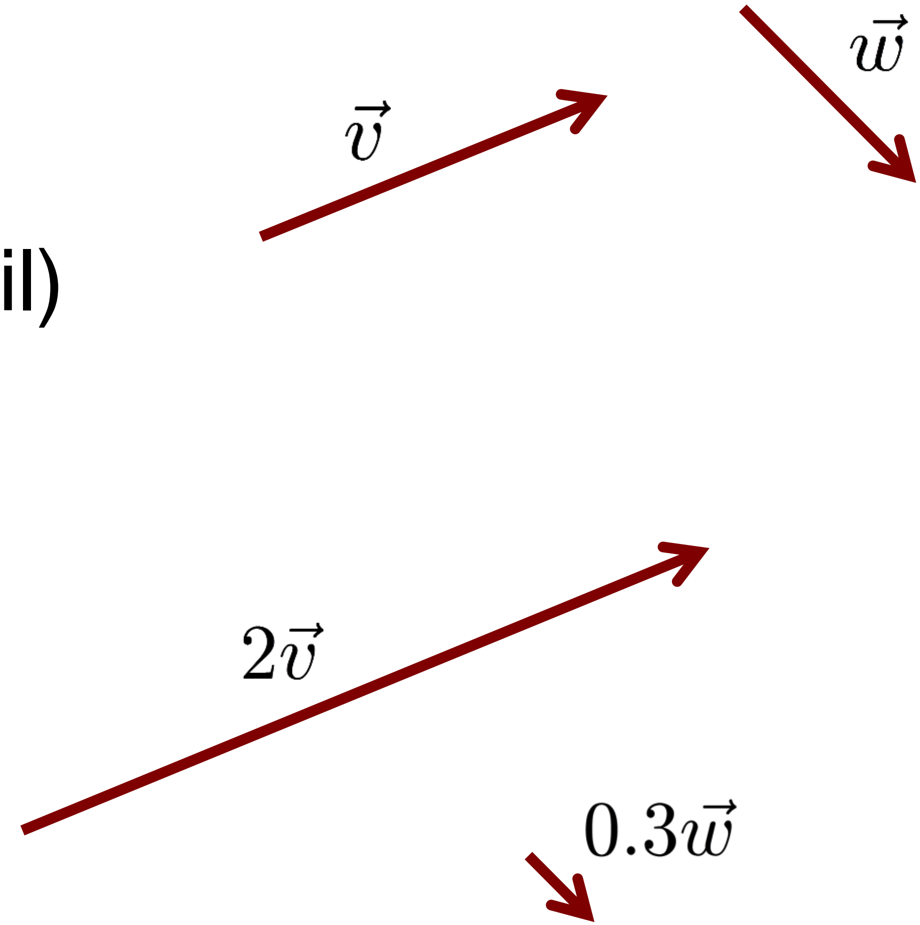
- subtracted



Vector Arithmetic

Vectors can be

- added (tip to tail)
- subtracted
- scaled

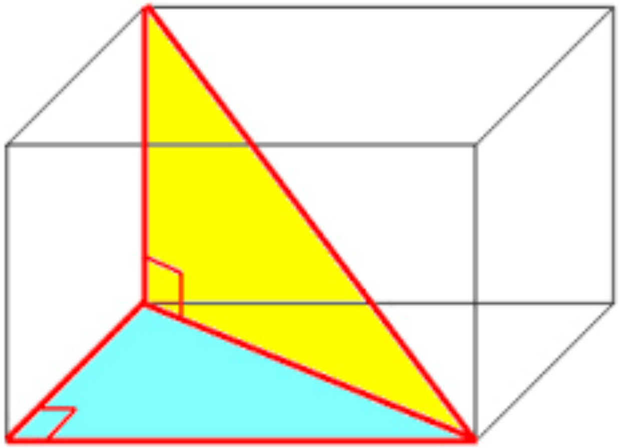


Vector Norm

Vectors have magnitude (length or **norm**)

$$\|\vec{v}\| = \sqrt{v_x^2 + v_y^2 + v_z^2 + \cdots}$$

- *n*-dimensional Pythagorean theorem

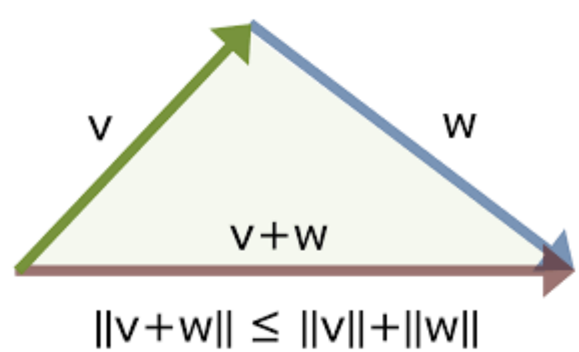


Vector Norm

Vectors have magnitude (length or **norm**)

$$\|\vec{v}\| = \sqrt{v_x^2 + v_y^2 + v_z^2 + \cdots}$$

Triangle inequality: $\|\vec{v} + \vec{w}\| \leq \|\vec{v}\| + \|\vec{w}\|$



Unit Vectors

Vectors with $\|\vec{v}\| = 1$ **unit** or **normalized**

- encode pure direction

Borrowed from physics: “hat notation” $\hat{v}$

Any non-zero vector can be normalized:

$$\hat{v} = \frac{\vec{v}}{\|\vec{v}\|}$$

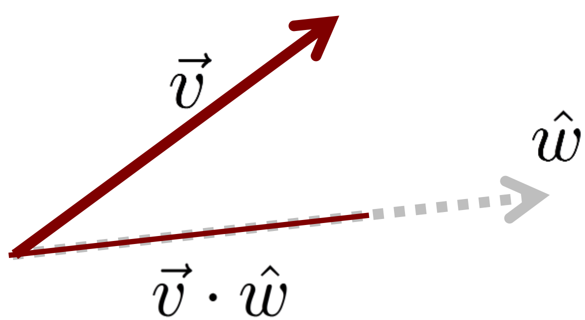
Dot Product

Takes two vectors, returns scalar

$$\vec{v} \cdot \vec{w} = v_x w_x + v_y w_y + v_z w_z + \cdots$$

- (works in any dimension)

$\vec{v} \cdot \hat{w}$ is length of $\vec{v}$ “in the $\hat{w}$ direction”



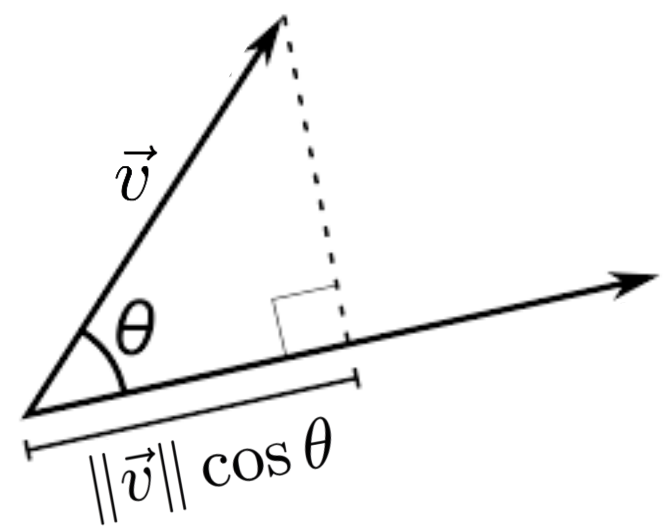
Dot Product

Takes two vectors, returns scalar

$$\vec{v} \cdot \vec{w} = v_x w_x + v_y w_y + v_z w_z + \cdots$$

Alternate formula:

$$\vec{v} \cdot \hat{w} = \|\vec{v}\| \cos\theta$$



Dot Product

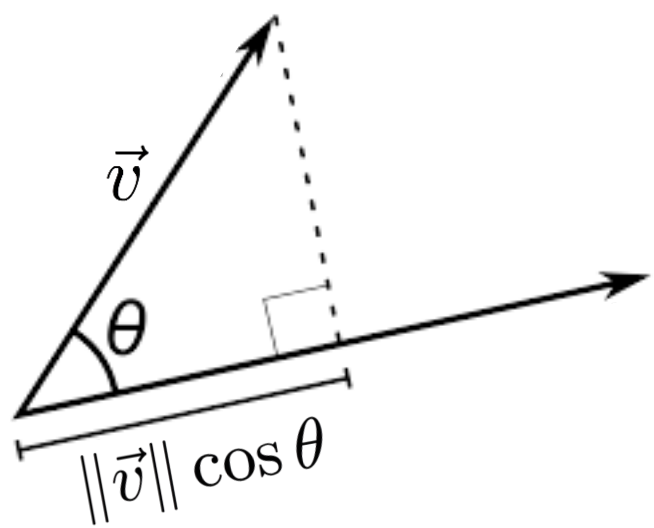
Takes two vectors, returns scalar

$$\vec{v} \cdot \vec{w} = v_x w_x + v_y w_y + v_z w_z + \cdots$$

Alternate formula:

$$\vec{v} \cdot \hat{w} = \|\vec{v}\| \cos\theta$$

$$\vec{v} \cdot \vec{w} = \|\vec{v}\| \|\vec{w}\| \cos\theta$$



Dot Product and Angles

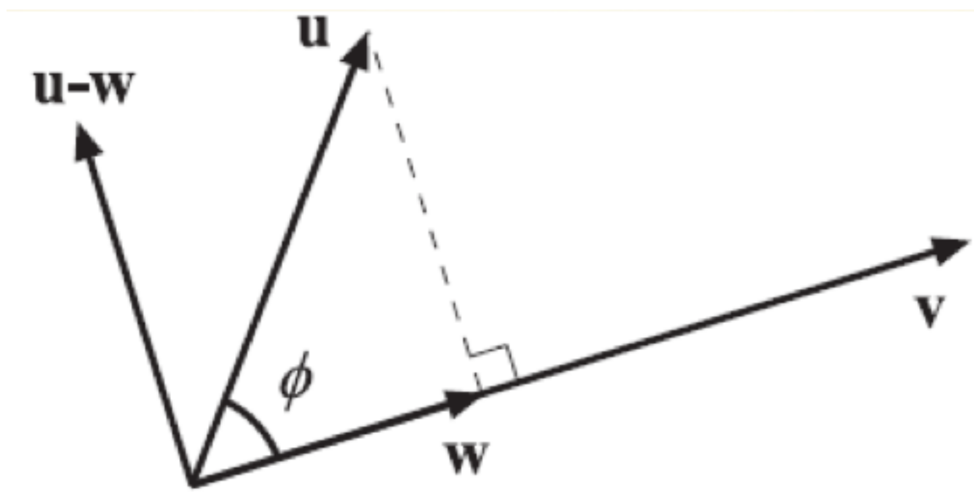
Note $\cos\theta = \frac{\vec{v}\cdot\vec{w}}{||\vec{v}||\,||\vec{w}||}$ requires only multiplications and sqrts

Useful because trig calls are **slow**

Also in a pinch (slow): $\theta = \arccos\frac{\vec{v}\cdot\vec{w}}{||\vec{v}||\,||\vec{w}||}$

Vector Projection

Projection onto and out of



$$\mathbf{w} = \frac{\mathbf{u} \cdot \mathbf{v}}{\mathbf{v} \cdot \mathbf{v}} \mathbf{v}$$

$$\mathbf{u} - \mathbf{w}$$

Cross Product

Takes two vectors, returns vector

$$\vec{v} \times \vec{w} = (v_y w_z - v_z w_y, v_z w_x - v_x w_z, v_x w_y - v_y w_x)$$

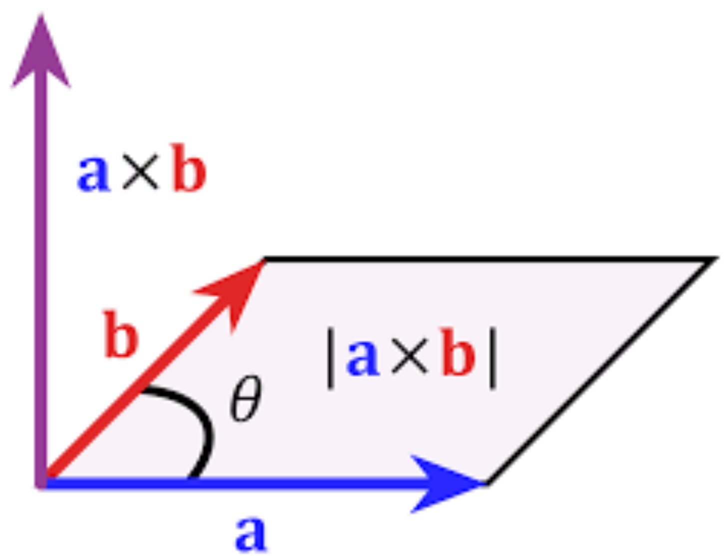
- works **only** in 3D

Direction: perpendicular to both $\vec{v}, \vec{w}$

Magnitude: $\|\vec{v} \times \vec{w}\| = \|\vec{v}\| \|\vec{w}\| \sin\theta$

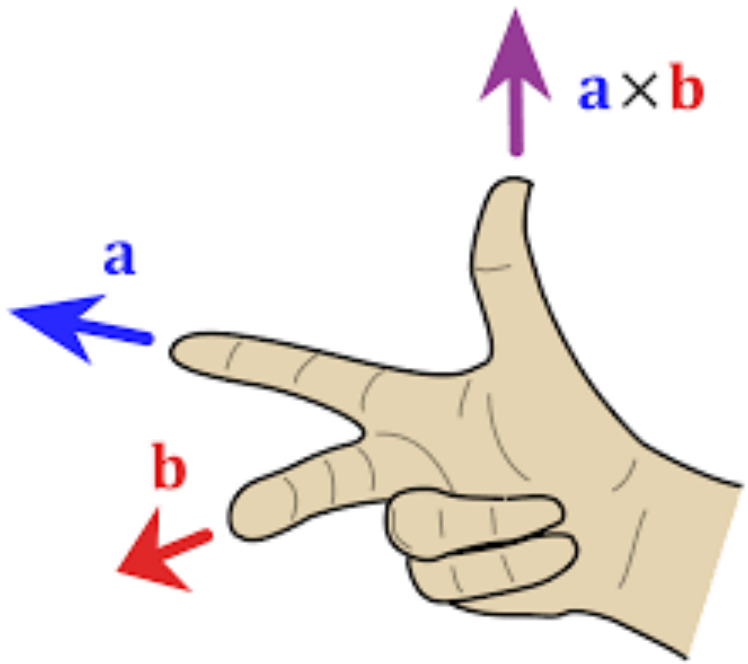
Cross Product Intuition

- Magnitude is **area of parallelogram** formed by vectors
- Orthogonal direction to vectors

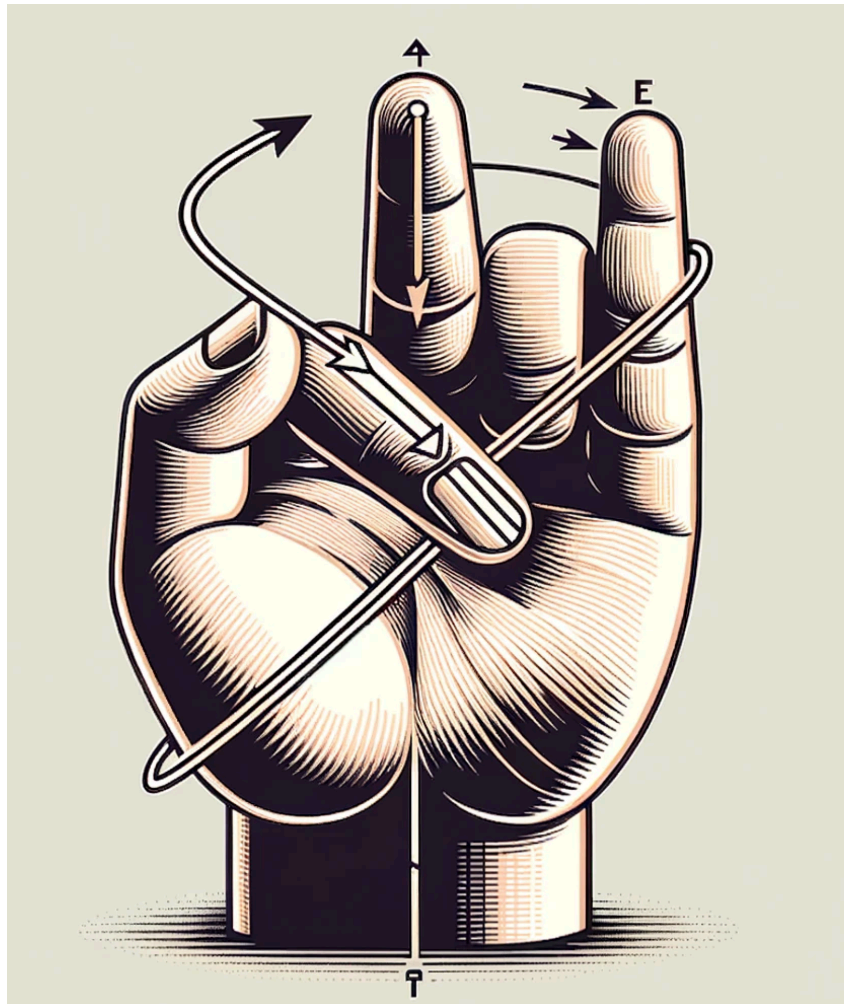


Cross Product Intuition

Use **right-hand rule**



“Right-hand” “Rule” (via DALL-E)



Cross Product Uses

Easily computes unit vector perpendicular to two given vectors

$$\hat{n} = \frac{u \times v}{\|u \times v\|}$$

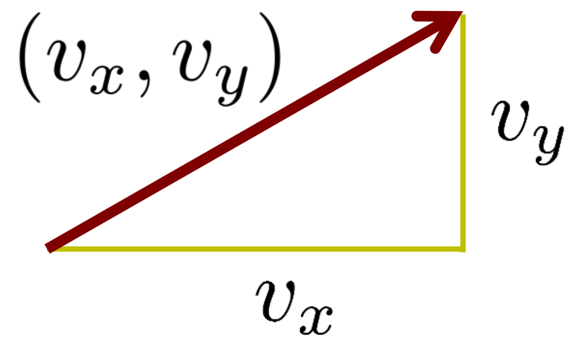
Relation to angles: $\sin\theta = \frac{\|u \times v\|}{\|u\|\|v\|}$

Even better: $\tan\theta = \frac{\|u \times v\|}{u \cdot v}$ no sqrts!

Euclidean Coordinates

A vector in 2D (v_x, v_y) can be interpreted as instructions

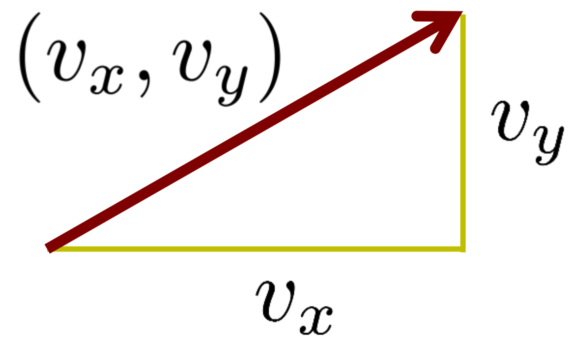
"move to the right v_x and up v_y"



Euclidean Coordinates

A vector in 2D (v_x, v_y) can be interpreted as instructions

“move to the right v_x *and up* v_y*”*



In other words: $(v_x, v_y) = v_x\hat{x} + v_y\hat{y}$

(Finite) Vector Spaces

We say: 2D vectors are **vector space** of vectors **spanned** by **basis vectors** $\{\hat{x}, \hat{y}\}$

- basis vectors: “directions” to travel
- span: all linear combinations

Dimension

Dimension is size of biggest set of linearly independent basis vectors

Adding all vectors in vector space to point *p* when dimension is:

- 0: just *p*
- 1: line through *p*
- 2: plane through *p*
- 3+: hyperplane through *p*

Vectors and Bases

- Pick a basis, order the vectors in it
  - All vectors in the space can be represented as a sequence of coordinates
- Examples:
  - Cartesian 3-space
  - Basis [i j k]
  - Linear combination: xi + yj + zk
  - Coordinate representation [x y z]

Identity Matrix

Square diagonal matrix:

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

Identity matrix doesn't change the matrix it's multiplied by

Inverse Matrix

- For some square matrices, inverse exists:

  $$AA^{-1} = A^{-1}A = I$$

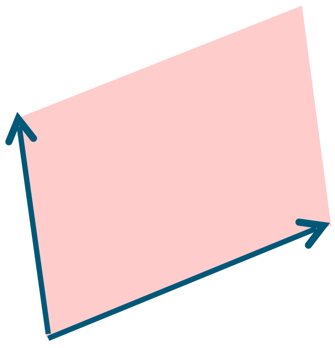
- A non-invertible matrix is called **singular** (determinant is 0)
- Very expensive to compute on large matrices!

Determinant

Maps square matrix to real number

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc$$

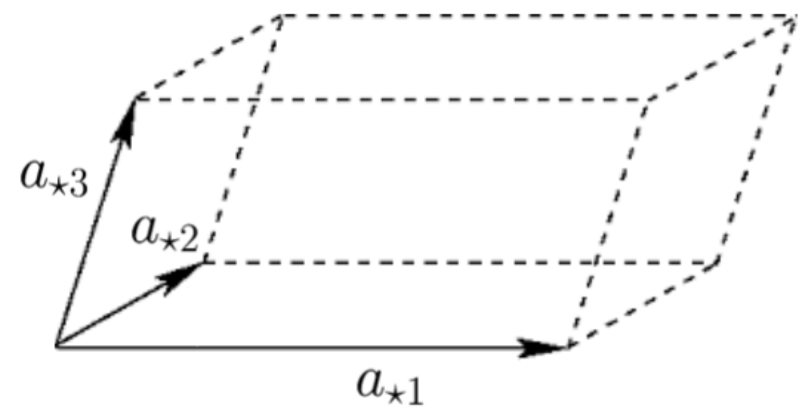
In 2D measures signed volume of parallelogram



Determinant

In 3D, measures signed volume of parallelpiped

$$\det \left[\vec{a}_{\star 1} \middle| \vec{a}_{\star 2} \middle| \vec{a}_{\star 3} \right]$$



Transpose

Flips indices; “reflect about diagonal”

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}^T = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \\ a_{13} & a_{23} \end{bmatrix}$$

Transpose of vector is **row vector**

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}^T = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}$$

Properties of Orthogonal Matrices

Orthogonal matrices have columns and rows that are orthonormal

- All vectors of unit length
- All vectors perpendicular to each other

Guaranteed to be invertible

Inverse is the transpose