- Exam – when?
- January 7?

Academic year 2015/2016

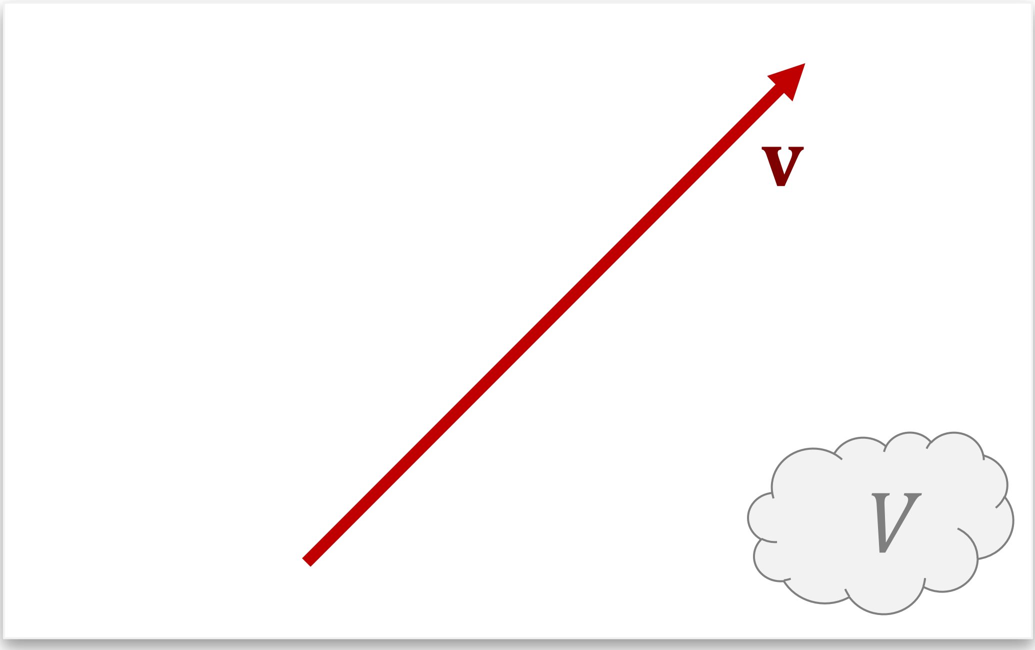
Autumn term: August 31, 2015 – January 18, 2016

- Study period 1: 15-08-31 – 15-10-16
- Own work: 15-10-19 – 15-10-22
- Exam period 1: 15-10-23 – 15-10-30
- Period 2: 15-11-02 – 15-12-18
- Own work: 15-12-21 – 16-01-05
- Own work/Re-exam 1: 16-01-07 – 16-01-09
- Exam period 2: 16-01-11 – 16-01-18

Introduction to Visualization and Computer Graphics
DH2320, Fall 2015
Prof. Dr. Tino Weinkauf

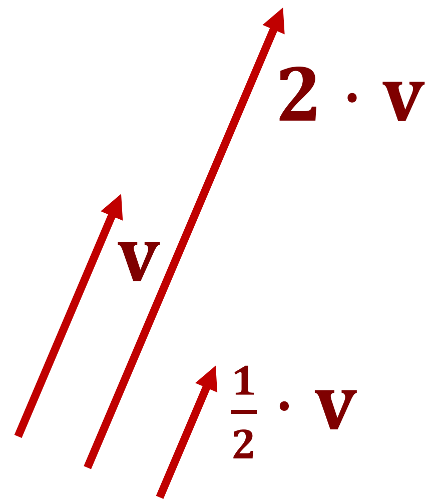
Mathematics

Vectors and Points



- Vectors
  - A vector is an arrow in space
  - We use bold letters for vectors: $\mathbf{u}, \mathbf{v}, \mathbf{w}, \mathbf{x}, \mathbf{y}, \mathbf{z}$ ...
  - Vector space V: set of possible vectors

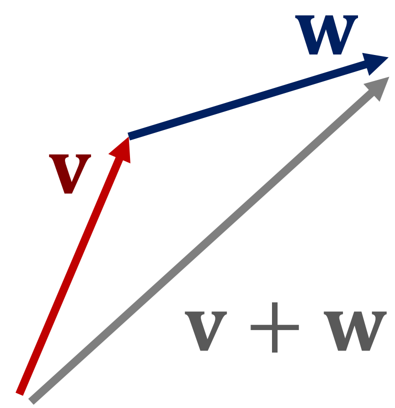
Vector Operations



Scaling Vectors

vector-scalar product

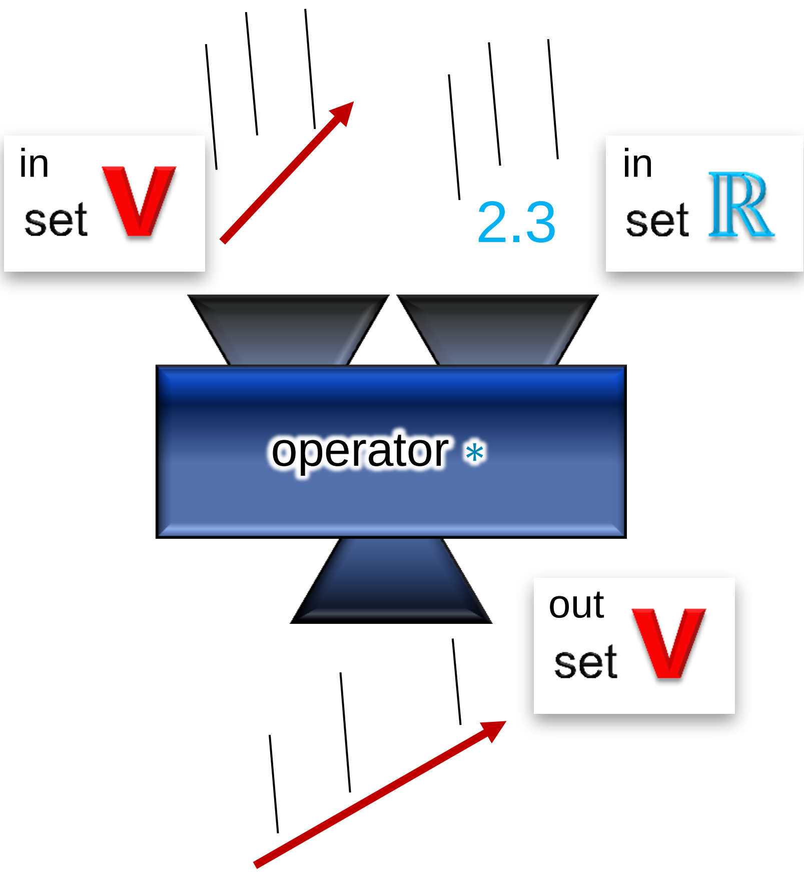
$\lambda \cdot \mathbf{v} \quad (\lambda \in \mathbb{R},\ \mathbf{v} \in V)$



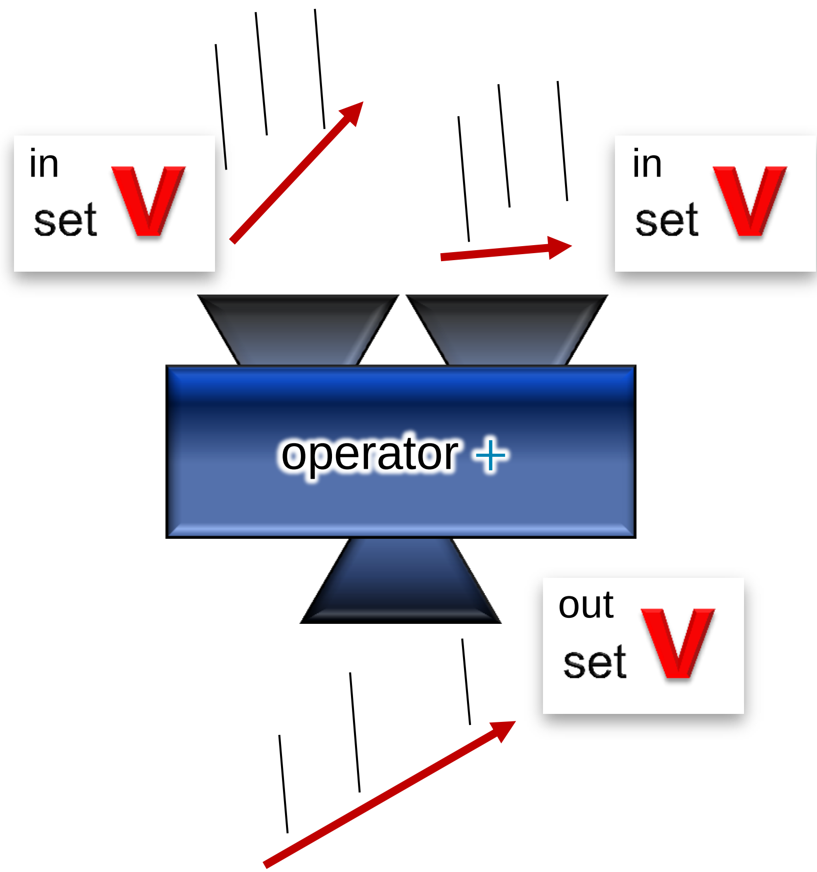
Adding Vectors

vector-addition

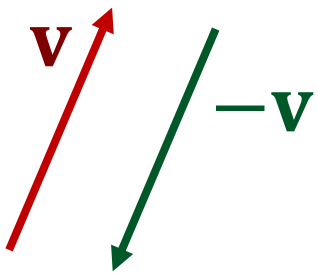
$\mathbf{v} + \mathbf{w} \quad (\mathbf{v}, \mathbf{w} \in V)$



Vector-Scalar Multiplication

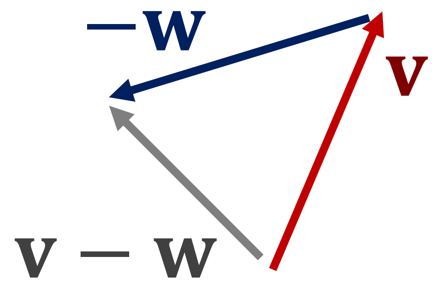


Vector Addition



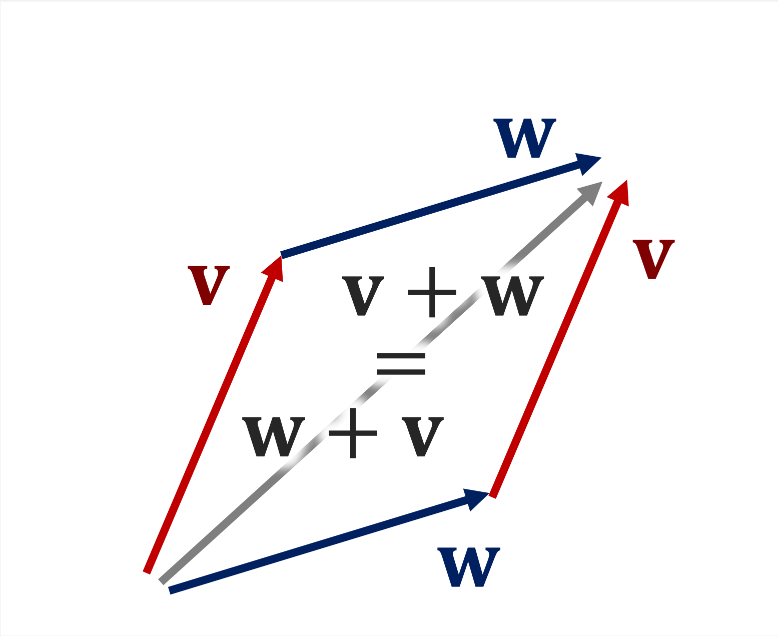
Reversing Vectors

*) special case of scaling



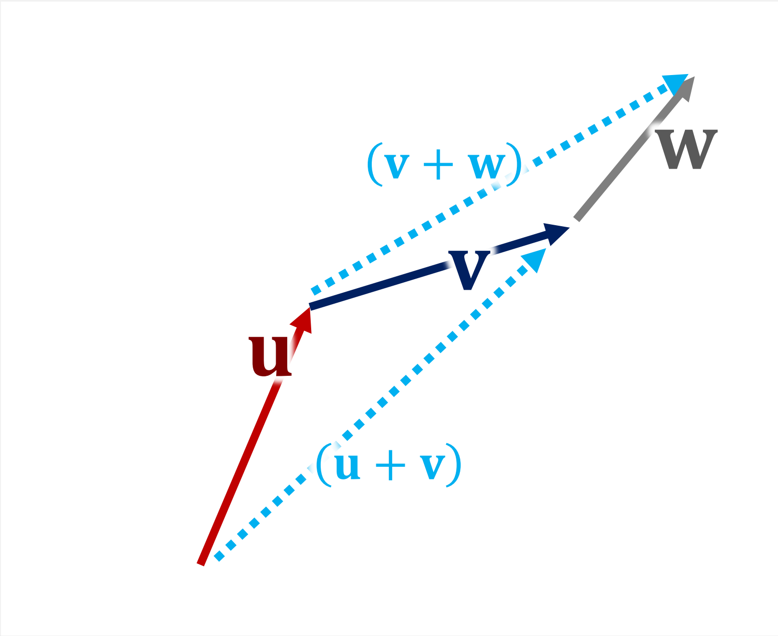
Subtracting Vectors

*) special case of addition



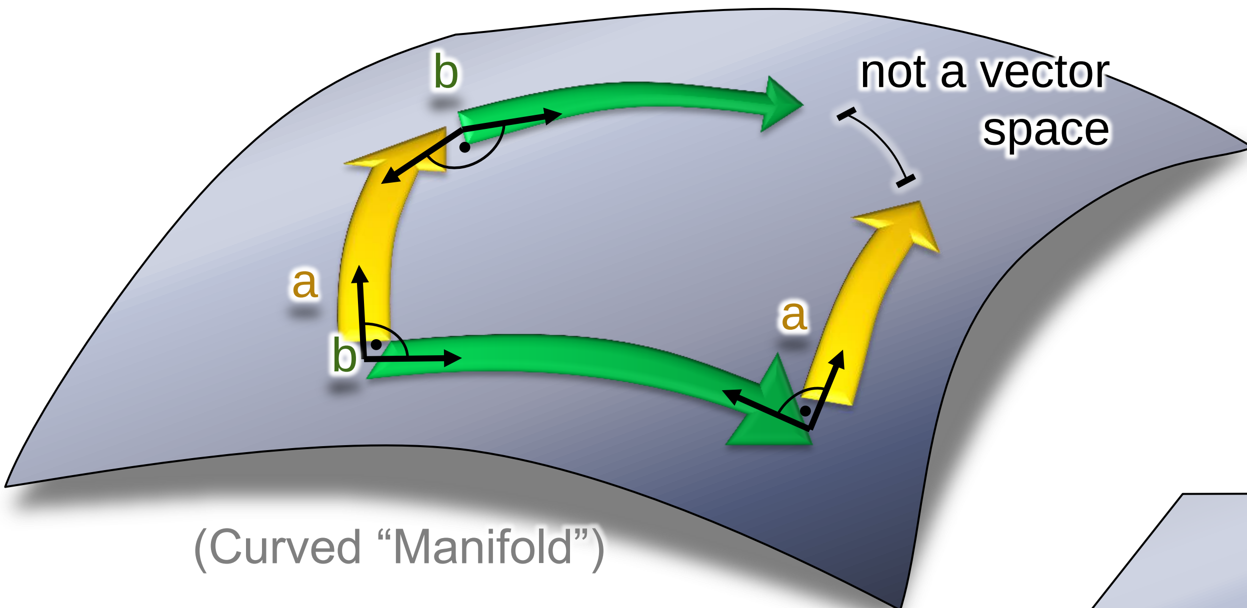
Commutative

$$\mathbf{v} + \mathbf{w} = \mathbf{w} + \mathbf{v}$$

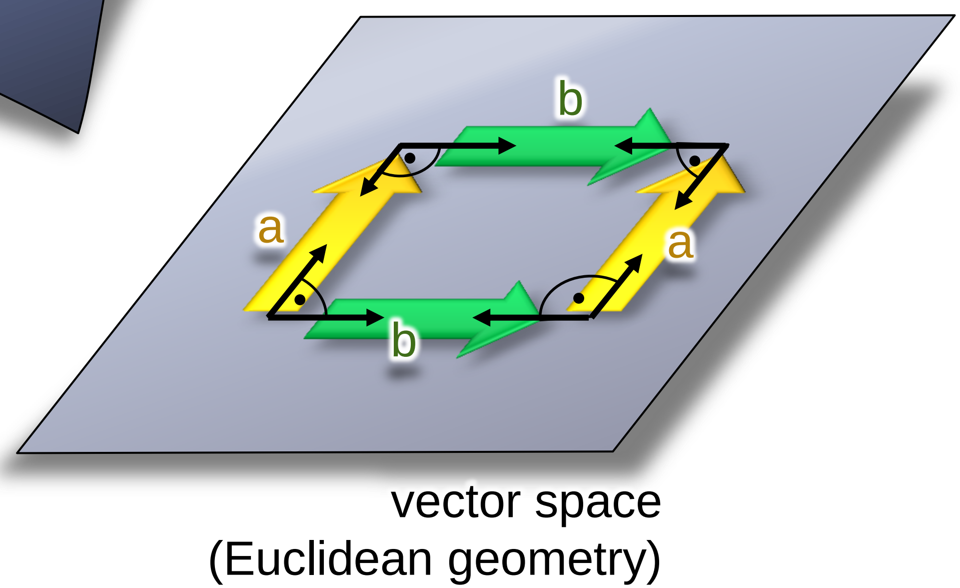


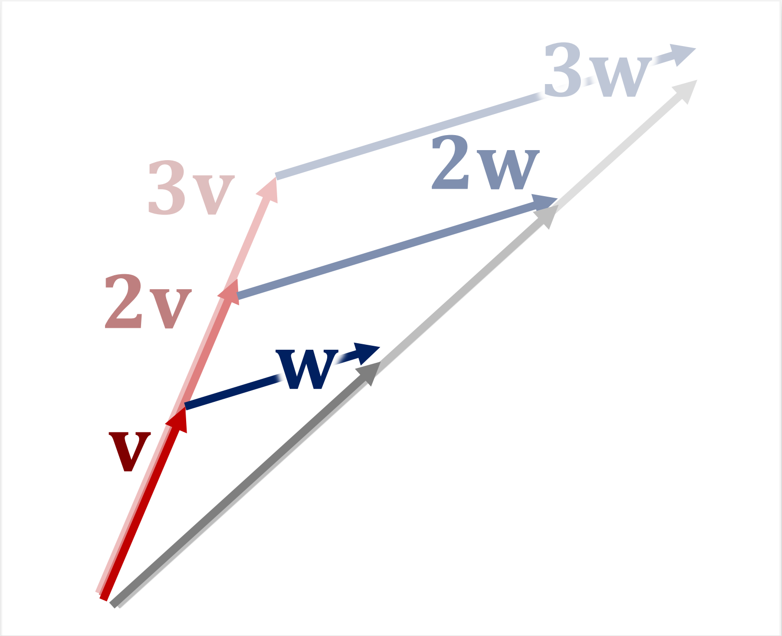
Associative

$$\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$$



*) associativity still holds

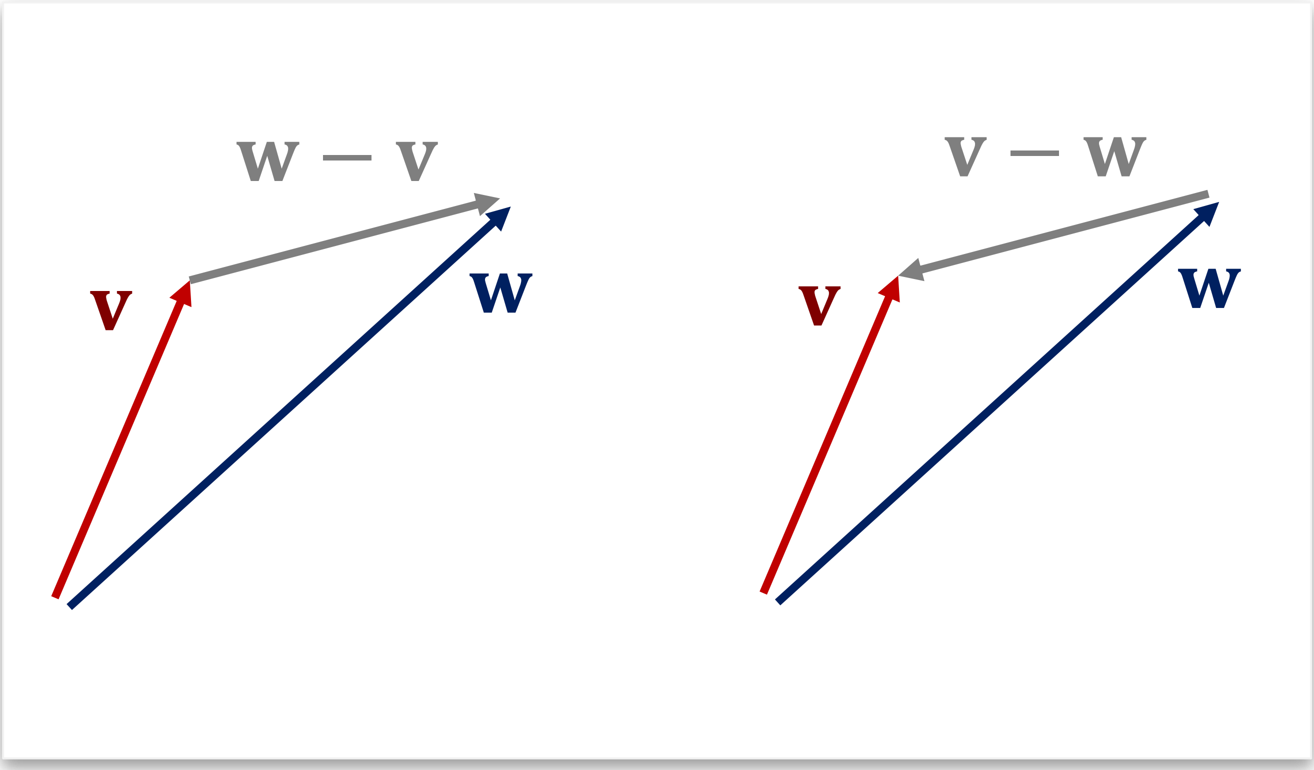




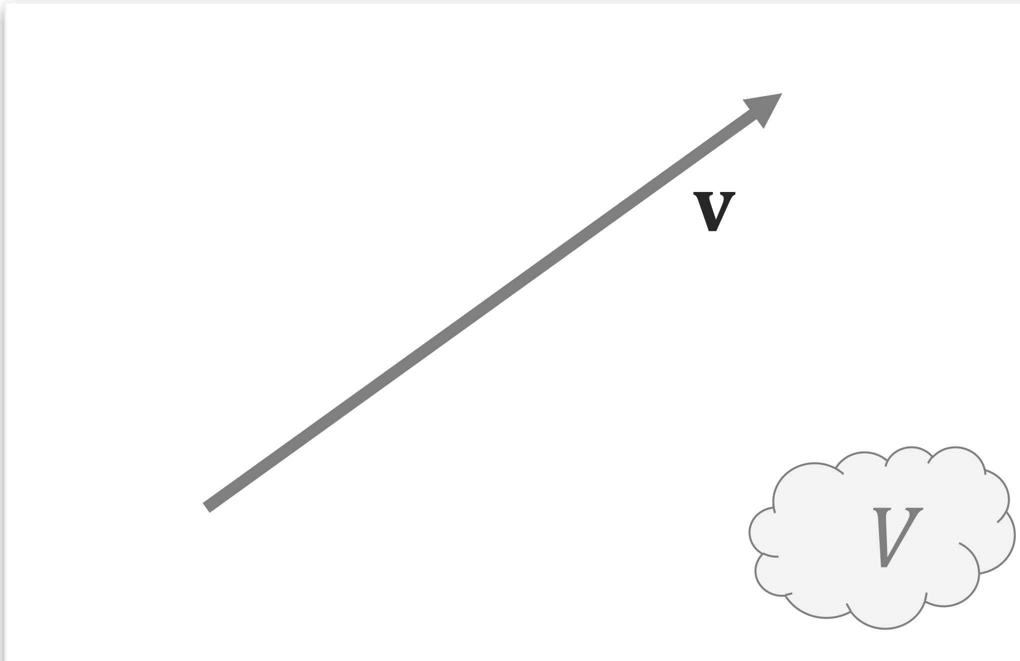
Distributive

$$\lambda(\mathbf{v} + \mathbf{w}) = \lambda\mathbf{v} + \lambda\mathbf{w}$$

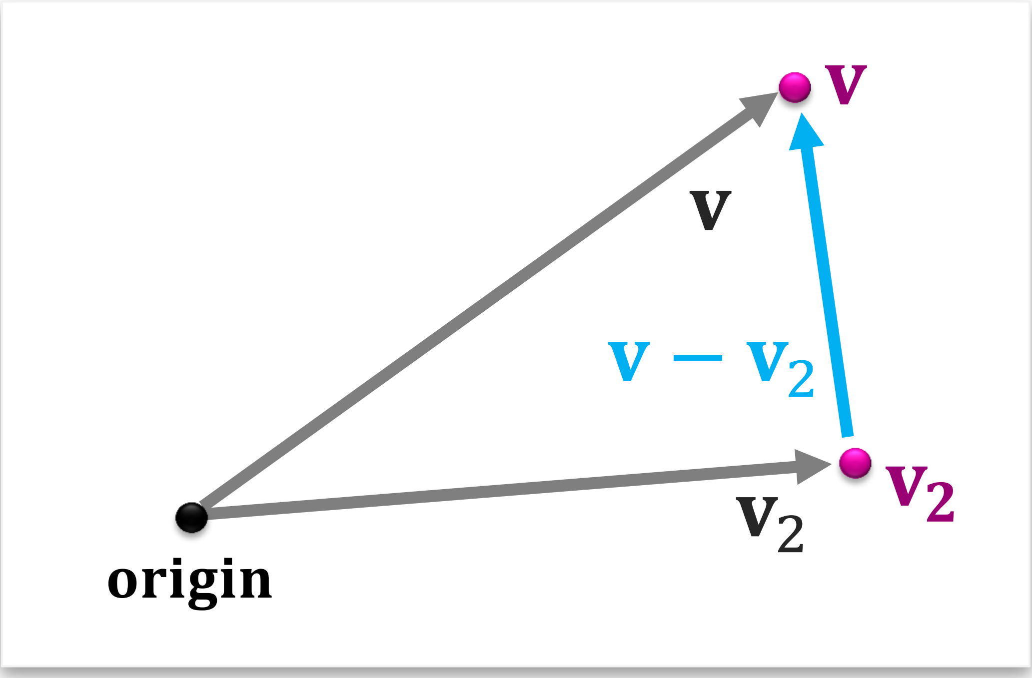
$$\lambda(\mu\mathbf{v}) = \lambda\mu(\mathbf{v})$$



$$\mathbf{v} - \mathbf{w} = -(\mathbf{w} - \mathbf{v})$$

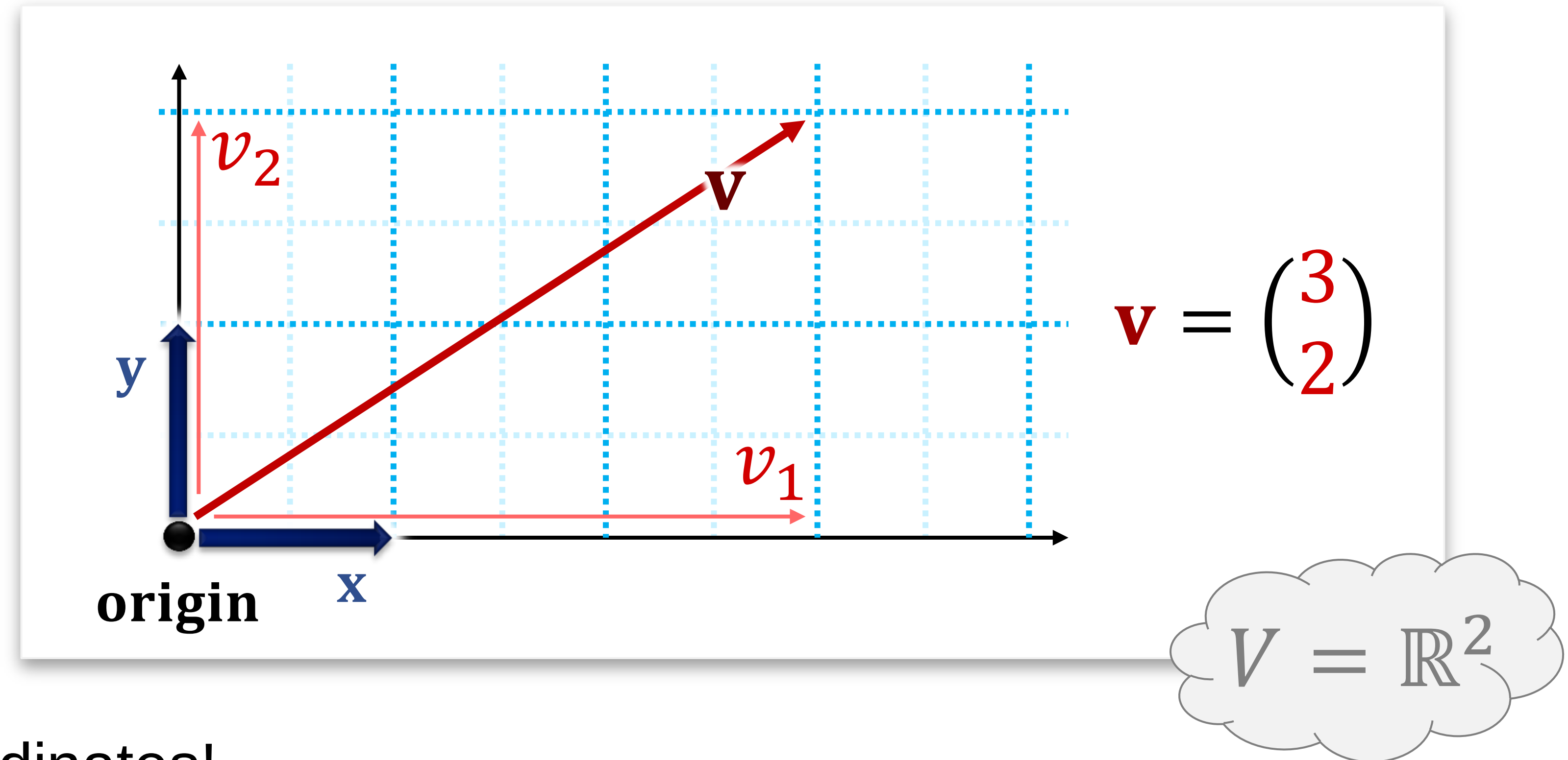


- Vectors
  - A vector is an arrow in space



- Points
  - Fix an origin
  - Store vector from origin to point
  - „Vectors are differences of points“

Algebraic Representation
(Implementation)

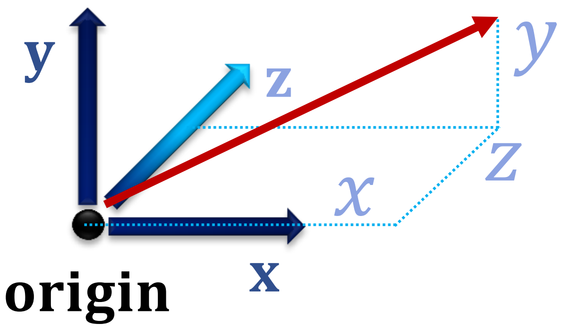


Coordinates!

$$\mathbf{v} = \begin{pmatrix} x\text{–coord.} \\ y\text{–coord.} \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

Project on coordinate vectors

- We can add more entries:

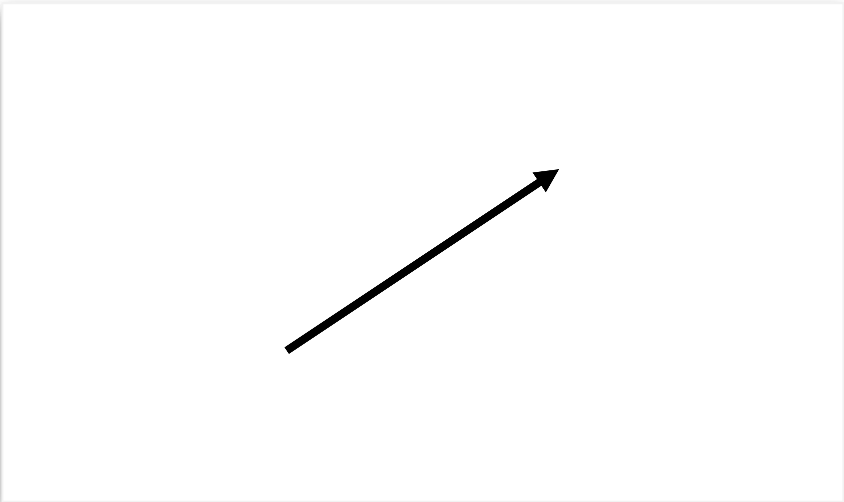


$$\mathbf{v} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \qquad V = \mathbb{R}^3$$

- Or even more entries:

$$d = \text{“}dimension\text{”}$$

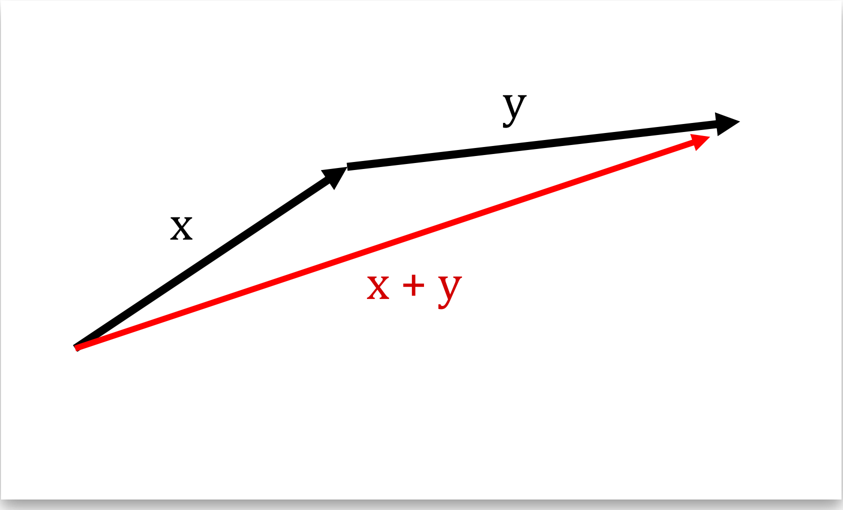
$$\mathbf{v} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_d \end{pmatrix} \qquad V = \mathbb{R}^\mathrm{d}$$



Geometry:
vectors are arrows in space

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

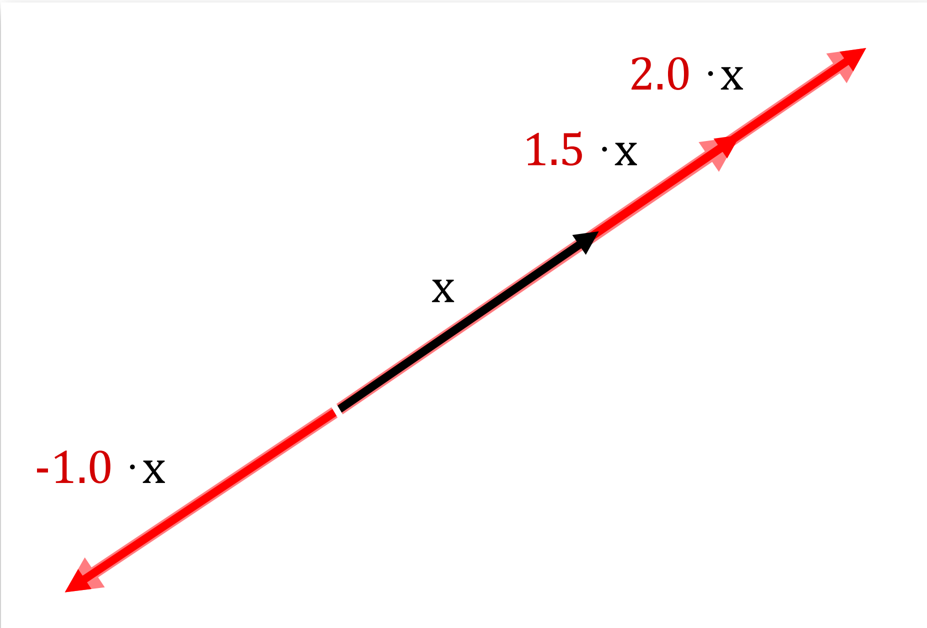
Algebra:
arrays of numbers



$$\mathbf{x} + \mathbf{y} = \begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \end{pmatrix}$$

Adding Vectors:
Concatenation

Algebra:
adding numbers



$$\lambda \mathbf{x} = \begin{pmatrix} \lambda x_1 \\ \lambda x_2 \end{pmatrix}$$

Scalar-Vector Multiplication:
Scaling (incl. mirroring)

Algebra:
multiplying with real number

- Null vector

$$\mathbf{0} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

  - Does not change other vectors in addition
  - $\mathbf{v} + \mathbf{0} = \mathbf{0} + \mathbf{v} = \mathbf{v}$ for all vectors $\mathbf{v}$

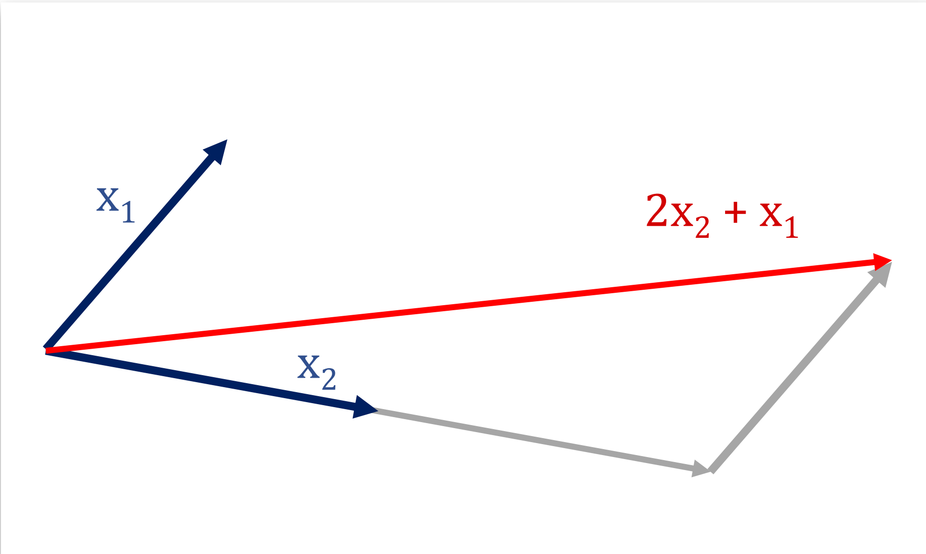
- Definition: A *real vector space* of dimension d
  - The set of all d-tupels:

$$\underbrace{\mathbb{R} \times \cdots \times \mathbb{R}}_{d-\text{times}} = \mathbb{R}^d \qquad \mathbf{x} \in \mathbb{R}^d : \mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix}$$

  - With two operations

$$\mathbf{x} + \mathbf{y} := \begin{pmatrix} x_1 + y_1 \\ \vdots \\ x_d + y_d \end{pmatrix} \qquad \lambda \cdot \mathbf{x} = \lambda \mathbf{x} := \begin{pmatrix} \lambda x_1 \\ \vdots \\ \lambda x_d \end{pmatrix}$$

$$\mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \quad \lambda \in \mathbb{R}$$



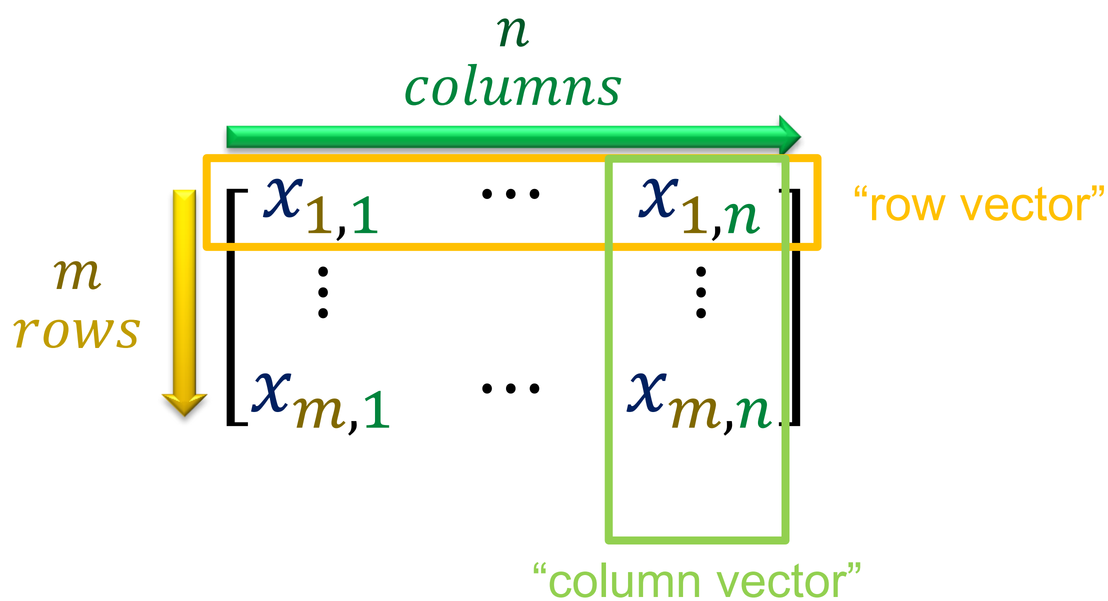
Geometrically

$$\mathbf{p} = \sum_{i=1}^{n} \lambda_i \mathbf{x}_i$$

Algebraically

The concept of ***linear combinations*** is the corner stone of graphics and visualization.

Linear Combinations & Matrices

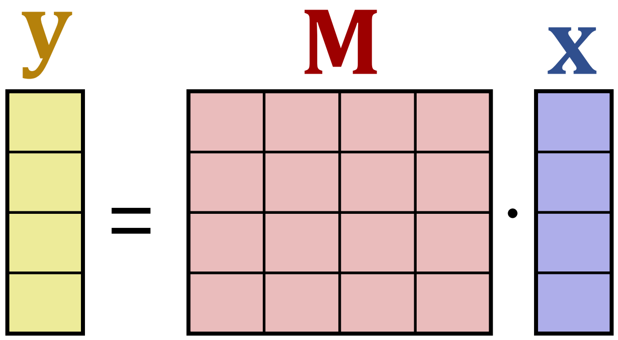


- Matrix elements

$$x_{row,column}$$

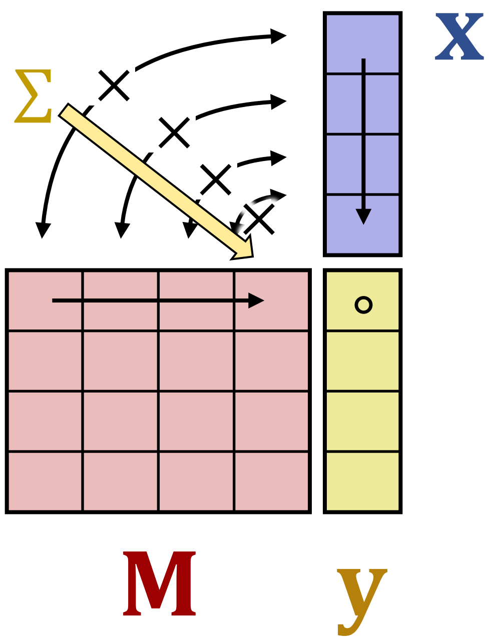
- Row first, then column
  - *“y”*-coordinate of the array first (common convention)

- Algebraic rule:
  - Vector-matrix product:

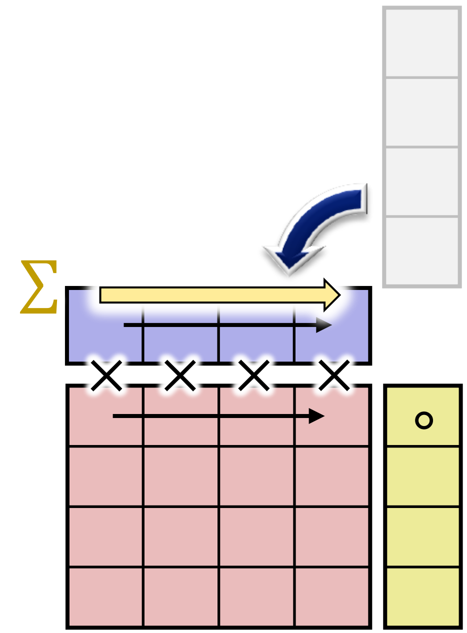


$$\mathbf{y} = \mathbf{M} \cdot \mathbf{x}$$

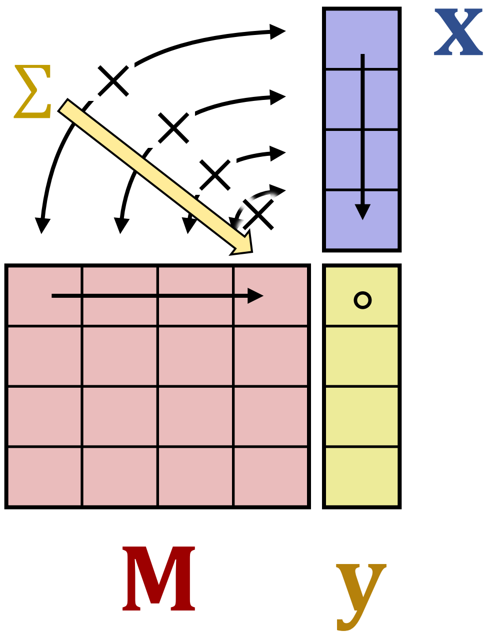
- Algebraic rule:
  - Vector-matrix product:



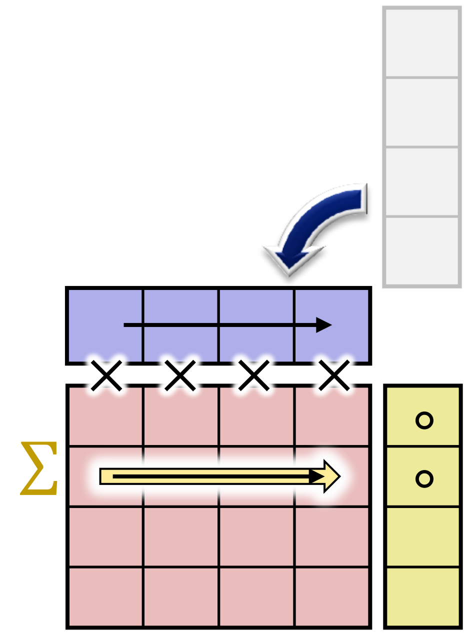
$$\mathbf{y} = \mathbf{M} \cdot \mathbf{x}$$



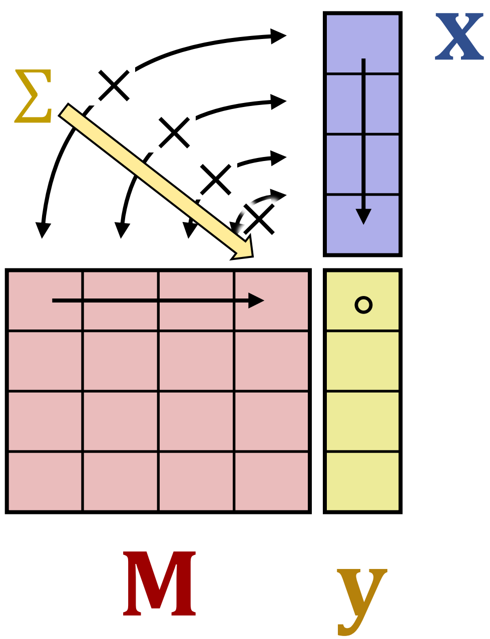
- Algebraic rule:
  - Vector-matrix product:



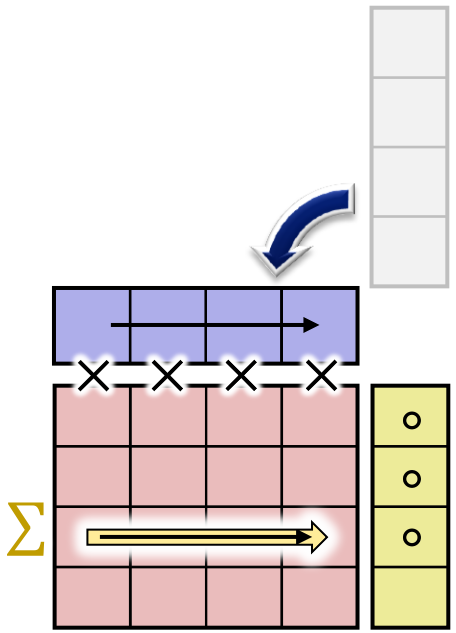
$$\mathbf{y} = \mathbf{M} \cdot \mathbf{x}$$



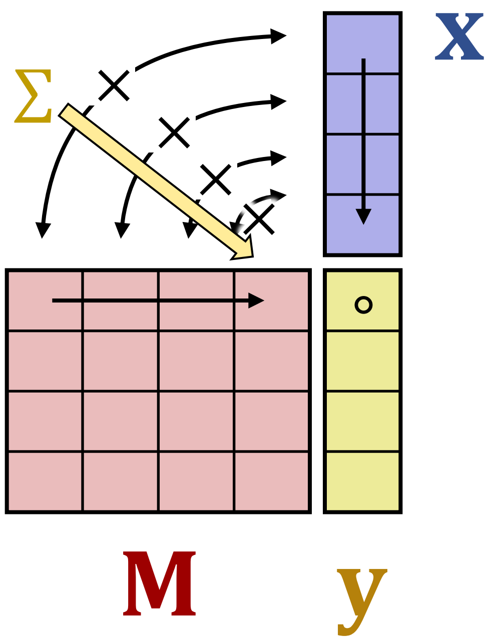
- Algebraic rule:
  - Vector-matrix product:



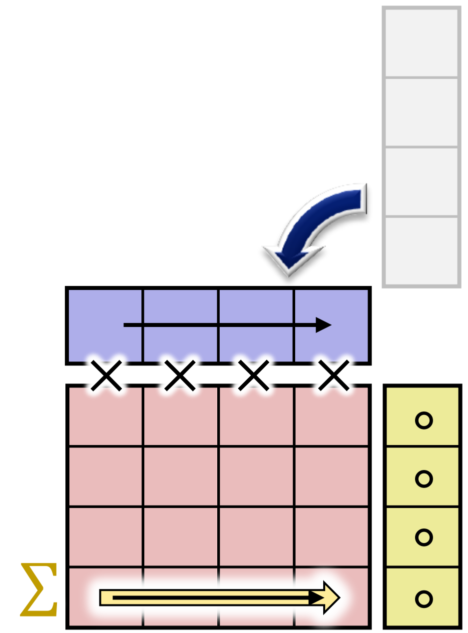
$$\mathbf{y} = \mathbf{M} \cdot \mathbf{x}$$



- Algebraic rule:
  - Vector-matrix product:



$$\mathbf{y} = \mathbf{M} \cdot \mathbf{x}$$



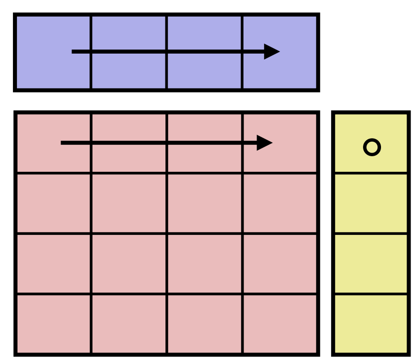
- Matrix-Vector Multiplication

column vectors of the matrix

Linear Combination

$$\begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix} := \sum_{i=1}^{n} \lambda_i \begin{bmatrix} x_{1,i} \\ \vdots \\ x_{m,i} \end{bmatrix} = \sum_{i=1}^{n} \lambda_i \mathbf{x}_i$$

$$= \begin{bmatrix} \lambda_1 \cdot x_{1,1} + \cdots + \lambda_n \cdot x_{1,n} \\ \vdots \\ \lambda_1 \cdot x_{m,1} + \cdots + \lambda_n \cdot x_{m,n} \end{bmatrix}$$



Standard Transformations

- Translate a point **p** along a vector **t**

- General case:

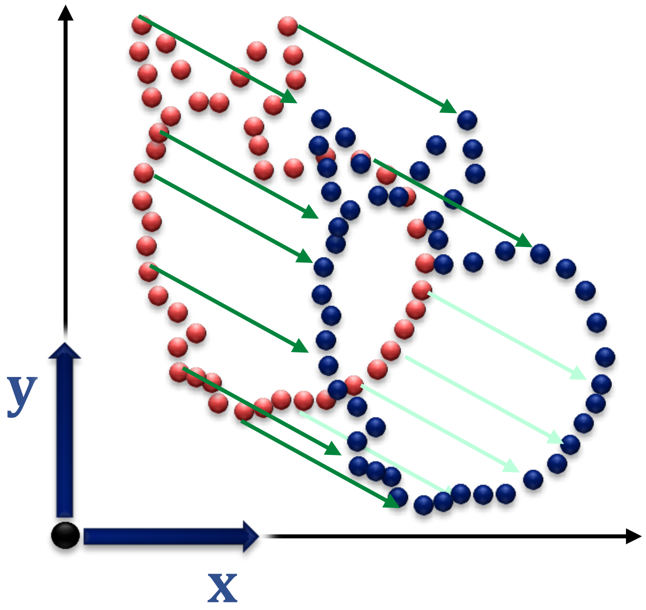
$$\mathbf{p}' = \mathbf{p} + \mathbf{t}$$

- 2D:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \end{bmatrix}$$

- 3D:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \\ z + t_z \end{bmatrix}$$



- Scale a point $\mathbf{p}$ in each dimension by the factors s_x, s_y, s_z

- General case:

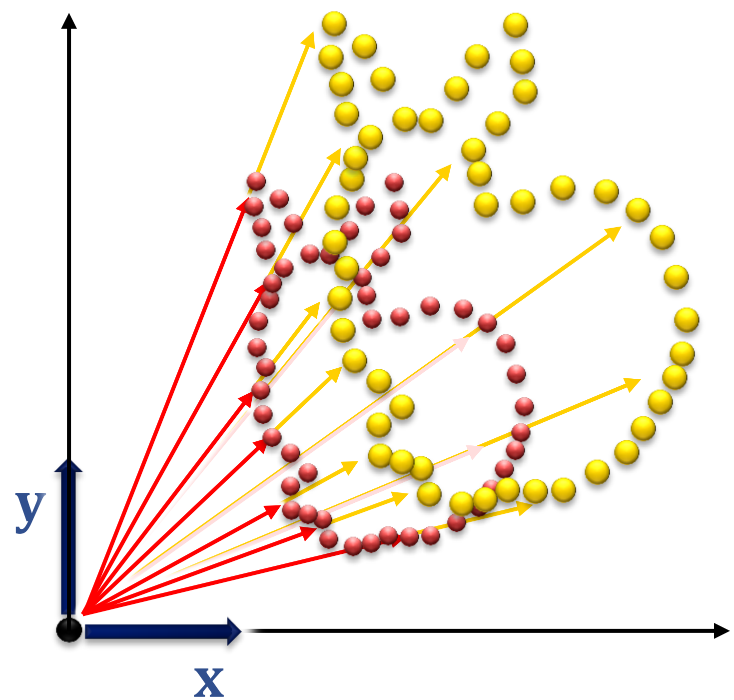
$$\mathbf{p}' = \mathbf{S} \cdot \mathbf{p} \qquad \mathbf{S}: \mathbb{R}^n \to \mathbb{R}^n, \qquad \mathbf{S} = \begin{bmatrix} s_1 & 0 & \cdots & 0 \\ 0 & s_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & s_n \end{bmatrix}$$

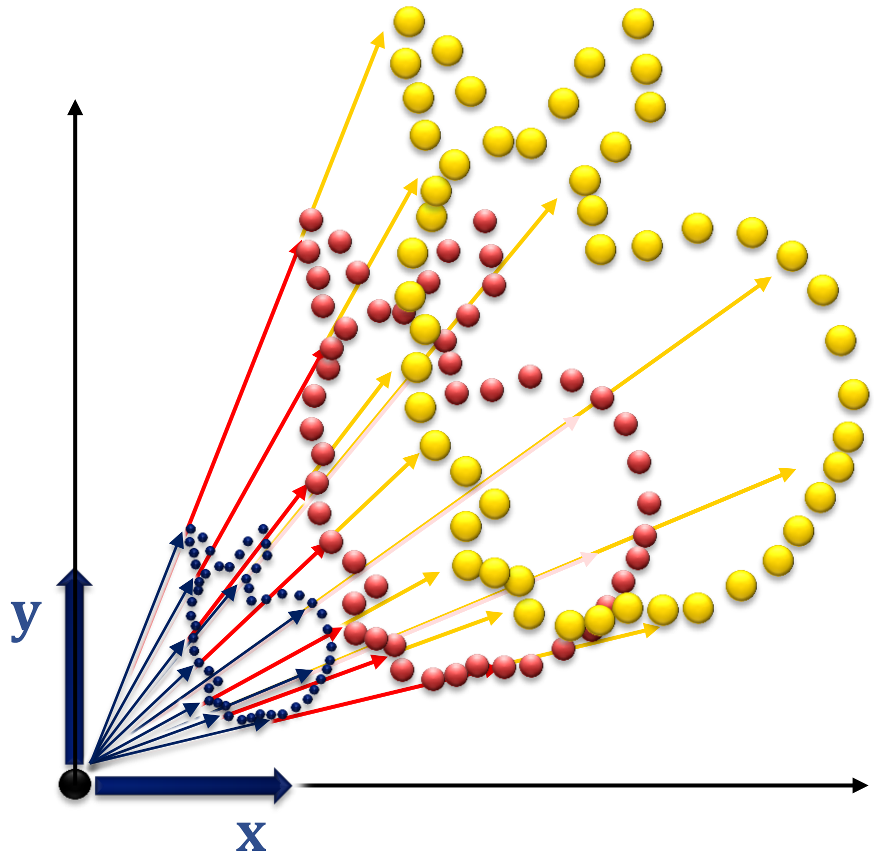
- 2D:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

- 3D:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

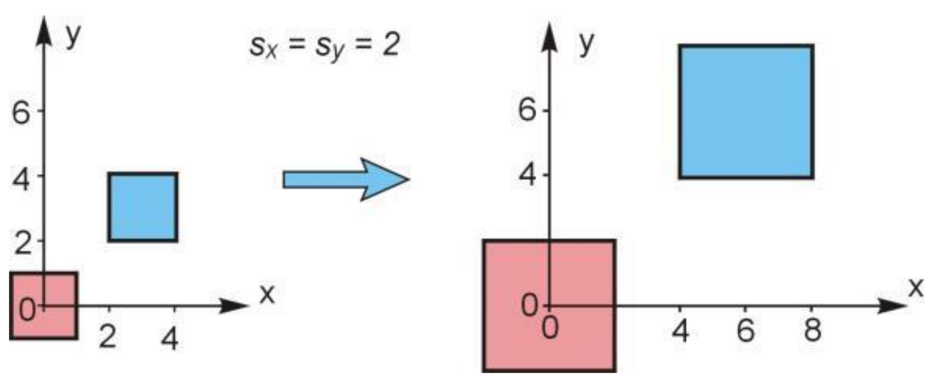


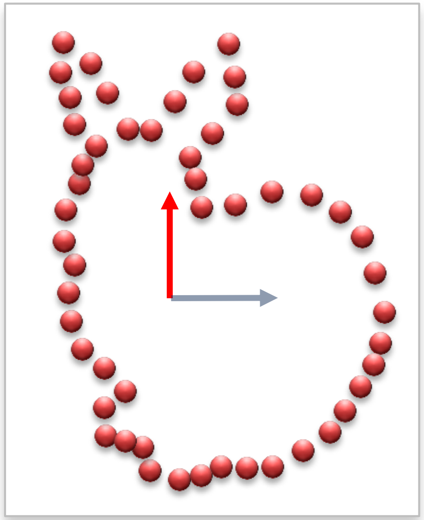


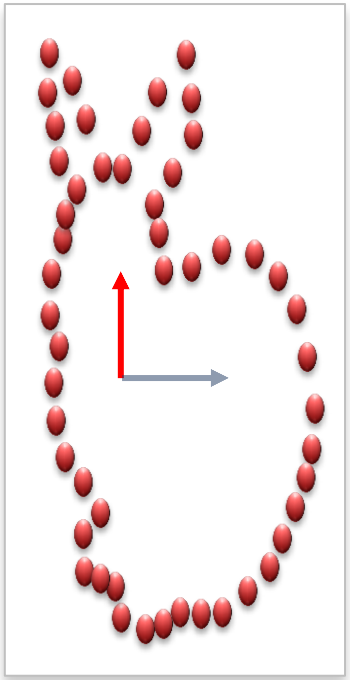
Making something uniformly smaller:
$s_x = s_y = s_z < 1$

Making something uniformly bigger:
$s_x = s_y = s_z > 1$

Note:
Center is at the origin





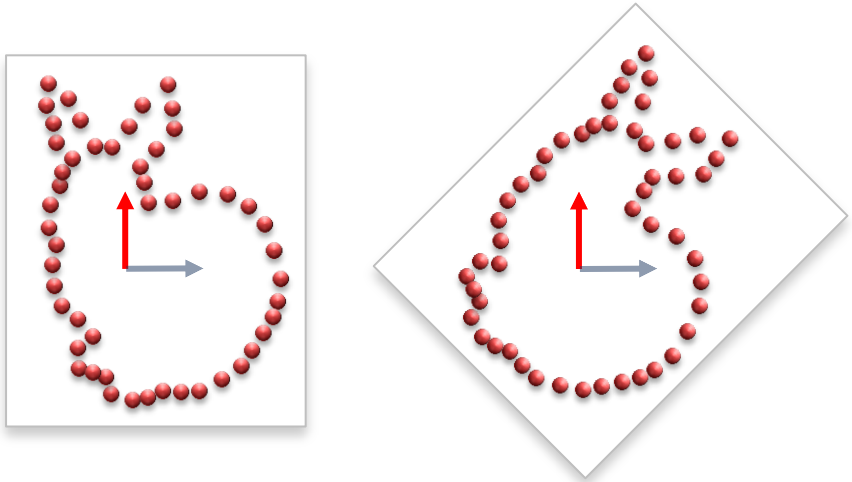


$$s_x \neq s_y \neq s_z$$

- Rotate a point **p** around the origin with an angle ***α*** in counter-clockwise direction

- 2D:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



$$x' = r * \cos(\alpha + \phi)$$

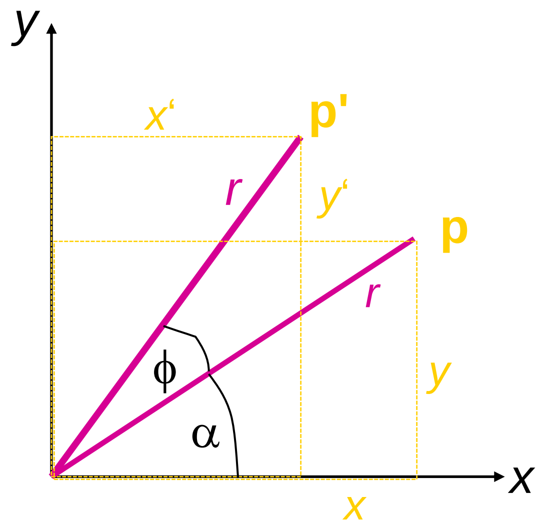
$$x' = \underline{r * \cos\alpha} * \cos\phi - \underline{r * \sin\alpha} * \sin\phi$$

$$x' = x * \cos\phi - y * \sin\phi$$

$$y' = r * \sin(\alpha + \phi)$$

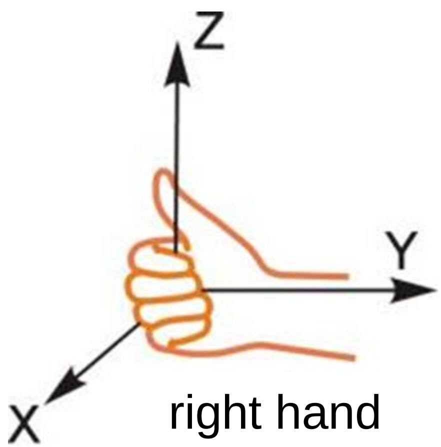
$$y' = \underline{r * \cos\alpha} * \sin\phi + \underline{r * \sin\alpha} * \cos\phi$$

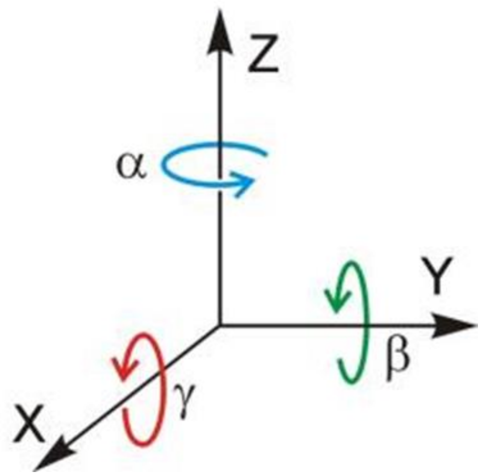
$$y' = x * \sin\phi + y * \cos\phi$$



Remark: The α from this slide is not the α from the previous slide!

- Rotate a point $\mathbf{p}$ around a rotation axis with an angle α in counter-clockwise direction





- Rotation matrices for the rotation around the coordinate axes:

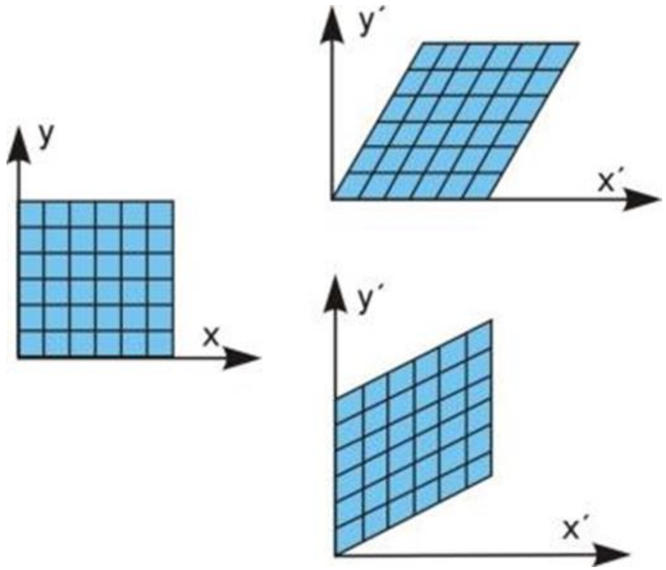
$$\mathbf{R}_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix}$$

$$\mathbf{R}_y = \begin{pmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{pmatrix}$$

$$\mathbf{R}_z = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- A shear is given as
- 2D:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & s_y \\ s_x & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & s_y \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

shear in x-direction

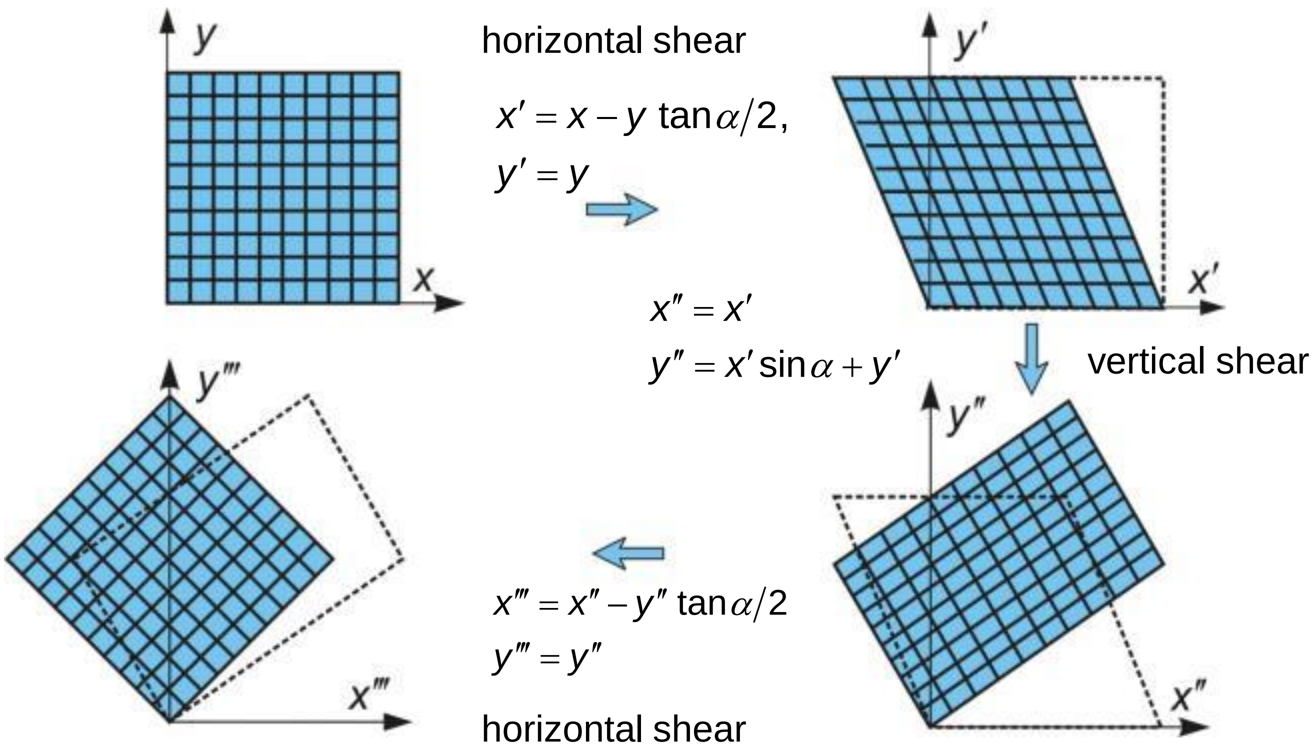
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ s_x & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

shear in y-direction

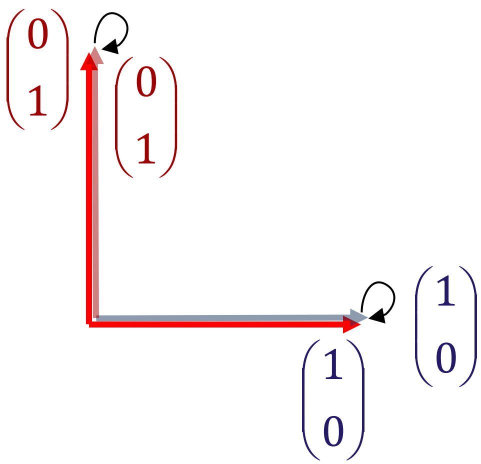
- A shear is given as
- 3D:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} 1 & s_{yx} & s_{zx} \\ s_{xy} & 1 & s_{zy} \\ s_{xz} & s_{yz} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

- Shears can be used to describe rotations
- Example: Rotation of 2D objects using three subsequent shear transformations $\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & -\tan\alpha/2 \\ 0 & 1 \end{pmatrix} \bullet \begin{pmatrix} 1 & 0 \\ \sin\alpha & 1 \end{pmatrix} \bullet \begin{pmatrix} 1 & -\tan\alpha/2 \\ 0 & 1 \end{pmatrix} \bullet \begin{pmatrix} x \\ y \end{pmatrix}$



- The ***Identity matrix*** keeps points in their original location.



$$\mathbf{M}_{identity} = \mathbf{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

General case

$$\mathbf{I}: \mathbb{R}^n \rightarrow \mathbb{R}^n, \qquad \mathbf{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

Homogeneous Coordinates (short version)

- Translations are not linear
  - $\mathbf{x} \to \mathbf{M}\mathbf{x}$ cannot encode translations
  - **Proof:** Origin cannot be moved:

$$\mathbf{M} \cdot \mathbf{0} = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

- Solution: Just add a constant one
  - Increase dimension $\mathbb{R}^d \to \mathbb{R}^{d+1}$
  - Last entry = 1 in vectors
    - “Cheap Trick”, “Evil Hack”

$$\mathbf{M}' \cdot \mathbf{x} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & t_1 \\ m_{21} & m_{22} & m_{23} & t_2 \\ m_{31} & m_{32} & m_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} \ddots & & \iddots & | \\ & \mathbf{M} & & \mathbf{t} \\ \iddots & & \ddots & | \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} | \\ \mathbf{x} \\ | \\ 1 \end{pmatrix} = \begin{pmatrix} | \\ \mathbf{Mx} + \mathbf{t} \\ | \\ 1 \end{pmatrix}$$

- General case

$$\mathbf{M} \cdot \mathbf{x} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x' \\ y' \\ z' \\ w' \end{pmatrix}$$

  - w' might be different from 1
  - Convention: Divide by w-coord. before using

$$\text{Result: } \begin{pmatrix} x'/w' \\ y'/w' \\ z'/w' \\ 1 \end{pmatrix}$$

- General case

$$\mathbf{M} \cdot \mathbf{x} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} \equiv \begin{pmatrix} y_1/y_4 \\ y_2/y_4 \\ y_3/y_4 \\ 1 \end{pmatrix}$$

- Rules:
  - Before using as 3D point, divide by last (4th) entry
  - No normalization required during subsequent transformations (matrix-multiplications, see later)

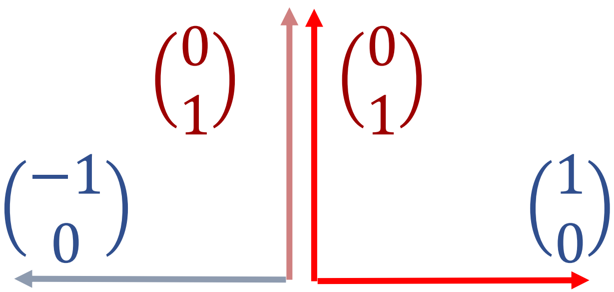
- Projective Geometry
  - Not just an evil hack
  - Deep & interesting theoretical background
  - More on this later
- For simplicity
  - We’ll treat it as a computational trick for now
    - Focus on the graphics application
  - Remember for now:
    - We can build “4D Translation matrices” for 3D+1 points
    - We can “divide” by a common linear factor

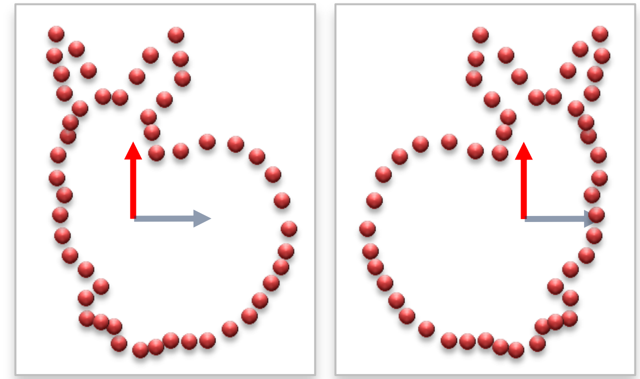
Overview Standard Transformations with Homogeneous Coordinates

	Translation	Scaling	Shearing
2D	$\mathbf{T}(t_x, t_y) = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix}$	$\mathbf{S}(s_x, s_y) = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\mathbf{H}_x = \begin{pmatrix} 1 & h_y & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
3D	$\mathbf{T}(t_x, t_y, t_z) = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\mathbf{S}(s_x, s_y, s_z) = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\mathbf{H} = \begin{pmatrix} 1 & s_1 & s_2 & 0 \\ s_3 & 1 & s_4 & 0 \\ s_5 & s_6 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

2D-Rotation	3D-Rotation	
$\mathbf{R}(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix}$	Rotation around x-axis	$\mathbf{R}_x(\phi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
	Rotation around y-axis	$\mathbf{R}_y(\phi) = \begin{pmatrix} \cos\phi & 0 & \sin\phi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
	Rotation around z-axis	$\mathbf{R}_z(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Further Transformations





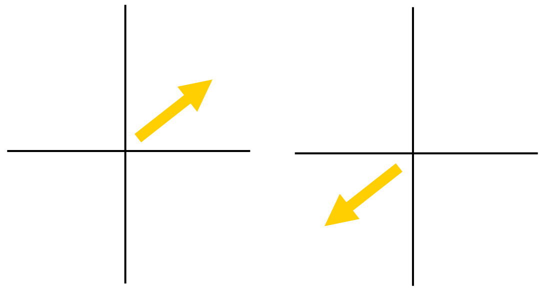
$$\mathbf{M}_{refl} = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

Reflection Axis

General case

$$\mathbf{S}_\lambda : \mathbb{R}^n \rightarrow \mathbb{R}^n, \qquad \mathbf{S}_\lambda = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & -1 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

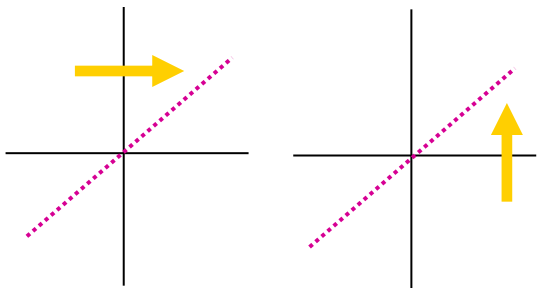
$$\begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$



before *after*

reflection over the origin

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



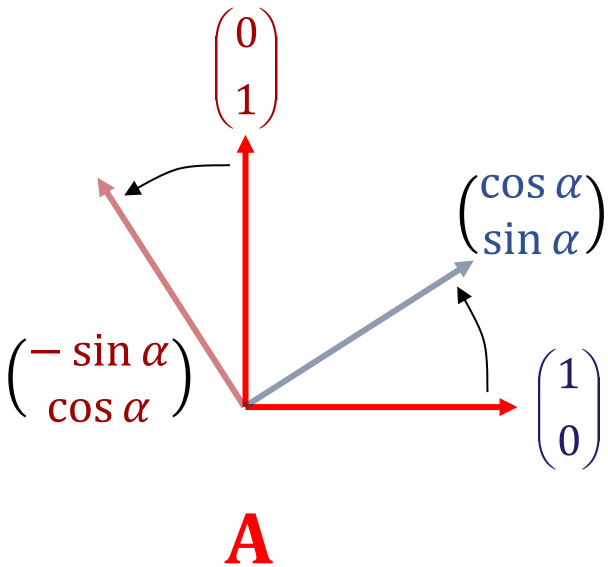
before *after*

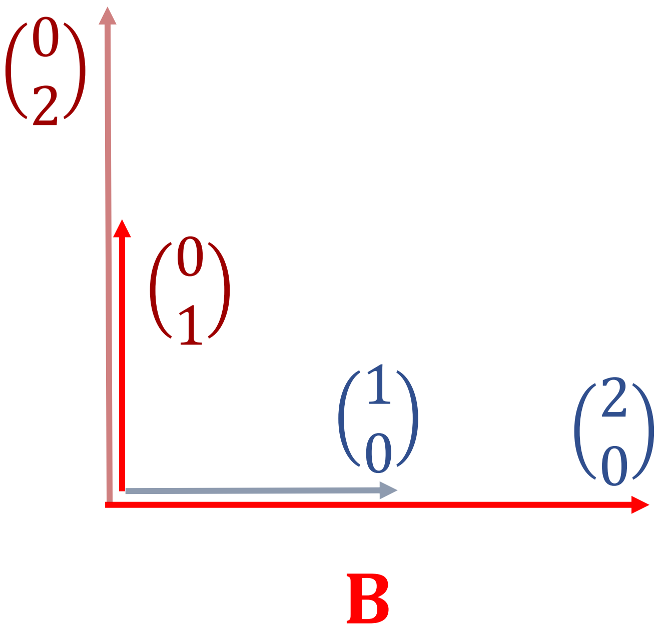
reflection at the line y=x

Combining Transformations

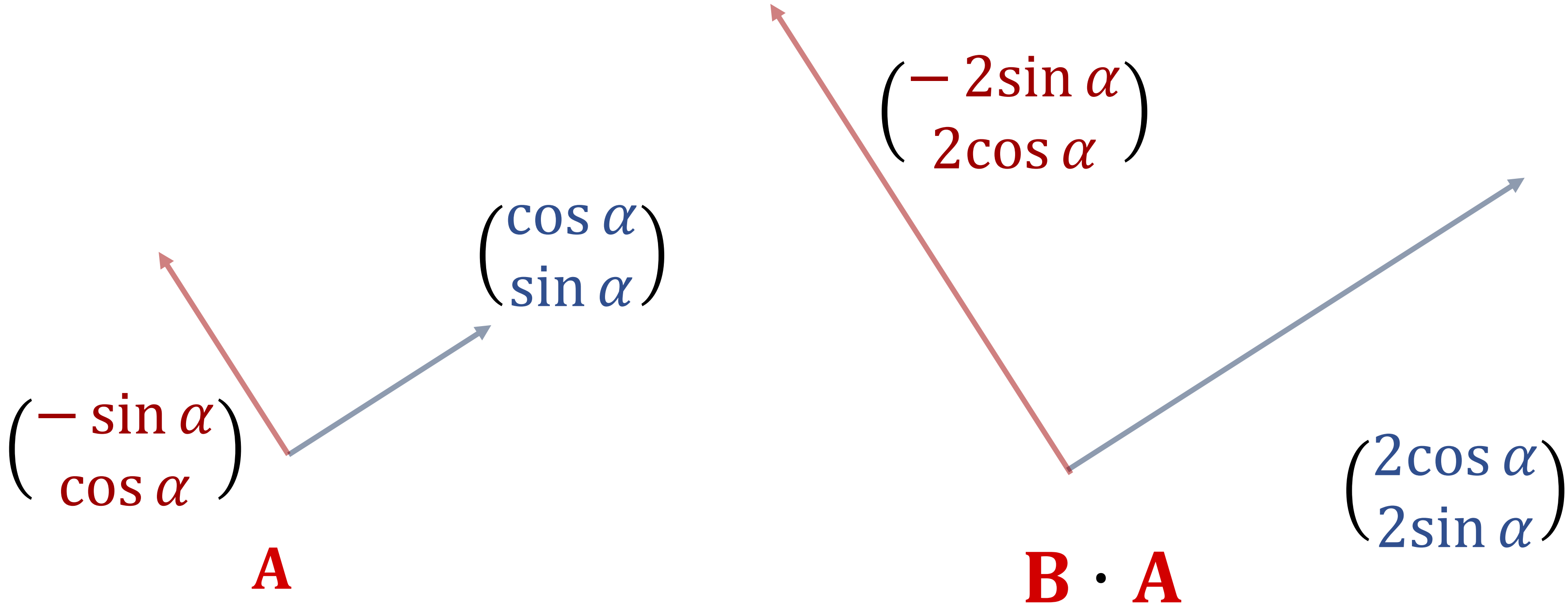
- You can combine all of these
- Example: General axis of rotation
  - First rotate rotation axis to x-axis
  - Rotate around x
  - Rotate back
- Question
  - How to combine multiple transformation matrices?

- Execute multiple transformations, one after another
  - Written as product: matrix multiplication
  - $(\mathbf{B} \cdot \mathbf{A}) \cdot \mathbf{x}$:
    - Apply $\mathbf{A}$ to $\mathbf{x}$ first
    - Then $\mathbf{B}$
    - $(\mathbf{B} \cdot \mathbf{A})$ is again a matrix



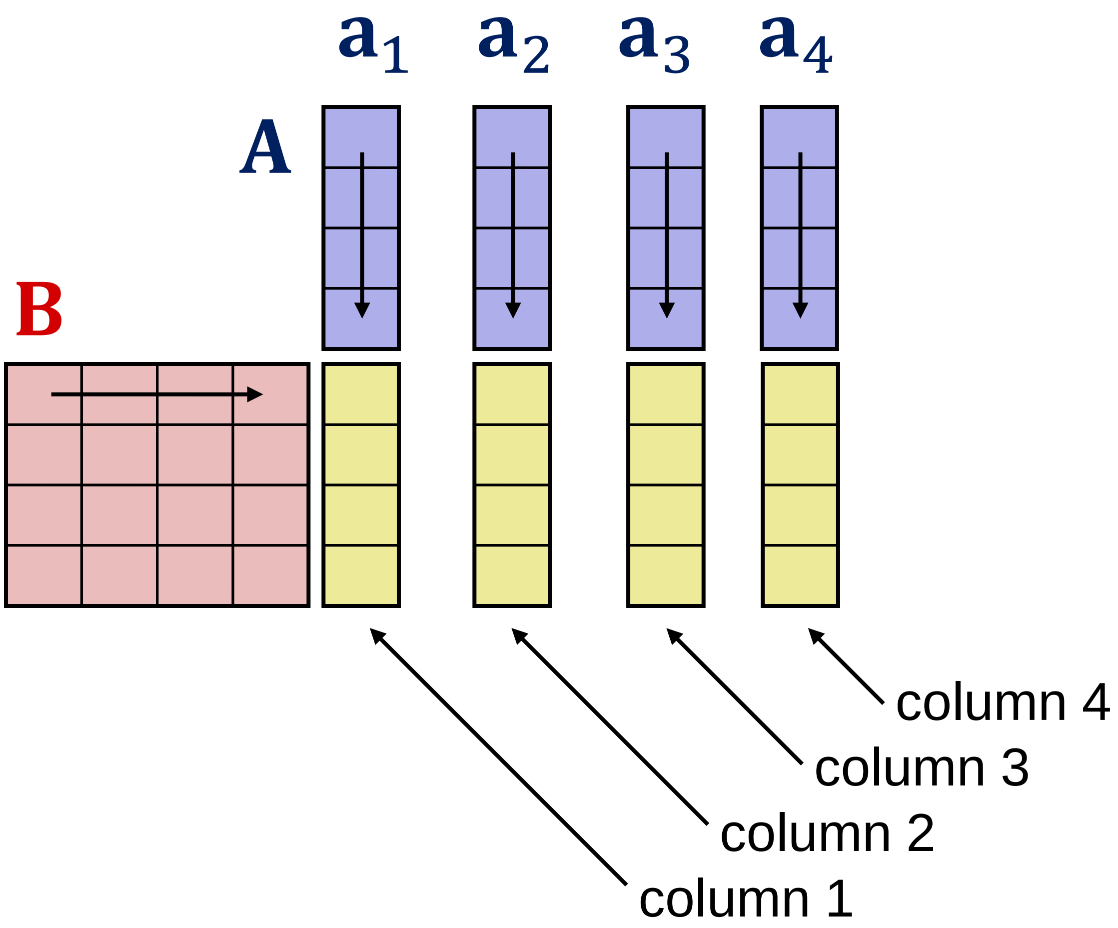


- Consider (**B** · **A**):
  - Rotate first (**A**)
  - Then scale (**B**)

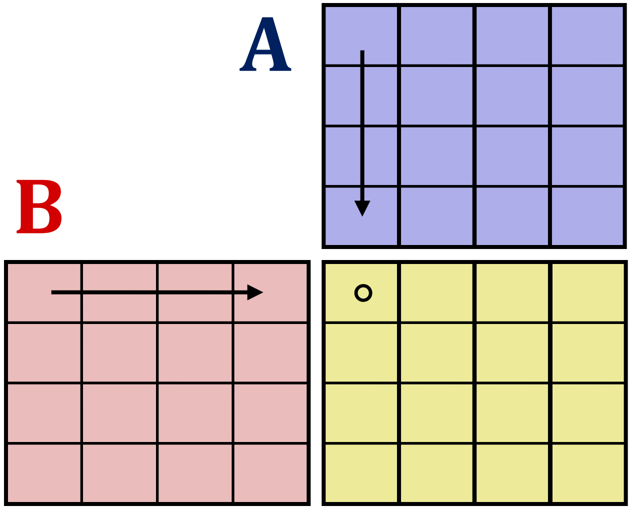


- How to compute $(\mathbf{B} \cdot \mathbf{A})$?
  - Transform basis vectors
  - Transform again

- Matrix product:



- Matrix product:

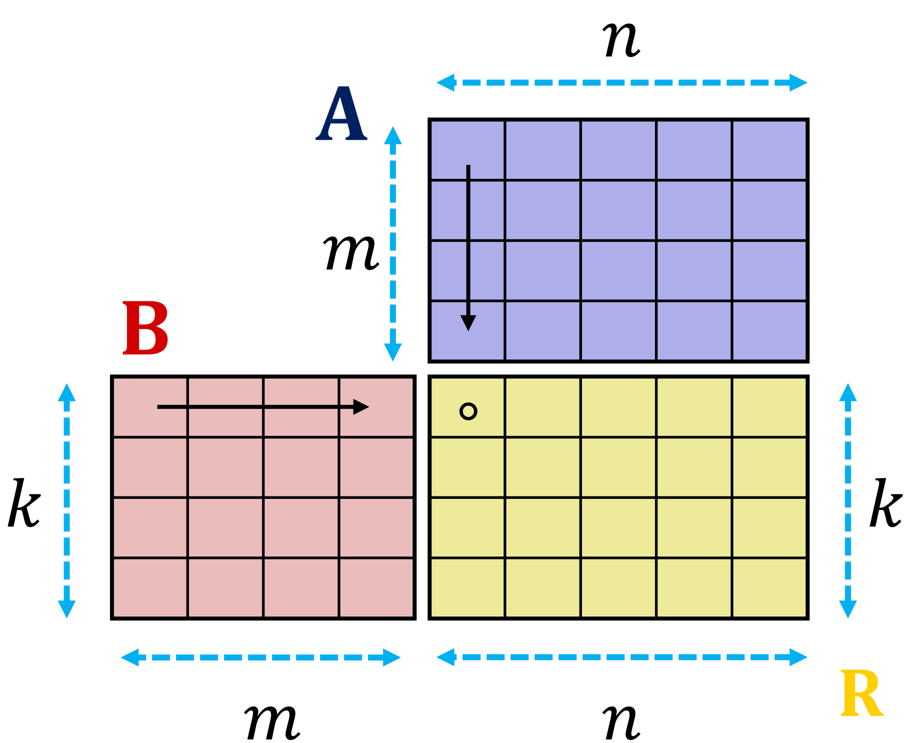


- General matrix products:
  - **B** · **A**: possible if #Row(**A**) = #Columns(**B**)

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & & \vdots \\ a_{m,1} & \cdots & a_{m,n} \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} b_{1,1} & \cdots & b_{1,m} \\ \vdots & & \vdots \\ b_{k,1} & \cdots & b_{k,m} \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} r_{1,1} & \cdots & r_{1,n} \\ \vdots & & \vdots \\ r_{k,1} & \cdots & r_{k,n} \end{bmatrix}$$



$$\mathbf{R} = \mathbf{B} \cdot \mathbf{A}$$

$$r_{i,j} = \sum_{q=1}^{m} a_{q,j} \cdot b_{i,q}$$

- Matrix-Multiplication
  - Associative

$$(\mathbf{A} \cdot \mathbf{B}) \cdot \mathbf{C} = \mathbf{A} \cdot (\mathbf{B} \cdot \mathbf{C})$$

  - Includes vector-multiplication

$$(\mathbf{A} \cdot \mathbf{B}) \cdot \mathbf{v} = \mathbf{A} \cdot (\mathbf{B} \cdot \mathbf{v})$$

  - In general, not commutative:

It might be that $\mathbf{A} \cdot \mathbf{B} \neq \mathbf{B} \cdot \mathbf{A}$

  - Linear

$$\mathbf{A} \cdot (\mathbf{v} + \mathbf{w}) = \mathbf{A} \cdot \mathbf{v} + \mathbf{A} \cdot \mathbf{w}$$
$$\mathbf{A} \cdot (\lambda \cdot \mathbf{v}) = \lambda \cdot (\mathbf{A} \cdot \mathbf{v})$$

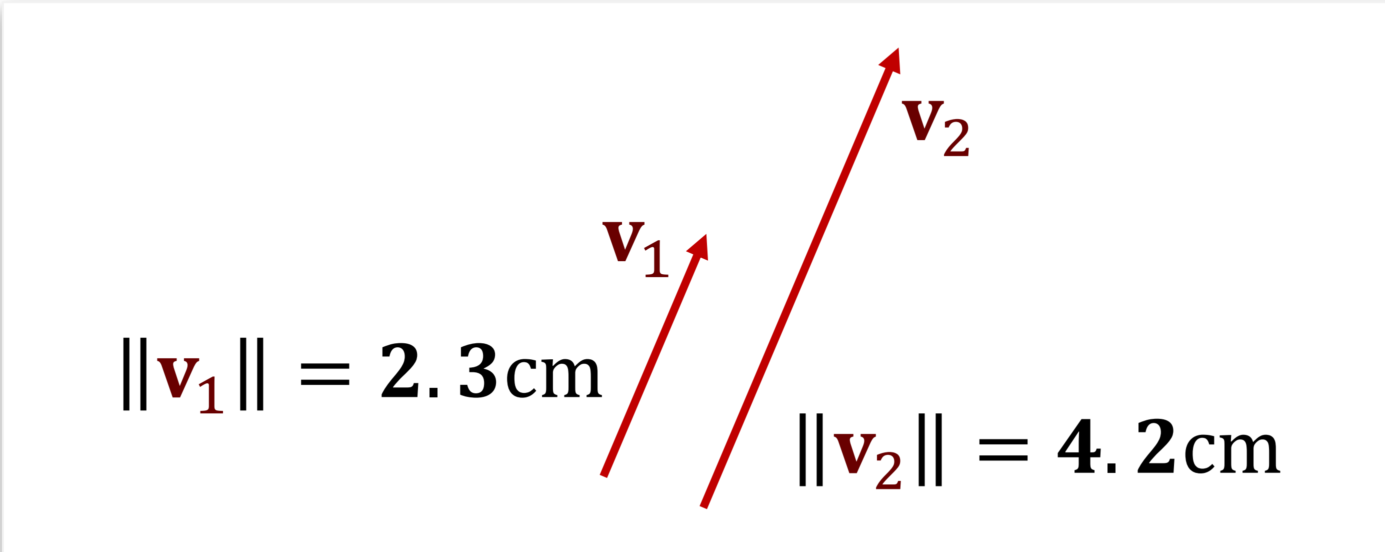
Settings

$\lambda \in \mathbb{R}$

$\mathbf{A}, \mathbf{B}, \mathbf{C}$ - matrices

$\mathbf{v}, \mathbf{w}$ - vectors

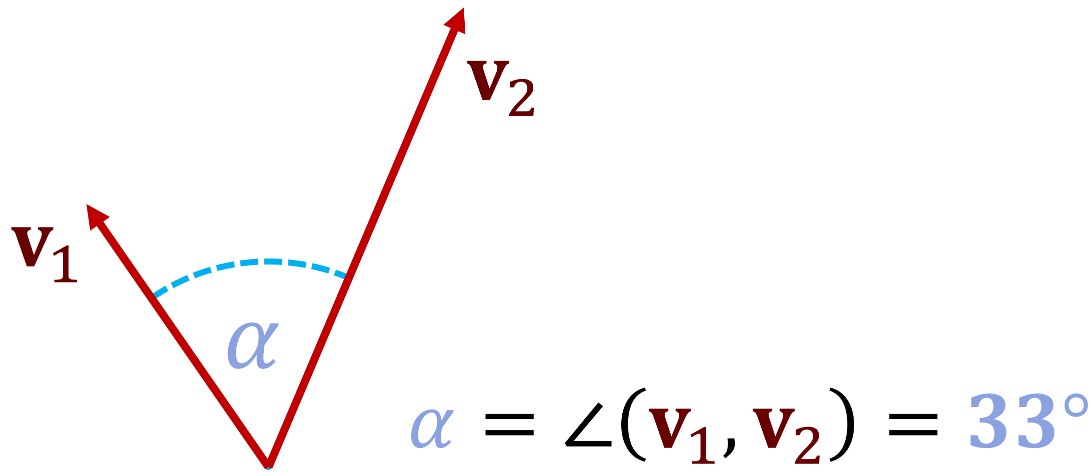
More Vector Operations: Scalar Products



Length of Vectors

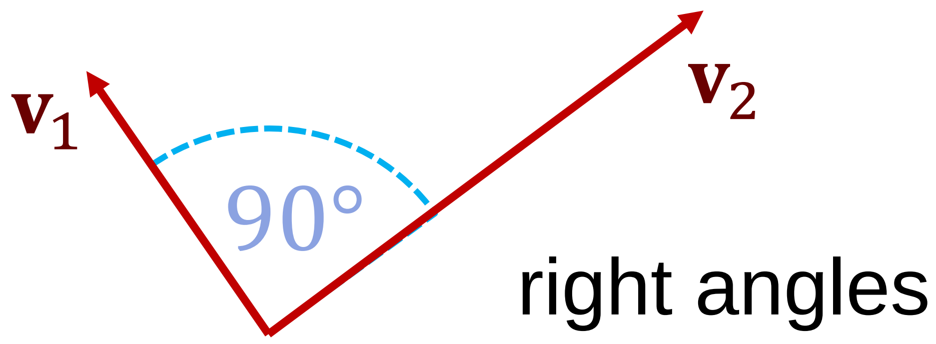
"length" or "norm"

$||\mathbf{v}||$ yields real number ≥ 0

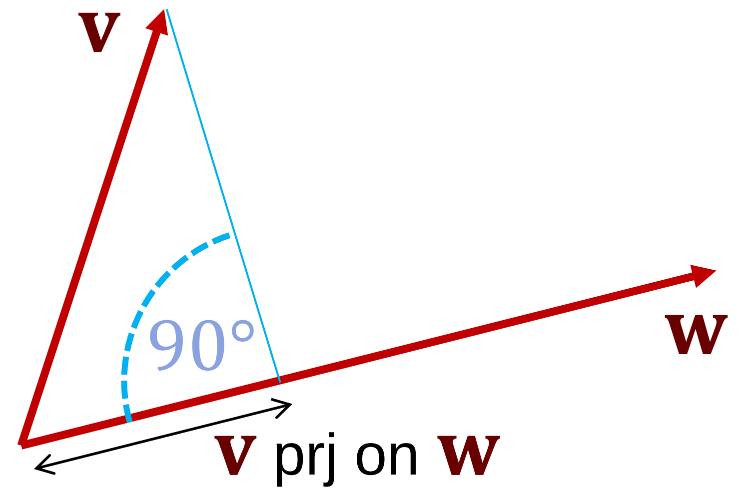


Angle between Vectors

angle $\angle(\mathbf{v}_1, \mathbf{v}_2)$
yields real number
$[0, \ldots, 2\pi) = [0, \ldots, 360°)$

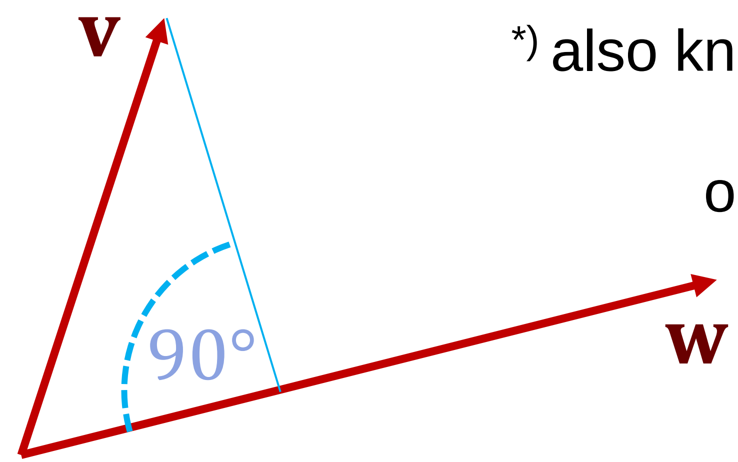


Angle between Vectors



Projection

Projection: determine
length of $\mathbf{v}$ along direction of $\mathbf{w}$

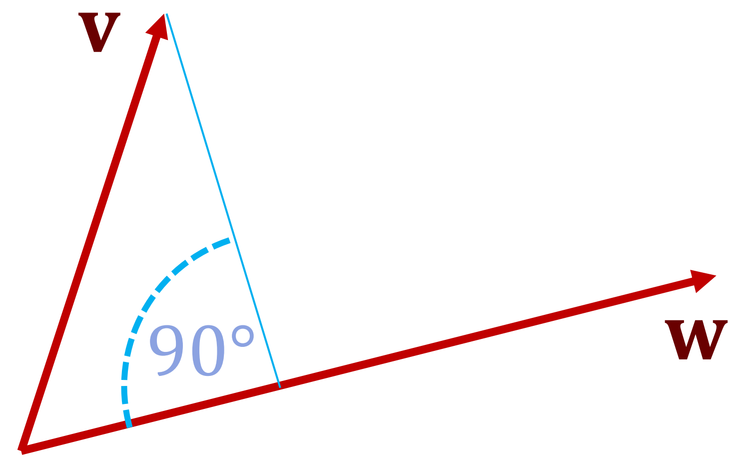


*) also known as *inner product* or *dot-product*

Scalar Product*)

$$\mathbf{v} \cdot \mathbf{w} = \|\mathbf{v}\| \cdot \|\mathbf{w}\| \cdot \cos \angle(\mathbf{v}, \mathbf{w})$$

$$\mathbf{v} \cdot \mathbf{w} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \cdot \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} := v_1 \cdot w_1 + v_2 \cdot w_2$$



Scalar Product*)

$$\mathbf{v} \cdot \mathbf{w} = \|\mathbf{v}\| \cdot \|\mathbf{w}\| \cdot \cos \angle(\mathbf{v}, \mathbf{w})$$

also: $\langle \mathbf{v}, \mathbf{w} \rangle$

Length: $\|\mathbf{v}\| = \sqrt{\mathbf{v} \cdot \mathbf{v}}$

Angle: $\angle(\mathbf{v}, \mathbf{w}) = \arccos(\mathbf{v} \cdot \mathbf{w})$

Projection: „$\mathbf{v}$ prj on $\mathbf{w}$" $= \frac{\mathbf{v} \cdot \mathbf{w}}{\|\mathbf{w}\|}$

$$\mathbf{v} \cdot \mathbf{w} = \|\mathbf{v}\| \cdot \|\mathbf{w}\| \cdot \cos \angle(\mathbf{v}, \mathbf{w})$$

Comprises: length, projection, angles

- Properties
  - Symmetry (commutativity)

$$\langle \mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{v}, \mathbf{u} \rangle$$

  - Bilinearity

$$\langle \lambda \mathbf{v}, \mathbf{w} \rangle = \lambda \langle \mathbf{v}, \mathbf{w} \rangle = \langle \mathbf{v}, \lambda \mathbf{w} \rangle$$

$$\langle \mathbf{u} + \mathbf{v}, \mathbf{w} \rangle = \langle \mathbf{u}, \mathbf{w} \rangle + \langle \mathbf{v}, \mathbf{w} \rangle$$

(symmetry: same for second argument)

  - Positive definite

$$\langle \mathbf{u}, \mathbf{u} \rangle \geq 0, \qquad [\langle \mathbf{u}, \mathbf{u} \rangle = \mathbf{0}] \Rightarrow [\mathbf{u} = \mathbf{0}]$$

Settings

$$\lambda \in \mathbb{R}$$

$$\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^d$$

- Do not mix
  - Scalar-vector product
  - Inner (scalar) product

- In general

$$\langle \mathbf{x}, \mathbf{y} \rangle \cdot \mathbf{z} \neq \mathbf{x} \cdot \langle \mathbf{y}, \mathbf{z} \rangle$$

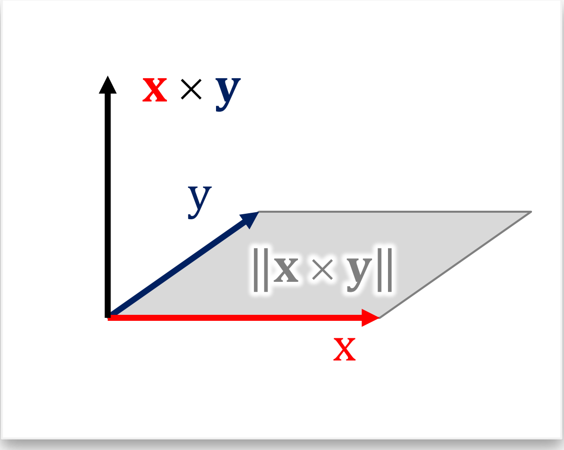
- Beware of notation:

- $(\mathbf{x} \cdot \mathbf{y}) \cdot \mathbf{z} \neq \mathbf{x} \cdot (\mathbf{y} \cdot \mathbf{z})$

- (no violation of associativity: different operations)

- Cross-Product: Exists Only For 3D Vectors!
  - $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$
  - $\mathbf{x} \times \mathbf{y} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \times \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} := \begin{pmatrix} x_2 y_3 - x_3 y_2 \\ x_3 y_1 - x_1 y_3 \\ x_1 y_2 - x_2 y_1 \end{pmatrix}$

- Geometrically: Theorem
  - $\mathbf{x} \times \mathbf{y}$ orthogonal to $\mathbf{x}, \mathbf{y}$
  - Right-handed system $(\mathbf{x}, \mathbf{y}, \mathbf{x} \times \mathbf{y})$
  - $\|\mathbf{x} \times \mathbf{y}\| = \|\mathbf{x}\| \cdot \|\mathbf{y}\| \cdot \sin\angle(\mathbf{x}, \mathbf{y})$



- Bilinearity
  - Distributive: $\mathbf{u} \times (\mathbf{v} + \mathbf{w}) = \mathbf{u} \times \mathbf{v} + \mathbf{u} \times \mathbf{w}$
  - Scalar-Mult.: $(\lambda \mathbf{u}) \times \mathbf{v} = \mathbf{u} \times (\lambda \mathbf{v}) = \lambda (\mathbf{u} \times \mathbf{v})$

- **But beware of**
  - **Anti**-Commutative: $\mathbf{u} \times \mathbf{v} = -\mathbf{v} \times \mathbf{u}$
  - **Not** associative;
    we can have $(\mathbf{u} \times \mathbf{v}) \times \mathbf{w} \neq \mathbf{u} \times (\mathbf{v} \times \mathbf{w})$