Tutorial 1

Survival Mathematics

Computer Graphics

Summer Semester 2020

Ludwig-Maximilians-Universität München

Welcome!

Agenda

- Point and Vector
- Coordinate Systems
- Scalar and Vector Operations
- Matrix and Determinant
- Basics of JavaScript

Tutorial 1: Survival Mathematics

- Point and Vector
- Coordinate Systems
- Scalar and Vector Operations
- Matrix and Determinant
- Basics of JavaScript

Point v.s. Vector

- A point encodes a specific ***location***
  - An exact information
  - A reference is needed
  - In the Cartesian coordinate system, the reference point is the ***origin***

- A vector encodes ***direction*** and ***magnitude***
  - Given a reference point, a vector can look like a point, e.g. $\mathbf{v} = (x_1, x_2, x_3)^\top \in \mathbb{R}^3$

Task 1

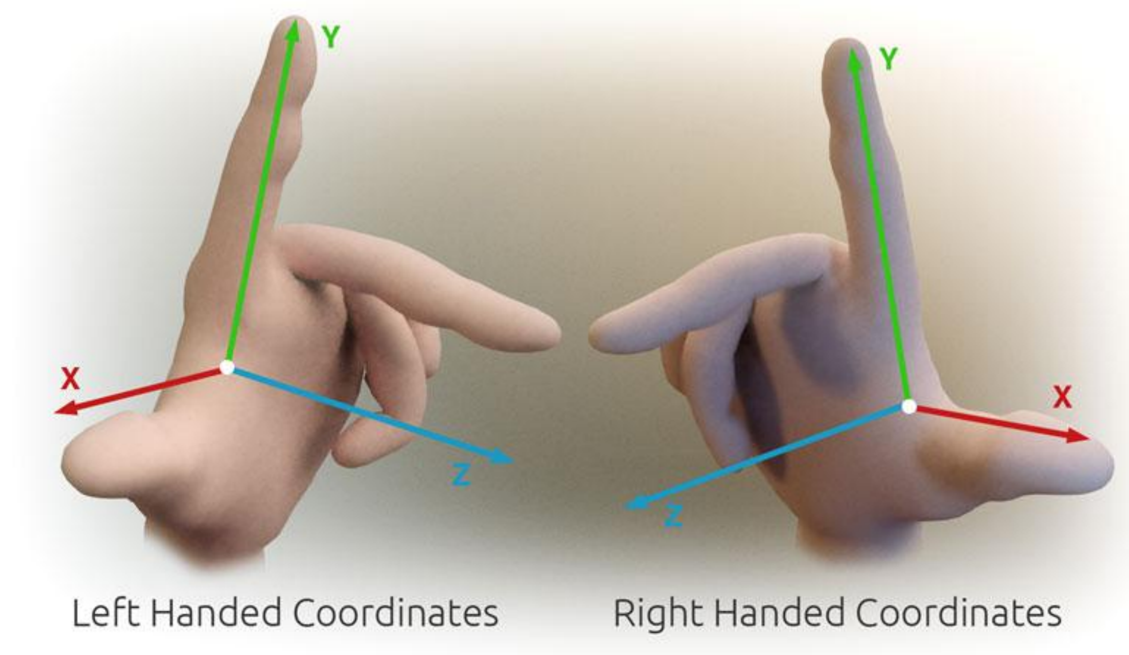
- "The lecture was held at 10 a.m. yesterday" ⇒ ***Point***
  - Reference point: today
  - Location: 10 a.m.
- "The exam lasts 90 minutes" ⇒ ***Vector***
  - Direction: time lapse
  - Magnitude: 90 minutes
- "The metro station is 100 meters away to the south of the office" ⇒ ***Point***
  - Reference point: the office
  - Location: 100 meters away to the south
- "The highest standing jump is 1.651 meters" ⇒ ***Vector***
  - Direction: jump up
  - Magnitude: 1.651 meters

Tutorial 1: Survival Mathematics

- Point and Vector
- Coordinate Systems
- Scalar and Vector Operations
- Matrix and Determinant
- Basics of JavaScript

Coordinate Systems

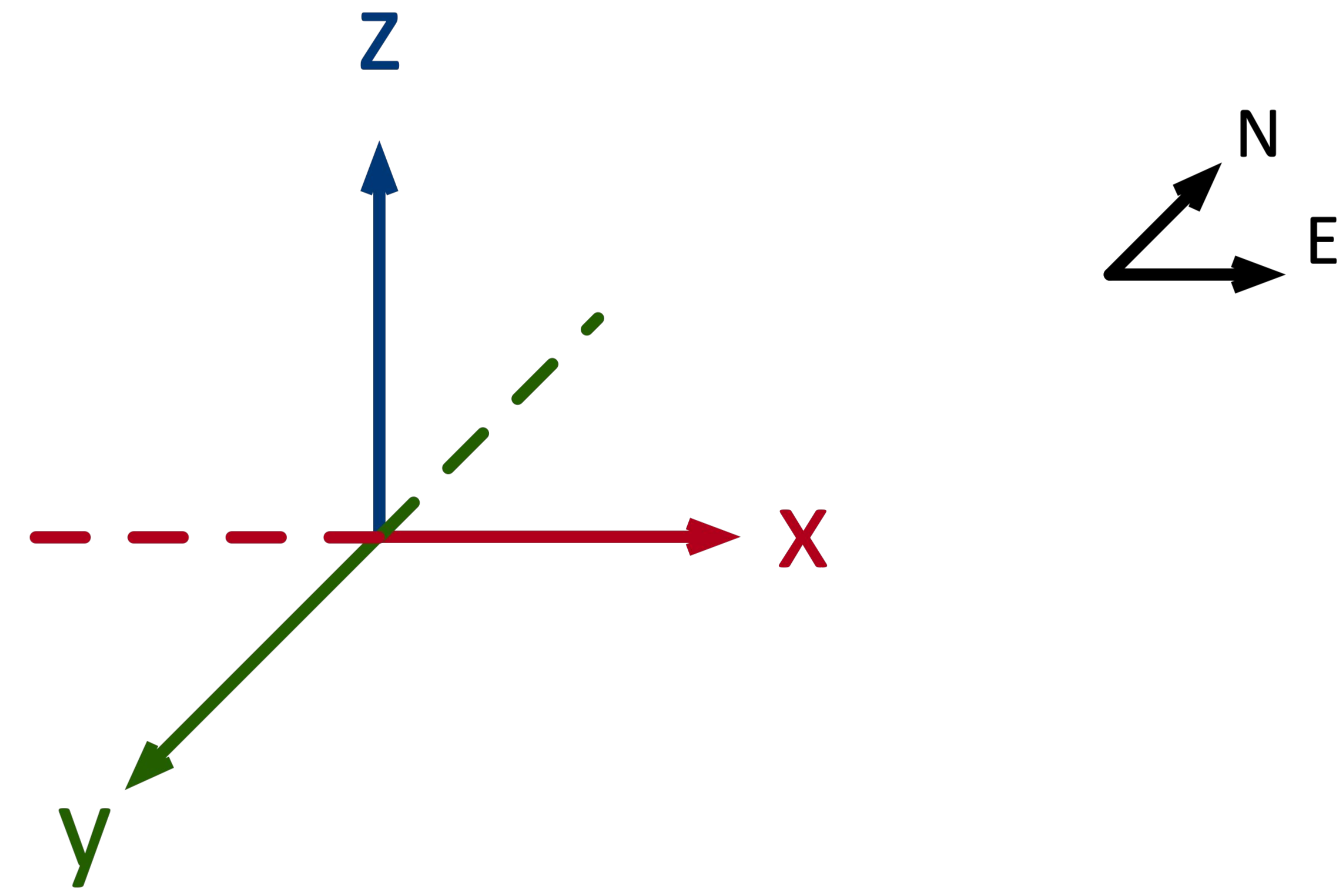
- Left handed coordinates v.s. Right handed coordinates
  - Y-axis upward (both)
- OpenGL: Right handed
  - positive Z-axis points at camera
- Direct3D: Left handed
  - Z-axis on the opposite side comparing to OpenGL =>
  - positive Z-axis points away from camera



- Why?
  - Historical reason: personal preference, random decision

Task 2: b) Left or right?

- x-axis points to the east
- y-axis points to the south
- z-axis points to the top



⇒ Left-handed

OpenGL is *right-handed*

"*each axis lies in the same line with respect to the corresponding axis*"

⇒ no guarantees on directions!

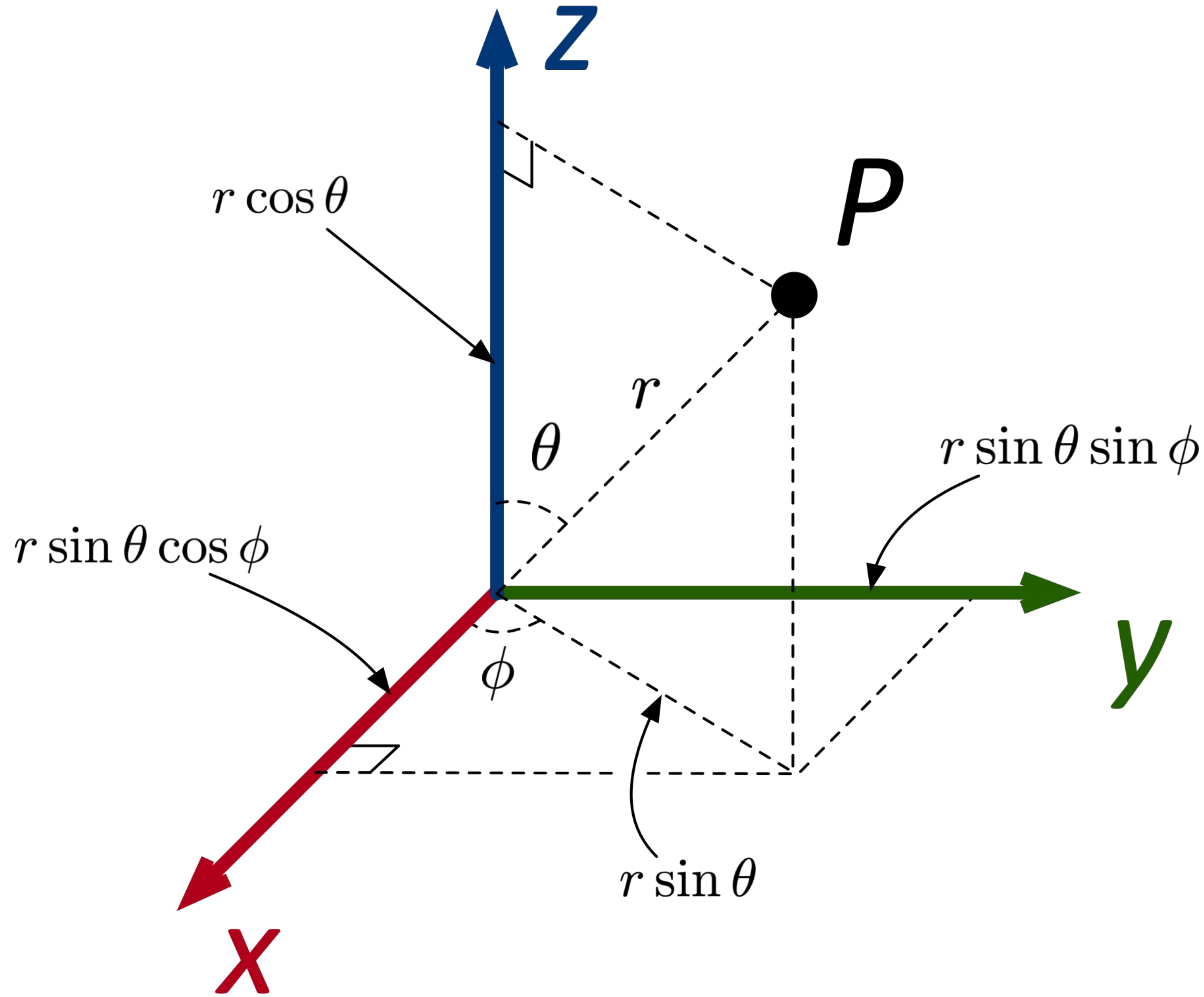
Three possibilities:

1. if the direction of x- and z-axis remains: (x,y,z) ⇒ (x,-y,z) ⇒ ***P = (3, -4, 5)***
2. if the direction of x- and y-axis remains: (x,y,z) ⇒ (x,y,-z) ⇒ ***P = (3, 4, -5)***
3. if the direction of y- and z-axis remains: (x,y,z) ⇒ (-x,y,z) ⇒ ***P = (-3, 4, 5)***

Task 2: c) Spherical coordinates

$$P = (r, \theta, \phi)$$

$$\rightarrow (r \sin\theta \cos\phi, r \sin\theta \sin\phi, r \cos\theta)$$



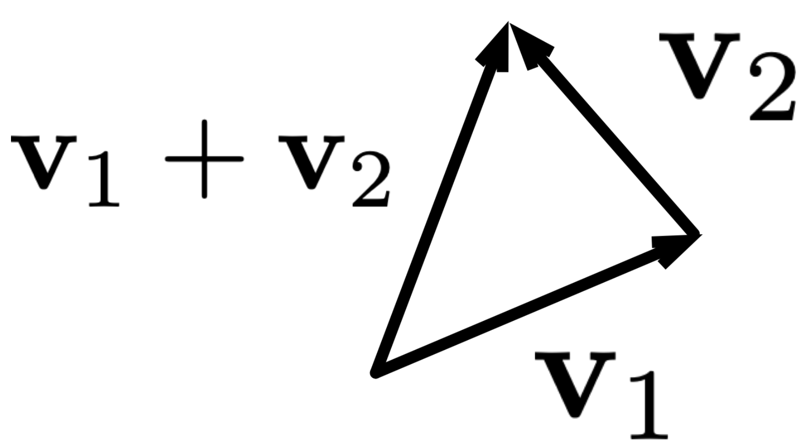
Q: Do you need consider the left or right handed coordinates here?

Tutorial 1: Survival Mathematics

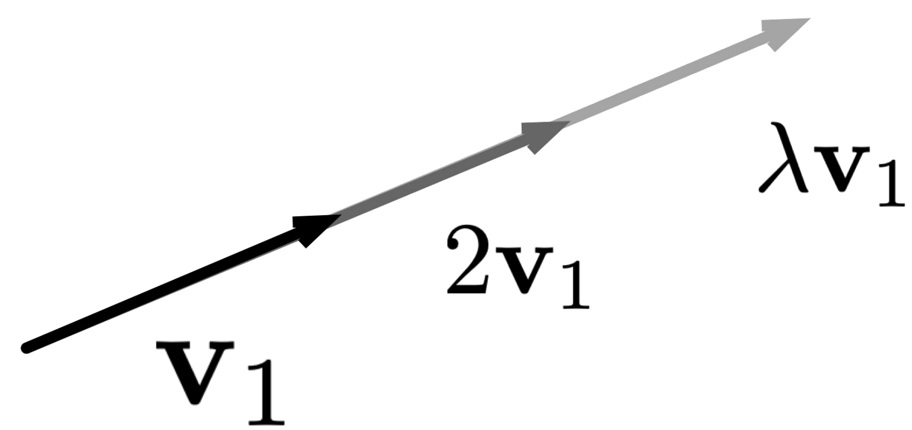
- Point and Vector
- Coordinate Systems
- Scalar and Vector Operations
- Matrix and Determinant
- Basics of JavaScript

Linear Space: Vector Operation, Span

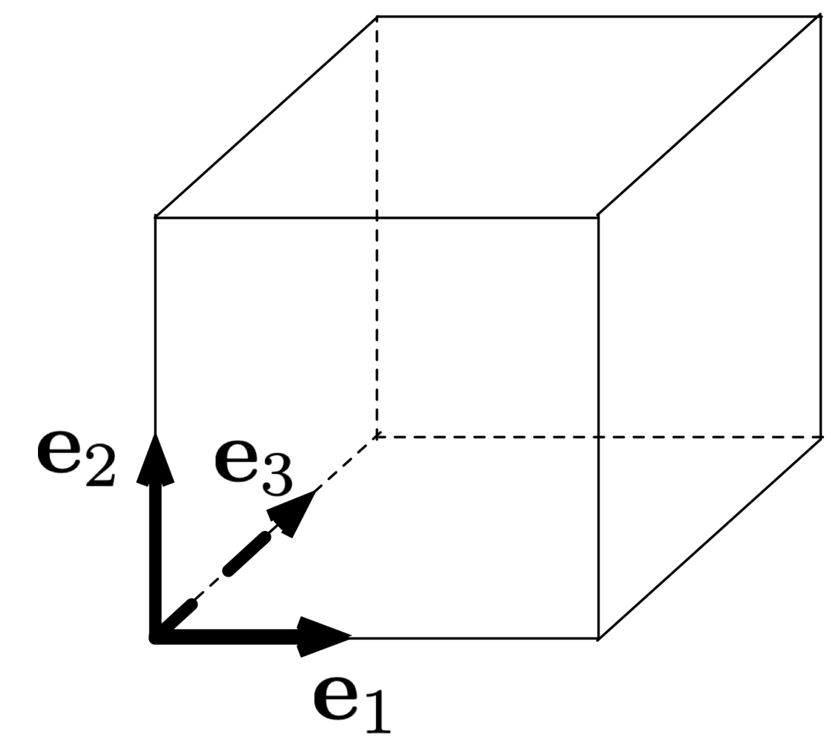
Vector Sum



Scalar-Vector Product



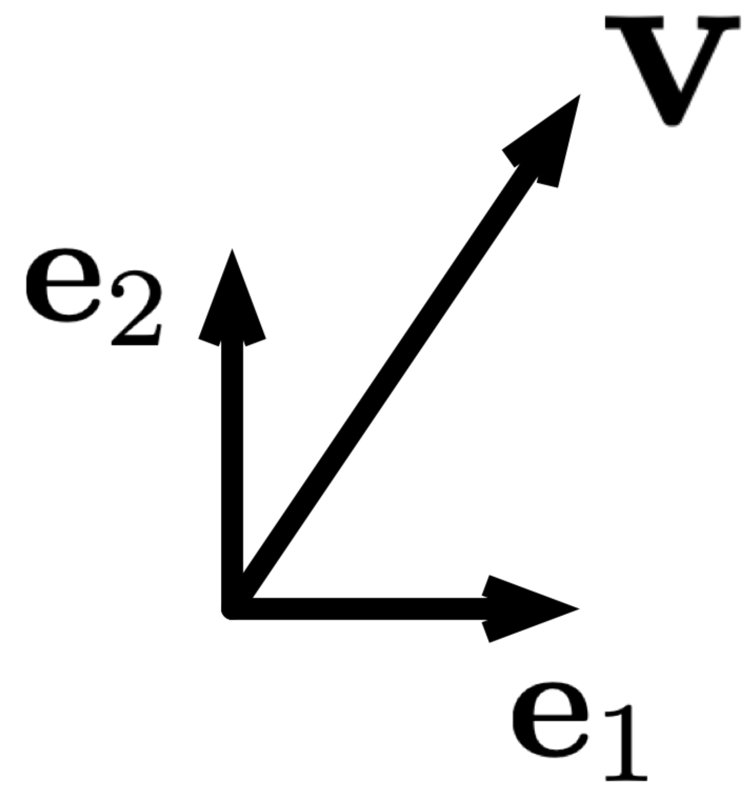
Span



$$\text{span}\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\} = \sum_{i=1}^{3} \lambda_i \mathbf{e}_i, \lambda_i \in [0, \Lambda]$$

Vector: Norm

If **e**₁ and **e**₂ are basis vectors

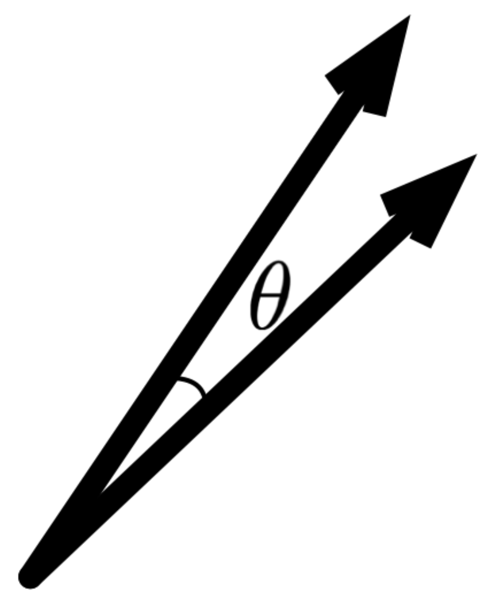


$$||\mathbf{v}|| = ||\lambda_1 \mathbf{e}_1 + \lambda_2 \mathbf{e}_2|| = \sqrt{\lambda_1^2 + \lambda_2^2}$$

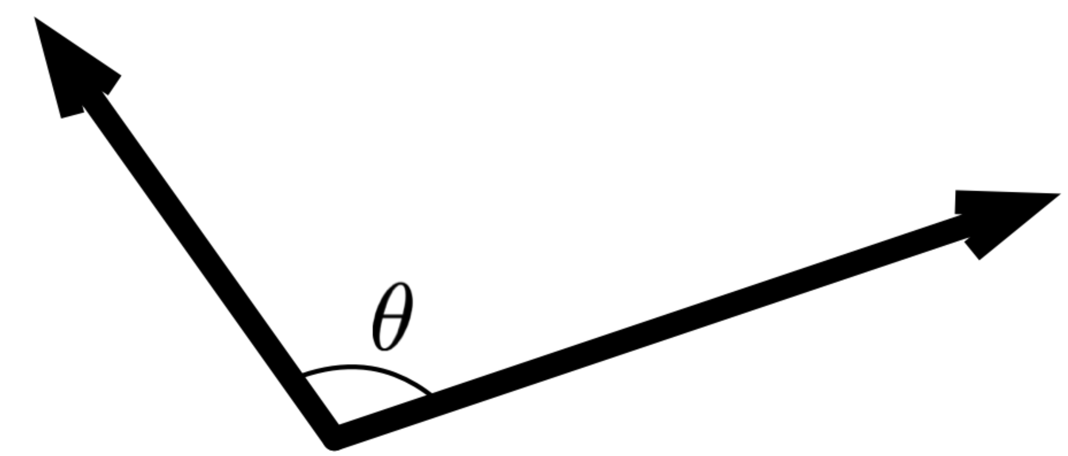
Essence of coordinates

If $\mathbf{e}_1$ and $\mathbf{e}_2$ are basis vectors, then the coordinates of $\mathbf{v} = (\lambda_1, \lambda_2)^\top$

Vector: Angle and Dot Product



"similar"



"different"

$$\text{“similarity”} = \cos\theta = \frac{\mathbf{v}_1^\top \cdot \mathbf{v}_2}{||\mathbf{v}_1||||\mathbf{v}_2||} \in [-1, 1]$$

Task 3

a) $$a\mathbf{v}_1 + b\mathbf{v}_2 + c\mathbf{v}_3 = 1 \times (2,1,2)^\top + 2 \times (1,1,3)^\top - 3 \times (1,2,-2)^\top$$
$$= (2,1,2)^\top + (2,2,6)^\top - (3,6,-6)^\top = (1,-3,14)^\top$$

b) $$||\mathbf{v}_1|| = \sqrt{2^2 + 1^2 + 2^2} = 3$$
$$||\mathbf{v}_2|| = \sqrt{1^2 + 1^2 + 3^2} = \sqrt{11}$$
$$||\mathbf{v}_3|| = \sqrt{1^2 + 2^2 + (-2)^2} = 3$$

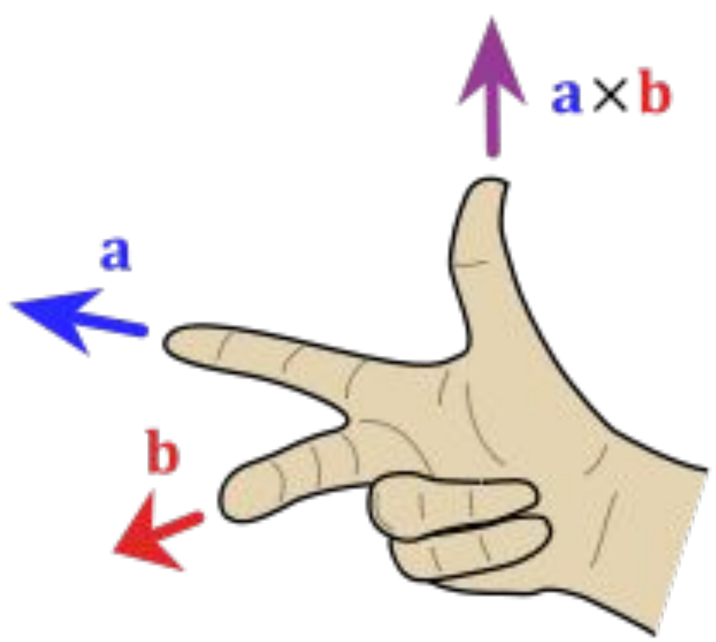
c) $$\angle(\mathbf{v}_1, \mathbf{v}_2) = \arccos\left(\frac{(2,1,2) \cdot (1,1,3)^\top}{3\sqrt{11}}\right) = \arccos\left(\frac{3\sqrt{11}}{11}\right)$$
$$\angle(\mathbf{v}_2, \mathbf{v}_3) = \arccos\left(\frac{(1,1,3) \cdot (1,2,-2)^\top}{3\sqrt{11}}\right) = \arccos\left(-\frac{\sqrt{11}}{11}\right) = \pi - \arccos\left(\frac{\sqrt{11}}{11}\right)$$
$$\angle(\mathbf{v}_3, \mathbf{v}_1) = \arccos\left(\frac{(1,2,-2) \cdot (2,1,2)^\top}{3 \times 3}\right) = \arccos 0 = \frac{\pi}{2} + 2\pi n, n \in \mathbb{N}$$

Vector: Cross Product

For 3D vectors, by *definition*:

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{pmatrix}$$

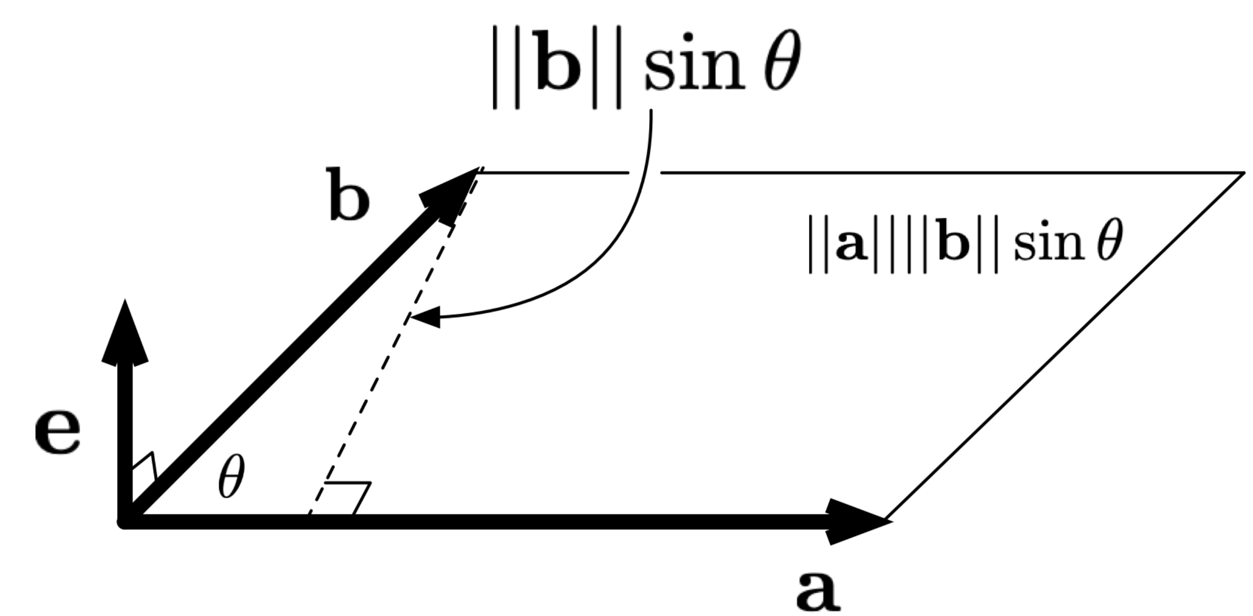
What's the meaning of this definition?!?



"right-handed"

*If **e** is a unit vector orthogonal w.r.t **a** and **b**:*

$$\mathbf{a} \times \mathbf{b} = \mathbf{e}||\mathbf{a}||||\mathbf{b}||\sin\theta$$



Task 3

d) $$\mathbf{v}_1 \times \mathbf{v}_2 = (1, -4, 1)^\top$$

$$\mathbf{v}_2 \times \mathbf{v}_1 = -\mathbf{v}_1 \times \mathbf{v}_2 = (-1, 4, -1)^\top$$

e) $$\mathbf{v}_1 \times (\mathbf{v}_2 + \mathbf{v}_3) = (-5, 2, 4)^\top$$

$$\mathbf{v}_1 \times \mathbf{v}_2 + \mathbf{v}_1 \times \mathbf{v}_3 = \mathbf{v}_1 \times (\mathbf{v}_2 + \mathbf{v}_3) = (-5, 2, 4)^\top$$

f) **Do we really need calculate?? cross product results in an *orthogonal* vector**

$$\mathbf{v}_1 \times \mathbf{v}_1 = \mathbf{v}_2 \times \mathbf{v}_2 = \mathbf{v}_3 \times \mathbf{v}_3 = \mathbf{0}$$ ***(zero vector, not scalar)***

g) **Do we really need calculate?? cross product results in an *orthogonal* vector**

$$\mathbf{v}_1^\top \cdot (\mathbf{v}_1 \times \mathbf{v}_2) = \mathbf{v}_2^\top \cdot (\mathbf{v}_1 \times \mathbf{v}_2) = 0$$ ***(scalar zero, not vector)***

Task 3 h) *Jacobi Identity*

Lemma: ***Lagrange's identity***

(won't prove here)

$$\mathbf{v}_1 \times (\mathbf{v}_2 \times \mathbf{v}_3) = (\mathbf{v}_1^\top \cdot \mathbf{v}_3)\mathbf{v}_2 - (\mathbf{v}_1^\top \cdot \mathbf{v}_2)\mathbf{v}_3$$

$$\mathbf{v}_2 \times (\mathbf{v}_3 \times \mathbf{v}_1) = (\mathbf{v}_2^\top \cdot \mathbf{v}_1)\mathbf{v}_3 - (\mathbf{v}_2^\top \cdot \mathbf{v}_3)\mathbf{v}_1$$

$$\mathbf{v}_3 \times (\mathbf{v}_1 \times \mathbf{v}_2) = (\mathbf{v}_3^\top \cdot \mathbf{v}_2)\mathbf{v}_1 - (\mathbf{v}_3^\top \cdot \mathbf{v}_1)\mathbf{v}_2$$

$$\mathbf{v}_1 \times (\mathbf{v}_2 \times \mathbf{v}_3) + \mathbf{v}_2 \times (\mathbf{v}_3 \times \mathbf{v}_1) + \mathbf{v}_3 \times (\mathbf{v}_1 \times \mathbf{v}_2)$$

$$= (\mathbf{v}_1^\top \cdot \mathbf{v}_3)\mathbf{v}_2 - (\mathbf{v}_1^\top \cdot \mathbf{v}_2)\mathbf{v}_3 + (\mathbf{v}_2^\top \cdot \mathbf{v}_1)\mathbf{v}_3 - (\mathbf{v}_2^\top \cdot \mathbf{v}_3)\mathbf{v}_1 + (\mathbf{v}_3^\top \cdot \mathbf{v}_2)\mathbf{v}_1 - (\mathbf{v}_3^\top \cdot \mathbf{v}_1)\mathbf{v}_2$$

$$= \mathbf{0}$$

(zero vector, not scalar)

So your final result should be a 0 vector.

Tutorial 1: Survival Mathematics

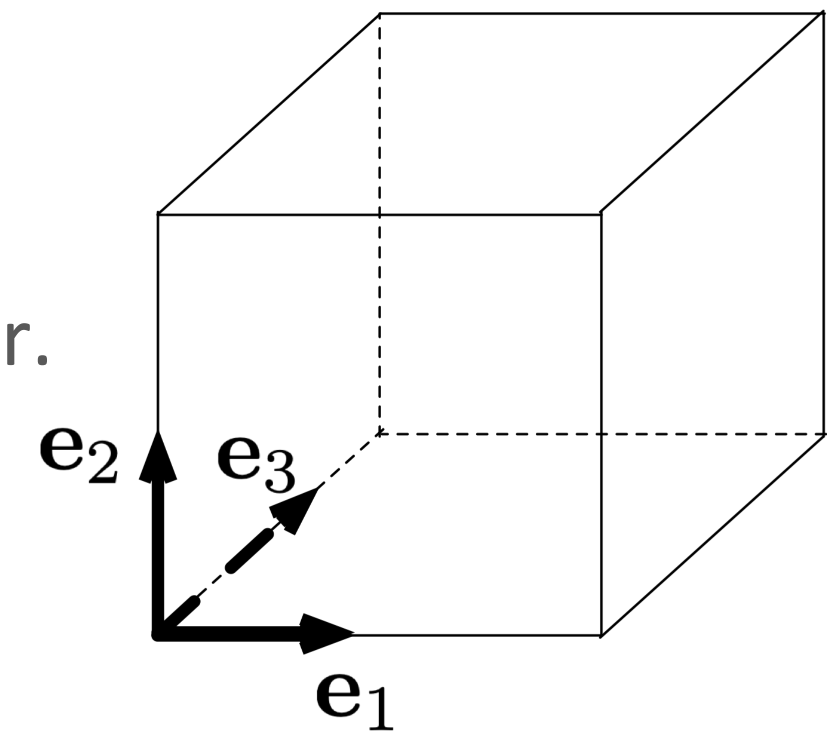
- Point and Vector
- Coordinate Systems
- Scalar and Vector Operations
- **Matrix and Determinant**
- Basics of JavaScript

Span (again)

A space of all possible linearly combined basis vectors.

Orthonormal basis: basis vectors being orthogonal to one another.

Cube Span



Task 4 a) b) and c)

3-dimensional space $\text{span}\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\} = \mathbb{R}^3$

orthonormal basis: unit vectors $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

$$\text{span}\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\} = \sum_{i=1}^{3} \lambda_i \mathbf{e}_i, \lambda_i \in [0, \Lambda]$$

$$S' = \{\mathbf{v} | \mathbf{v} = (2x + y, x + y, 2x + 3y)^\top, x, y \in \mathbb{R}\}$$

Note that S' is (isomorphic to) a 2D space, because $\mathbf{v}_1 + \mathbf{v}_2$ is a linear combination of $\mathbf{v}_1, \mathbf{v}_2$

Therefore $\text{span}\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_1 + \mathbf{v}_2\} = \mathbb{R}^2$ is also acceptable and preferred (only for this course)

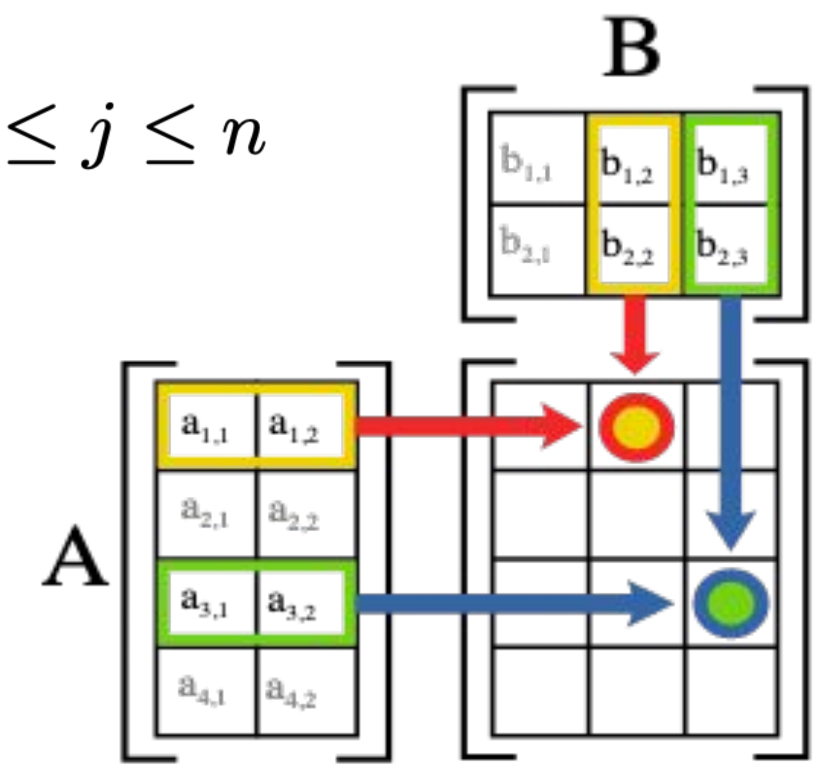
orthonormal basis for $\mathbb{R}^2$ $(\mathbf{e}_1, \mathbf{e}_2) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

Matrix

Addition, subtraction, scalar multiplication are element-wise computed.

Matrix multiplication is more interesting to us:

$$\text{Matrix } \mathbf{C}_{m\times n} = \mathbf{A}_{m\times p} \cdot \mathbf{B}_{p\times n} \text{ where } \quad c_{i,j} = \sum_{k=1}^{p} a_{i,k} b_{k,j}, 1 \leq i \leq m, 1 \leq j \leq n$$



Computation process is labor extensive, and boring.

⇒ *code it!*

What if $\mathbf{A}_{m\times p_1} \cdot \mathbf{B}_{p_2\times n}$ where $p_1 \neq p_2$??

Undefined.

Task 4

d) ***If we treat v_1^T as a 1x3 matrix multiplied by v_1 as a 3x1 matrix***, the result is a 1x1 matrix:

$$\mathbf{v}_1^\top \cdot \mathbf{v}_1 = (9)$$

Q: Hold on, 1x1 matrix? Shouldn't the result be a scalar?

A: No! You can multiply a scalar with an arbitrary matrix, but you cannot multiply a 1x1 matrix with an arbitrary matrix.

Q: What are you talking about? You said the *dot product* results in a scalar.

A: Clarification: we are running into a notation issue here.

Mathematically speaking, the dot product is different from matrix multiplication.

We are in a matrix multiplication context now. To address these notation conflicts, we can use another notion to represent the dot product: $< \mathbf{v}_1, \mathbf{v}_2 >$

Task 4

d) ***If we treat*** $\mathbf{v}_1$ ***as a 3x1 matrix multiplied by*** $\mathbf{v}_1^T$ ***as a 1x3 matrix***, the result is a 3x1 matrix:

$$\mathbf{v}_1 \cdot \mathbf{v}_1^\top = \begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix} \begin{pmatrix} 2 & 1 & 2 \end{pmatrix} = \begin{pmatrix} 4 & 2 & 4 \\ 2 & 1 & 2 \\ 4 & 2 & 4 \end{pmatrix}$$

e) Because the matrix multiplication 3x1 by 3x1 is **undefined**.

Determinant

For the determinant of a 2×2 matrix $\mathbf{B}$ is computed by:

$$\det(\mathbf{B}) = \begin{vmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{vmatrix} = b_{11}b_{22} - b_{21}b_{12}$$

And the determinant of 3×3 matrix $\mathbf{C}$ is computed by:

$$\det(\mathbf{C}) = \begin{vmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{vmatrix} = c_{11} \begin{vmatrix} c_{22} & c_{23} \\ c_{32} & c_{33} \end{vmatrix} - c_{12} \begin{vmatrix} c_{21} & c_{23} \\ c_{31} & c_{33} \end{vmatrix} + c_{13} \begin{vmatrix} c_{21} & c_{22} \\ c_{31} & c_{32} \end{vmatrix}$$

Vector: Cross Product (Revisited)

$$\mathbf{a} \times \mathbf{b} = \begin{pmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{pmatrix} = (a_2 b_3 - a_3 b_2) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + (a_3 b_1 - a_1 b_3) \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + (a_1 b_2 - a_2 b_1) \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

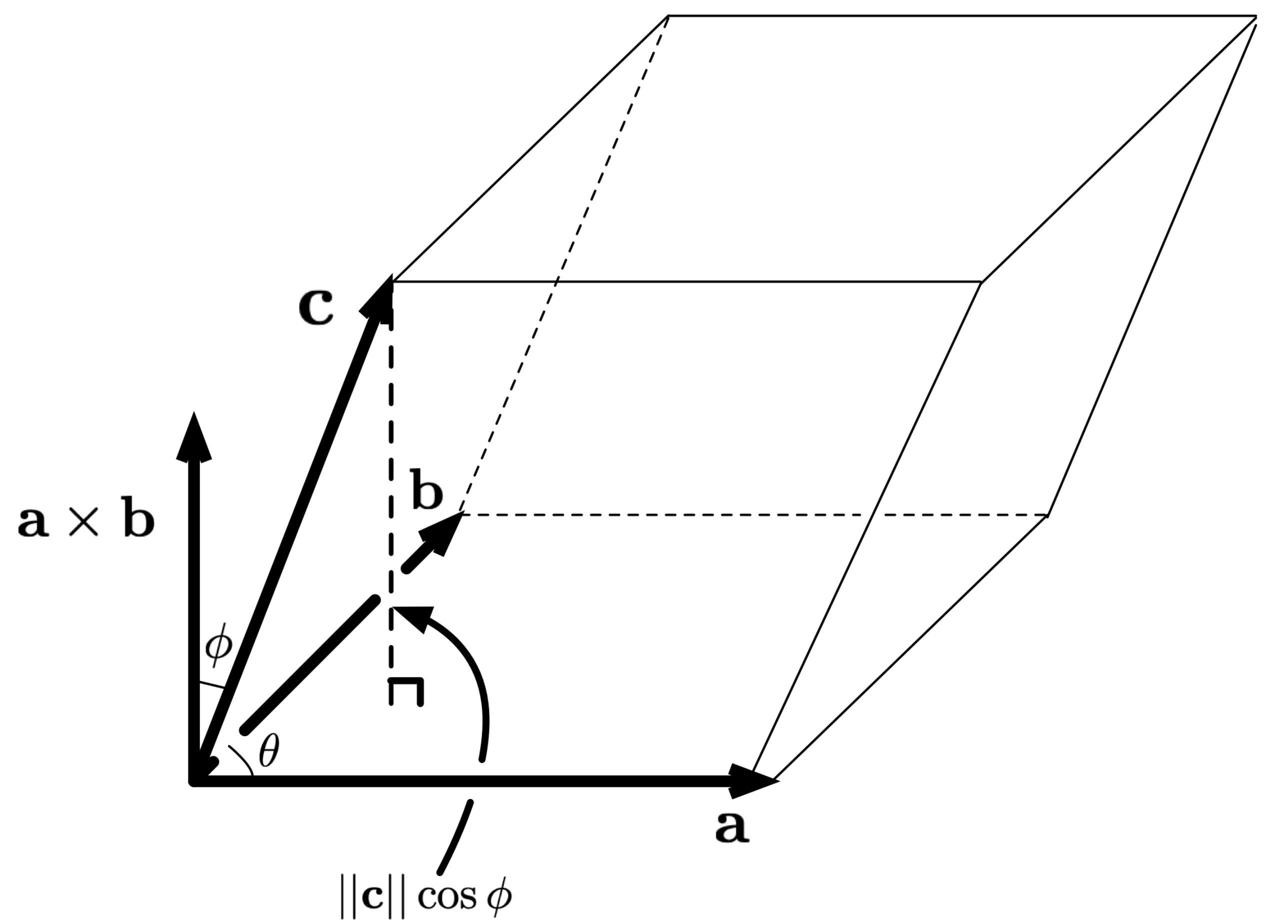
$$= \begin{vmatrix} a_2 & a_3 \\ b_2 & b_3 \end{vmatrix} \mathbf{e}_1 - \begin{vmatrix} a_1 & a_3 \\ b_1 & b_3 \end{vmatrix} \mathbf{e}_2 + \begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} \mathbf{e}_3$$

$$= \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix}$$

mnemonic!

Vector: Cross Product (Revisited)

$$\mathbf{c}^\top \cdot (\mathbf{a} \times \mathbf{b}) = (\mathbf{c}^\top \cdot \mathbf{e})||\mathbf{a}||||\mathbf{b}|| \sin\theta = \underbrace{||\mathbf{c}|| \cos\phi}_{\text{height}} \underbrace{||\mathbf{a}||||\mathbf{b}|| \sin\theta}_{\text{bottom surface}}$$



$$\mathbf{c}^\top \cdot (\mathbf{a} \times \mathbf{b}) = (c_1, c_2, c_3)^\top \cdot \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix} = \begin{vmatrix} c_1 & c_2 & c_3 \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix}$$

determinant is a volume!

Task 4

f) -9

g) Parallelepiped of $\mathbf{v}_1, \mathbf{v}_2, \lambda_1\mathbf{v}_1 + \lambda_2\mathbf{v}_2$? They are on the same plane, no volume!

Therefore $\det(V) = 0$

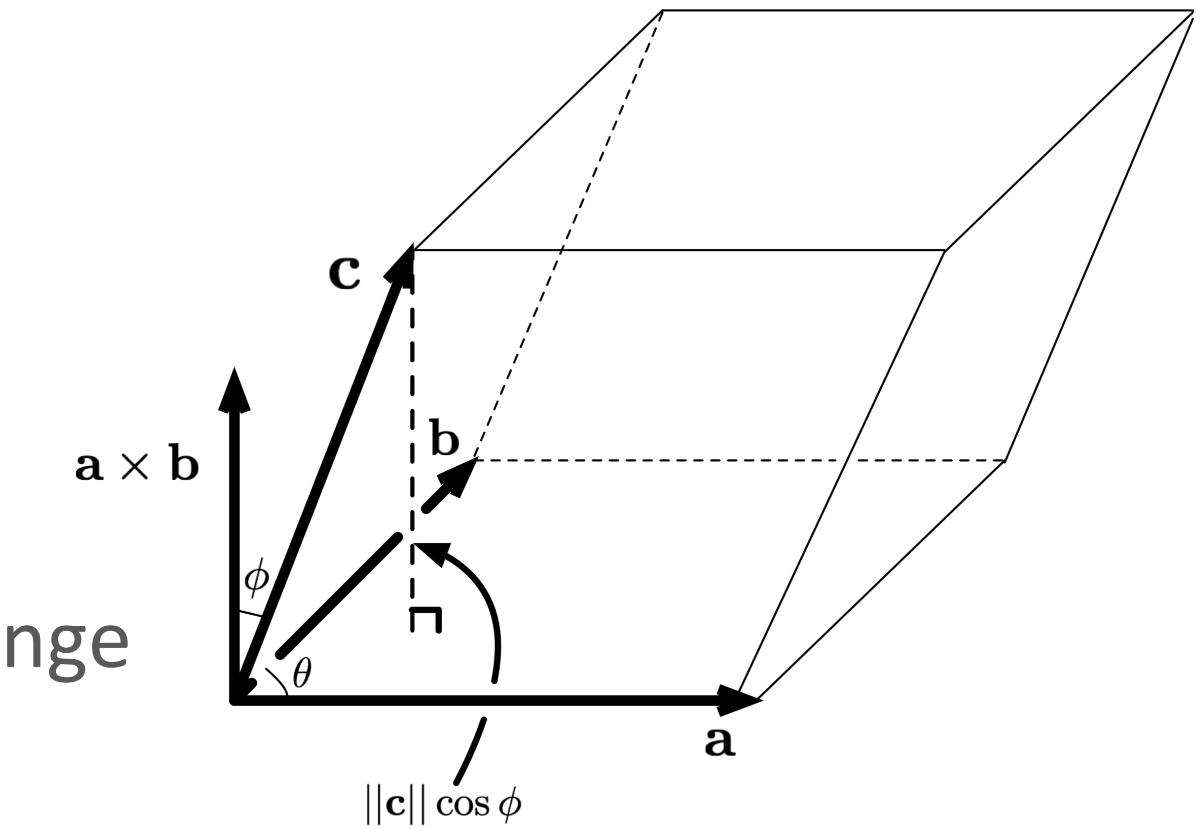
h) linear independent: $\det(V) \neq 0$ $\Rightarrow$ geometric meaning: parallelepiped.

linear dependent: $\det(V) = 0$ $\Rightarrow$ geometric meaning: 2D plane

i) All equal to the volume of the parallelepiped

$$\mathbf{v}_1^\top \cdot (\mathbf{v}_2 \times \mathbf{v}_3) = \mathbf{v}_2^\top \cdot (\mathbf{v}_3 \times \mathbf{v}_1) = \mathbf{v}_3^\top \cdot (\mathbf{v}_1 \times \mathbf{v}_2)$$

More Determinants



Lemmas (won't prove here):

1. $\det(V) = \det(V^\top)$

2. If we swap two rows (columns), the determinant will change its sign.

$$\mathbf{c}^\top \cdot (\mathbf{a} \times \mathbf{b}) = (c_1, c_2, c_3)^\top \cdot \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix} = \begin{vmatrix} c_1 & c_2 & c_3 \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix}$$

$$= \begin{vmatrix} c_1 & a_1 & b_1 \\ c_2 & a_2 & b_2 \\ c_3 & a_3 & b_3 \end{vmatrix} = - \begin{vmatrix} a_1 & c_1 & b_1 \\ a_2 & c_2 & b_2 \\ a_3 & c_3 & b_3 \end{vmatrix} = \begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix} = \det(\mathbf{a}, \mathbf{b}, \mathbf{c})$$

$$\det(\mathbf{a}, \mathbf{b}, \mathbf{c}) = \mathbf{a}^\top \cdot (\mathbf{b} \times \mathbf{c}) = \mathbf{b}^\top \cdot (\mathbf{c} \times \mathbf{a}) = \mathbf{c}^\top \cdot (\mathbf{a} \times \mathbf{b})$$

Task 4

j) (kinda) recursively defined. Watch the *sign*.

$$\begin{vmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{vmatrix} = c_{11}c_{22} - c_{21}c_{12}$$

$$\begin{vmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{vmatrix} = c_{11} \begin{vmatrix} c_{22} & c_{23} \\ c_{32} & c_{33} \end{vmatrix} - c_{12} \begin{vmatrix} c_{21} & c_{23} \\ c_{31} & c_{33} \end{vmatrix} + c_{13} \begin{vmatrix} c_{21} & c_{22} \\ c_{31} & c_{32} \end{vmatrix}$$

$$\begin{vmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{vmatrix} = c_{11} \begin{vmatrix} c_{22} & c_{23} & c_{24} \\ c_{32} & c_{33} & c_{34} \\ c_{42} & c_{43} & c_{44} \end{vmatrix} - c_{12} \begin{vmatrix} c_{21} & c_{23} & c_{24} \\ c_{31} & c_{33} & c_{34} \\ c_{41} & c_{43} & c_{44} \end{vmatrix} + c_{13} \begin{vmatrix} c_{21} & c_{22} & c_{24} \\ c_{31} & c_{32} & c_{34} \\ c_{41} & c_{42} & c_{44} \end{vmatrix} - c_{14} \begin{vmatrix} c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \\ c_{41} & c_{42} & c_{43} \end{vmatrix}$$

Tutorial 1: Survival Mathematics

- Point and Vector
- Coordinate Systems
- Scalar and Vector Operations
- Matrix and Determinant
- Basics of JavaScript

Basic Concepts in JavaScript

- ***constant***: immutable data `const c = 3.14`
- ***variable***: mutable data `let v = 0`
- ***function***: a code block maps a list of parameters to a list of return values

  ```
  function F(p1, p2, p3) { … }    (normal function)
  const F = (p1, p2, p3) => { … } (arrow  function)
  ```

  Q: What are the differences?
- ***flow control***: `if/else/switch/for` statements (in almost every-language)
- ***class***: a special "function" with `constructor()` auto-executed when `new C()`

  ```
  class Matrix {
    constructor(m, n, ...xs) {
      this.m = m
      this.n = n
      this.xs = [...xs]
    }
    f() { … }
  }
  ```

  ```
  const m = new Matrix(1, 2,
                       1, 2,
                   )
        m.f()
  ```

Data Types in JavaScript

- ***number:*** `3.1415`
- ***string***: `"hello world!"`
- ***array***: `[1, 2, 3, 4]`
- ***object***: `{course: "MIMUC/CG1", year: 2020, difficulty: "very difficult"}`

Error Handling in JavaScript

```
try {
  throw "throw an error!"
} catch(err) {
  console.log(err) // prints throwed value: "throw an error"
}
```

NodeJS

- What is it?
  - JavaScript is a language (standard), and Node.js is an implementation of it.
- Why do we want it?
  - Browser independent JS execution runtime
  - Better engineering, e.g. dependency management

Node Package Manager

- "Standing on the shoulders of giants" -- Isaac Newton
- Manage declared dependencies in `package.json`, and save dependencies in `node_modules`. Basic usage:

`$ npm init`	create package.json
`$ npm i <pkg_name>`	install package <pkg_name>, e.g. three.js
`$ npm i -D <pkg_name>`	install dev package <pkg_name>, e.g. webpack

Task 5: Vector3

```
class Vector3 {
 constructor(x1, x2, x3) { … }
 sum(w) {
   this.x1 += w.x1
   this.x2 += w.x2
   this.x3 += w.x3
   return this
 }
 multiply(scalar) {
   this.x1 *= scalar
   this.x2 *= scalar
   this.x3 *= scalar
   return this
 }
 dot(w) {
   return this.x1 * w.x1 + this.x2 * w.x2
   + this.x3 * w.x3
 }
 norm() { return Math.sqrt(this.dot(this)) }
 cross(w) {
   const x = this.x2*w.x3 - this.x3*w.x2
   const y = this.x3*w.x1 - this.x1*w.x3
   const z = this.x1*w.x2 - this.x2*w.x1
   return new Vector3(x, y, z)
 }
 angle(w) {
   return Math.acos(this.dot(w) / (this.norm()*w.norm()))
 }
}
```

Task 5: Matrix.multiply

```
multiply(mat) {
  let C = new Matrix(this.m, mat.n, new Array(this.m*mat.n));
  for (let i = 0; i < this.m; i++) {
    for (let j = 0; j < mat.n; j++) {
      let total = 0;
      for (let k = 0; k < this.n; k++) {
        total += this.xs[i*this.n+k]*mat.xs[k*mat.n+j];
      }
      C.xs[i*mat.n+j] = total;
    }
  }
  return C
}
```

$$c_{i,j} = \sum_{k=1}^{p} \text{this}_{i,k} \text{mat}_{k,j}$$

Q: What is the time complexity of this implementation?

⇒ Optimizing matrix multiplication is a *hot* research topic!

Task 5: Matrix.det

```
det() {
  ...
  if (this.m === 2) {
    return this.xs[0]*this.xs[3] - this.xs[1]*this.xs[2]
  }
  // this.m === 3
  return this.xs[0] * (new Matrix(2, 2,
            this.xs[4], this.xs[5],
            this.xs[7], this.xs[8])).det()
        - this.xs[1] * (new Matrix(2, 2,
            this.xs[3], this.xs[5],
            this.xs[6], this.xs[8])).det()
        + this.xs[2] * (new Matrix(2, 2,
            this.xs[3], this.xs[4],
            this.xs[6], this.xs[7])).det()
}
```

$$\det(\mathbf{C}_{3\times 3}) = c_{11} \begin{vmatrix} c_{22} & c_{23} \\ c_{32} & c_{33} \end{vmatrix} - c_{12} \begin{vmatrix} c_{21} & c_{23} \\ c_{31} & c_{33} \end{vmatrix} + c_{13} \begin{vmatrix} c_{21} & c_{22} \\ c_{31} & c_{32} \end{vmatrix}$$

Take Away

- Figure out the geometric meaning behind a formula
- Be thoughtful about your answers, think and write all possibilities
- Programming is important for this course, and you won’t be able to follow along if you refuse to code

Thanks!
What are your questions?