launch binder

(https://mybinder.org/v2/gh/KingaS03/Introduction-to-Python-2020-June/master)

Open in Colab

(https://colab.research.google.com/github/KingaS03/Introduction-to-Python-2020-June)

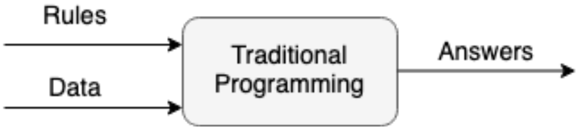
1. Introduction to the Mathematics Module for ML and DS

We are going to consider a common machine learning context, as this is illustrating all the major components of our course.

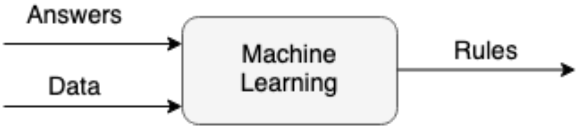
1.1. Machine learning versus classical programming

First let's take a look at machine learning and compare it with classical programming.

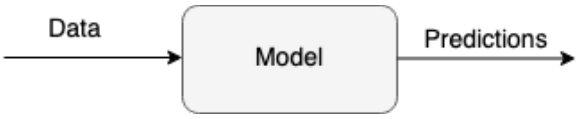
Machine Learning is the "field of study that gives computers the ability to learn without being explicitly programmed." - Arthur Samuel, 1959



versus



In case of some machine learning problems the resulting rules can be conceived as a model, which for any input data is able to predict the associated output.



Mathematically we can think of this model as a function that assigns to an input value a predicted output value and at the same time it depends also on some model parameters (weights and intercept). The model parameters are determined in such a way to minimise the loss function. This phenomenon is concisely described in the following quote:

"A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E." - Tom Mitchell, 1997

1.2. Simple machine learning setting - Linear regression

One of the simplest settings of a machine learning algorithm is the linear regression. To get a quick intuition about how it works play with the below interactive graph. You can change the position of the blue datapoints by dragging them with the mouse. You can change the position of the red line by moving the two red points of it.

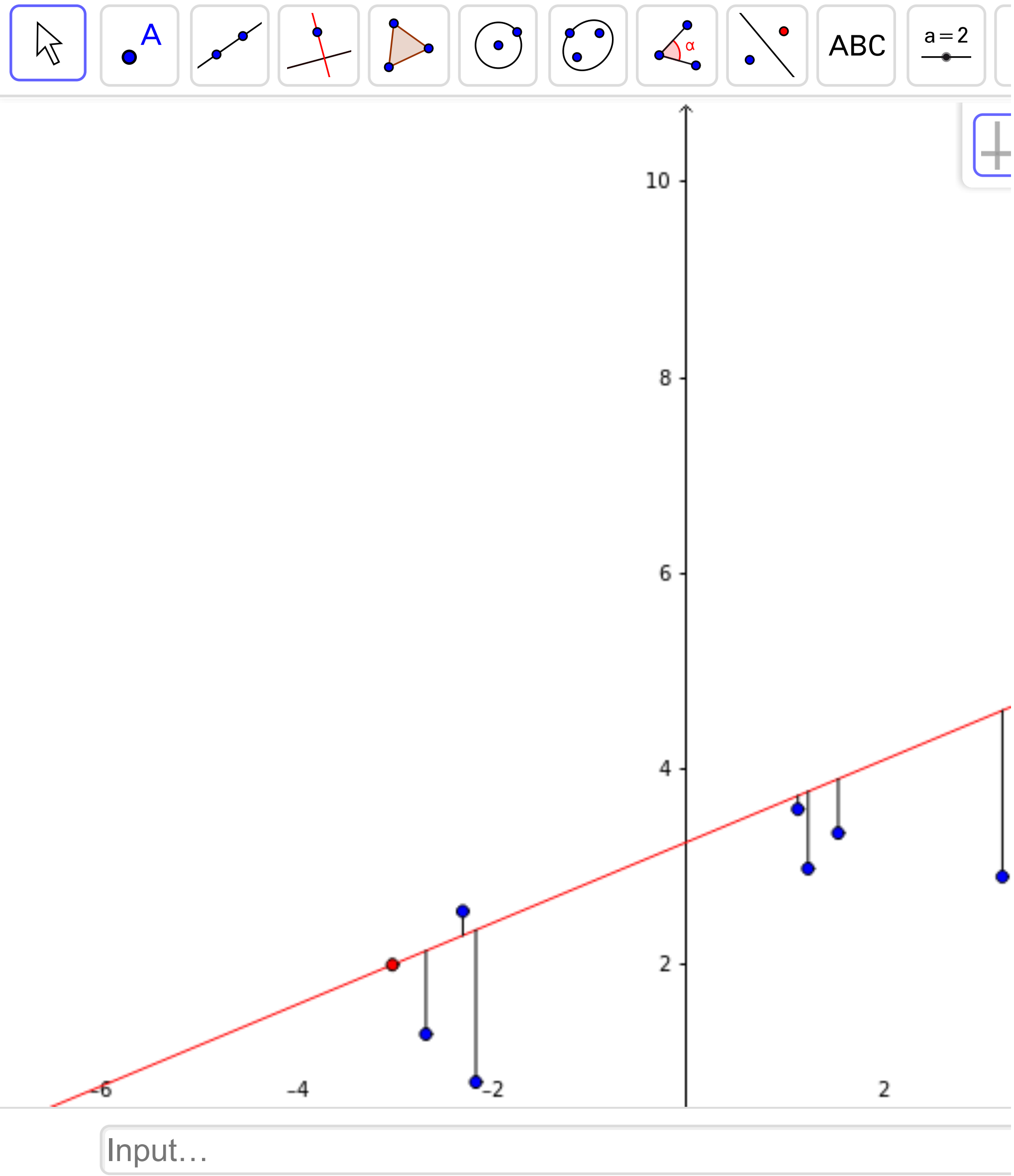
What happens on the plot on the right hand side if you change the position of one of the red points? How can you explain the observed behaviou?

Take a look at the blue end red values in the upper right corner. How do these values change?

What is the starting point of a regression analysis and what is its objective?

In [22]:
```
from IPython.display import IFrame

#IFrame("https://www.geogebra.org/m/xC6zq7Zv",800,800)
IFrame("https://www.geogebra.org/classic/gvtvpem2", 1400, 6
00)
```

Out[22]:



Let's take a look at a concrete numerical example:

We would like to predict the price of apartments as a linear function of their surface.

We consider the following data points:

Surface area in m^2	Price in tausends of CHF
40	275
70	500
80	470
100	650
115	690
120	750

The surface area, denoted by x, is the single **explanatory/dependent variable**.

The price, denoted by y, is the single **independent variable**.

The apartments, whose prices are enlisted in the above table are called **observations** and we will refer to their **features** (surface and price) as x_i, respectively y_i, where i is the index of the apartment ($i = \overline{0, 5}$).

We would like to approximate our data points by a line defined by the equation

$$y = w \cdot x + b,$$

where w is called **weight/gradient** and b is called **intercept**. These parameters w and b are determined in such a way that the mean squared error of the approximations is minimal.

For our set of apartments the $\boldsymbol{MSE}$ **(mean squared error)** can be calculated as follows:

$$\begin{aligned} MSE &= \frac{1}{6}\sum_{i=0}^{5}\Big(y_i - (w \cdot x_i + b)\Big)^2 \\ &= \frac{1}{6}\Big((275 - (w \cdot 40 + b))^2 + (500 - (w \cdot 70 + b))^2 + (470 - (w \cdot 80 + b))^2 + (650 - (w \cdot 100 + b))^2 + (\end{aligned}$$

MSE is a function of the parameters b and w.

The goal is to determine the parameters b and w in such a way to obtain the minimal MSE.

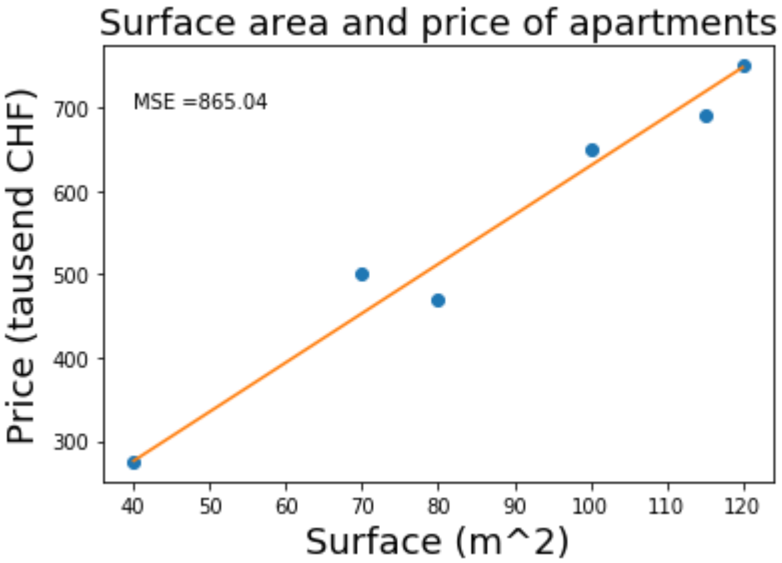
Experiment with the following code. Fit the red line to the data points by trying out different values for the parameters w and b.

In [60]:

```
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

# 300 random samples
x = np.array([40, 70, 80, 100, 115, 120])
y = np.array([275, 500, 470, 650, 690, 750])

plt.plot(x, y, 'o') #scatter plot of data points

w = 5.9 # change this value
b = 40 # change this value

plt.plot(x, b + w*x) #add line of best fit

MSE = np.mean((y-(b + w*x))**2)

# legend, title, and labels.
plt.text(40,700, f"MSE ={MSE:.2f}")
plt.title('Surface area and price of apartments', size=18)
plt.xlabel('Surface (m^2)', size=18)
plt.ylabel('Price (tausend CHF)', size=18)
```

Out[60]:

```
Text(0,0.5,'Price (tausend CHF)')
```



Compare your values with the optimal ones, by running the code `w, b = np.polyfit(x, y, 1)` .

Due to the simplicity of the linear model, it is possible to derive the explicit formulas for the parameters by calculating the partial derivatives of the MSE w.r.t. the parameters and setting the values of these to 0. Without detailed explanation

$$\frac{\partial MSE}{\partial w} = 0, \quad \frac{\partial MSE}{\partial b} = 0$$

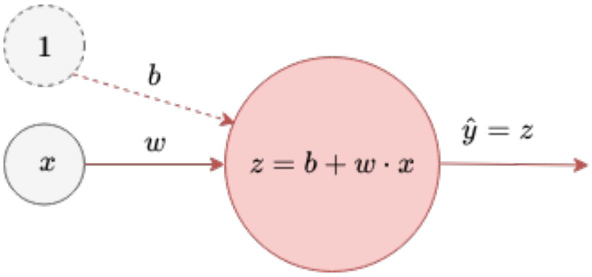
one can conclude:

$$\begin{cases} w = \dfrac{\sum_i (x_i - \bar{x}) \cdot (y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2} = 5.70 \\ b = \bar{y} - w \cdot \bar{x} = 57.30 \end{cases}$$

where $\bar{x} = \frac{\sum_i x_i}{n}$ and $\bar{y} = \frac{\sum_i y_i}{n}$

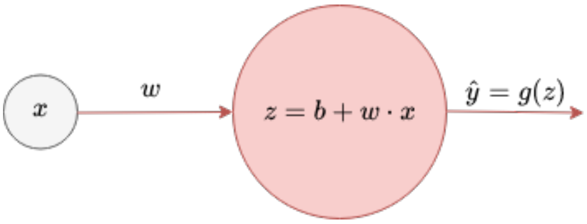
1.3. Neuronal networks

The above univariate linear regression model can be presented as



For the future notation we leave away the virtual input of 1.

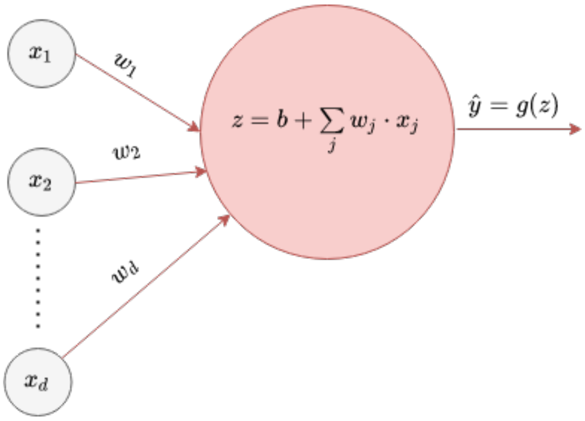
Now we make the model more complex until we get to the a two-layer neuronal network. First we apply an activation function g to the linear transformation $b + w \cdot x$.



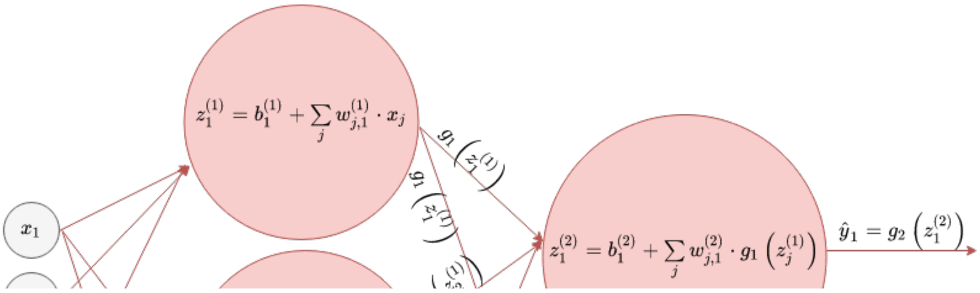
The above red ball corresponds to the smallest building unit of a neuronal network, namely a neuron. In a neuron:

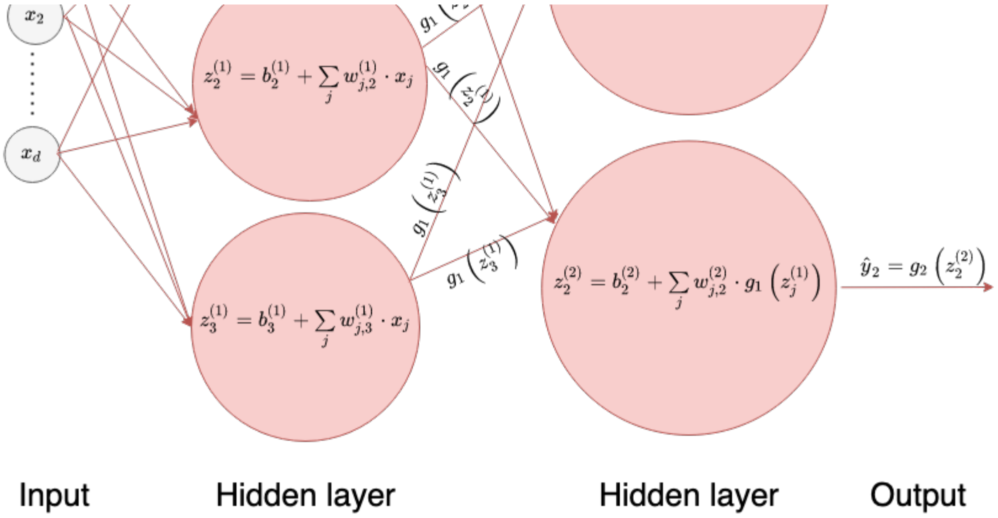
- there happens a linear transformation
- to which an activation function is applied and this provides the output of the neuron.

Next we allow for more inputs.



Finally we allow also for more outputs and we have two hidden layers.


$z_1^{(1)} = b_1^{(1)} + \sum_j w_{j,1}^{(1)} \cdot x_j$
x_1
$g_1\left(z_1^{(1)}\right)$
$g_1\left(z_1^{(1)}\right)$
$z_1^{(2)} = b_1^{(2)} + \sum_j w_{j,1}^{(2)} \cdot g_1\left(z_j^{(1)}\right)$
$\hat{y}_1 = g_2\left(z_1^{(2)}\right)$


x_2
x_d
$z_2^{(1)} = b_2^{(1)} + \sum_j w_{j,2}^{(1)} \cdot x_j$
$z_3^{(1)} = b_3^{(1)} + \sum_j w_{j,3}^{(1)} \cdot x_j$
$g_1\left(z_2^{(1)}\right)$
$g_1\left(z_3^{(1)}\right)$
$g_1\left(z_3^{(1)}\right)$
$z_2^{(2)} = b_2^{(2)} + \sum_j w_{j,2}^{(2)} \cdot g_1\left(z_j^{(1)}\right)$
$\hat{y}_2 = g_2\left(z_2^{(2)}\right)$
Input
Hidden layer
Hidden layer
Output

The number of hidden layers indicates that the last neuronal network is a two-layer neuronal network.

1.4. Motivation

Exactly as in the case of linear regression the weight matrices $W^{(1)}$, $W^{(2)}$, respectively the intercepts $b^{(1)}$ and b^2 will be parameters of the loss function which is subject to minimisation. In the general case there is no immediate straightforward formula for the optimal parameters.
The minimum of the loss function can be approximated by the **gradient descent** method.

$\Uparrow$

For the gradient descent method we should be able to derive the **partial derivatives** of the outputs w.r.t. all parameters of the model.

$\Uparrow$

For neuronal networks with more hidden layers and differentiable activation functions these partial derivatives can be deremined by the **chain rule**.

$\Uparrow$

To apply the chain rule for a setting like in the last network, one needs to perform **matrix multiplications**.

The number of machine learning algorithms is large. That's why generally a huge amount of input data is needed to determine the model parameters. Alternatively, if we don't possess that much data, we can reduce the dimensionality of the input data (and by that we end up also with a smaller number of model parameters). For dimensionality reduction we can use the **PCA (principal component analysis)**, which is the same as singular value decomposition. The first name is used more in the circle of statisticians and the second name is more popular among theoretical mathematicians. To derive PCA, we need the notion of **orthogonal projection**, **eigenvalues and eigenvectors**, **the method of Lagrange multipliers** and some **descriptive statistics**.

Furthermore, when the output of a neuronal network is a distribution, **probability theory** will be needed also to measure the distance between the observed distribution and the predicted one.

1.5. Schedule

-Linear algebra

- vector operations
  - vector addition,
  - vector substraction,
  - multiplication of a vector by a scalar
  - the dot product
- matrix operations
  - matrix addition
  - matrix substraction
  - multiplication of a matrix by a scalar
  - matrix multiplication
  - inverse of a square matrix
- projection and the dot product
- orthogonal matrices
- change of basis
- eigenvalues and eigenvectors of matrices

-Calculus

-PCA

-Probability theory and statistics

2. Linear algebra

2.1. Motivation

We are able to solve equations of the form: $ax + b = c$, where a, b, c are real coefficients and x is the unknown variable.

For example we can follow the next steps to solve the $5x + 3 = 13$ equation

$$\begin{aligned} 5x + 3 &= 13 \quad | -3 \\ 5x &= 10 \quad | : 5 \\ x &= 2 \end{aligned}$$

or equivalently

$$\begin{aligned} 5x + 3 &= 13 \quad | + (-3) \\ 5x &= 10 \quad | \cdot 5^{-1} = \frac{1}{5} \\ x &= 2 \end{aligned}$$

Let us consider the following set-up. You have beakfast together with some of your colleagues and you are paying by turn. You don't know the price of each ordered item, but you remember what was ordered on the previous three days and how much did your colleagues pay for it each time:

- 3 days ago your group has ordered 5 croissants, 4 coffees and 3 juices and they have payed 32.3 CHF.
- 2 days ago your group has ordered 4 croissant, 5 coffees and 3 juices and they payed 32.5 CHF.
- 1 day ago the group has ordered 6 croissants, 5 coffees and 2 juices and that costed 31 CHF.

Today the group has ordered 7 croissants, 4 coffees and 2 juices and you would like to know whether the amount of 35 CHF available on your uni card will cover the consumption or you need to recharge it before paying.

By introducing the notations

- x_1 for the price of a croissant,
- x_2 for the price of a coffee,
- x_3 for the price of a juice, then our information about the consumption of the previous 3 days can be summarised in the form of the following 3 linear equations

$$\begin{cases} 5 \cdot x_1 + 4 \cdot x_2 + 3 \cdot x_3 = 32.3 \\ 4 \cdot x_1 + 5 \cdot x_2 + 3 \cdot x_3 = 32.5 \\ 6 \cdot x_1 + 5 \cdot x_2 + 2 \cdot x_3 = 31 \end{cases}$$

The quantity ordered on the current day is $7 \cdot x_1 + 4 \cdot x_2 + 2 \cdot x_3$. To determine this one possibility is to calculate the price of each product separately, i.e. we solve the linear equation system first and then susbstitute the prices in the previous formula.

The above system in matrix form

$$\begin{pmatrix} 5 & 4 & 3 \\ 4 & 5 & 3 \\ 6 & 5 & 2 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 32.3 \\ 32.5 \\ 31 \end{pmatrix}$$

If we introduce for the matrix, respectively the two vectors in the above formula the notations A, x, b , then we get

$$A \cdot x = b$$

One can observe that formally this looks the same as the middle state of our introductory linear equation with real coefficients $5x = 10$. So our goal is to perform a similar operation as there, namely we are looking for teh operation that would make A dissappear from the left hand side of the equation. We will see later that this operation will be the inverse operation of multiplication by a matrix, namely multiplication by the inverse of a matrix.

In our applications we will encounter for example when deriving the weights of the multivariate linear regression, a matrix equation of the form: $A \cdot x + b = 0$. This example motivates the introduction of vector substraction, as well

2.2. Vectors

Definition of vectors

Vectors are elements of a linear vector space. The vector space we are going to work with is R^n, where n is the dimension of the space and it can be $1, 2, 3, 4, \ldots$. An element of such a vector space can be described by an ordered list of n components of the vector.

$x = (x_1, x_2, \ldots, x_n)$, where $x_1, x_2, \ldots, x_n \in \mathrm{R}$ is an element of R^n.

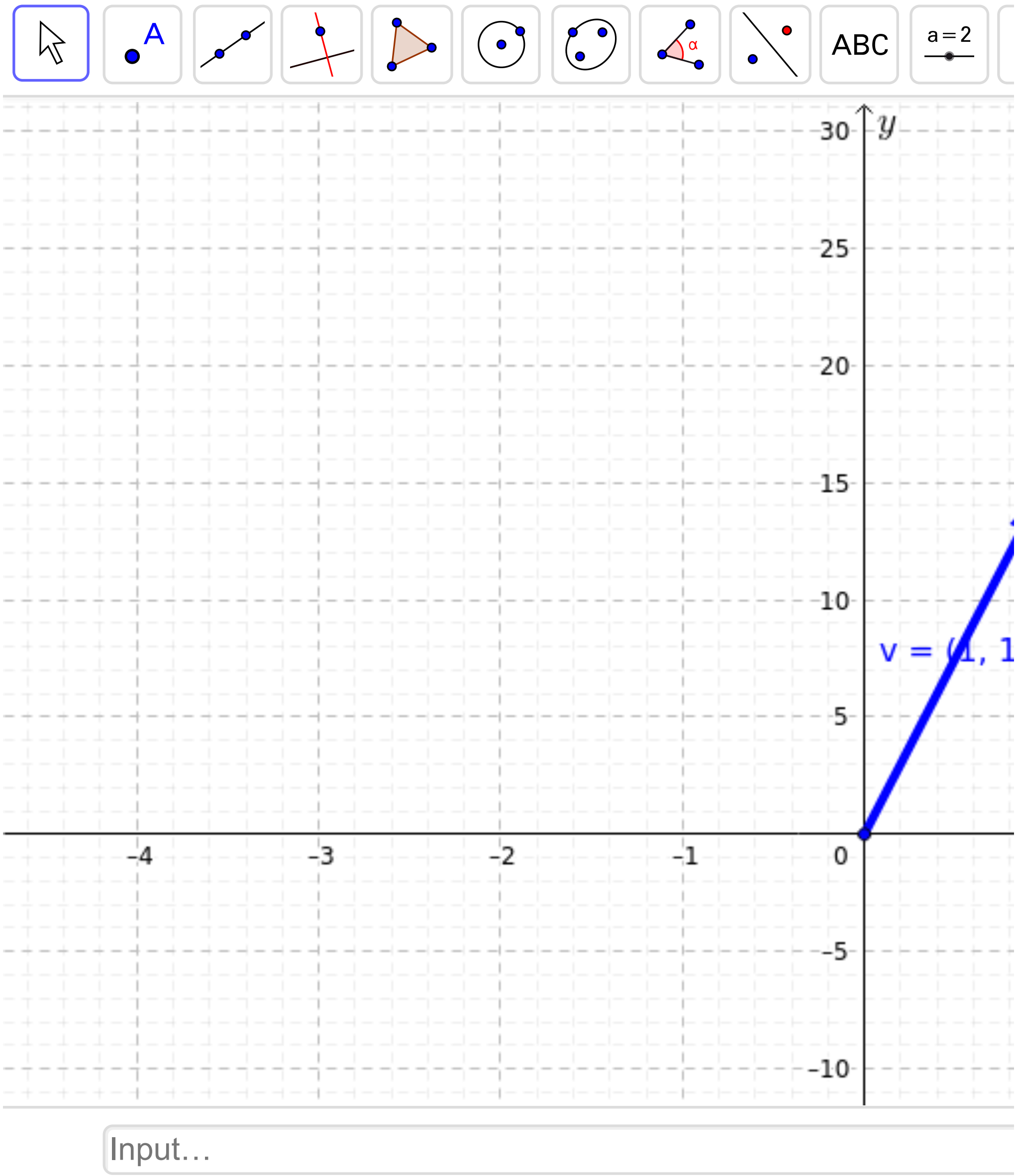
Example $x = (1, 2)$ is a vector of the two dimensional vector space R^2.

2.2.1. Geometrical representation of vectors

Below a 2-dimensional vector is represented. You can move its endpoints on the grid and you will see how do its components change.

```
In [14]: IFrame("https://www.geogebra.org/classic/cnvxpycc", 800, 60
         0)
```

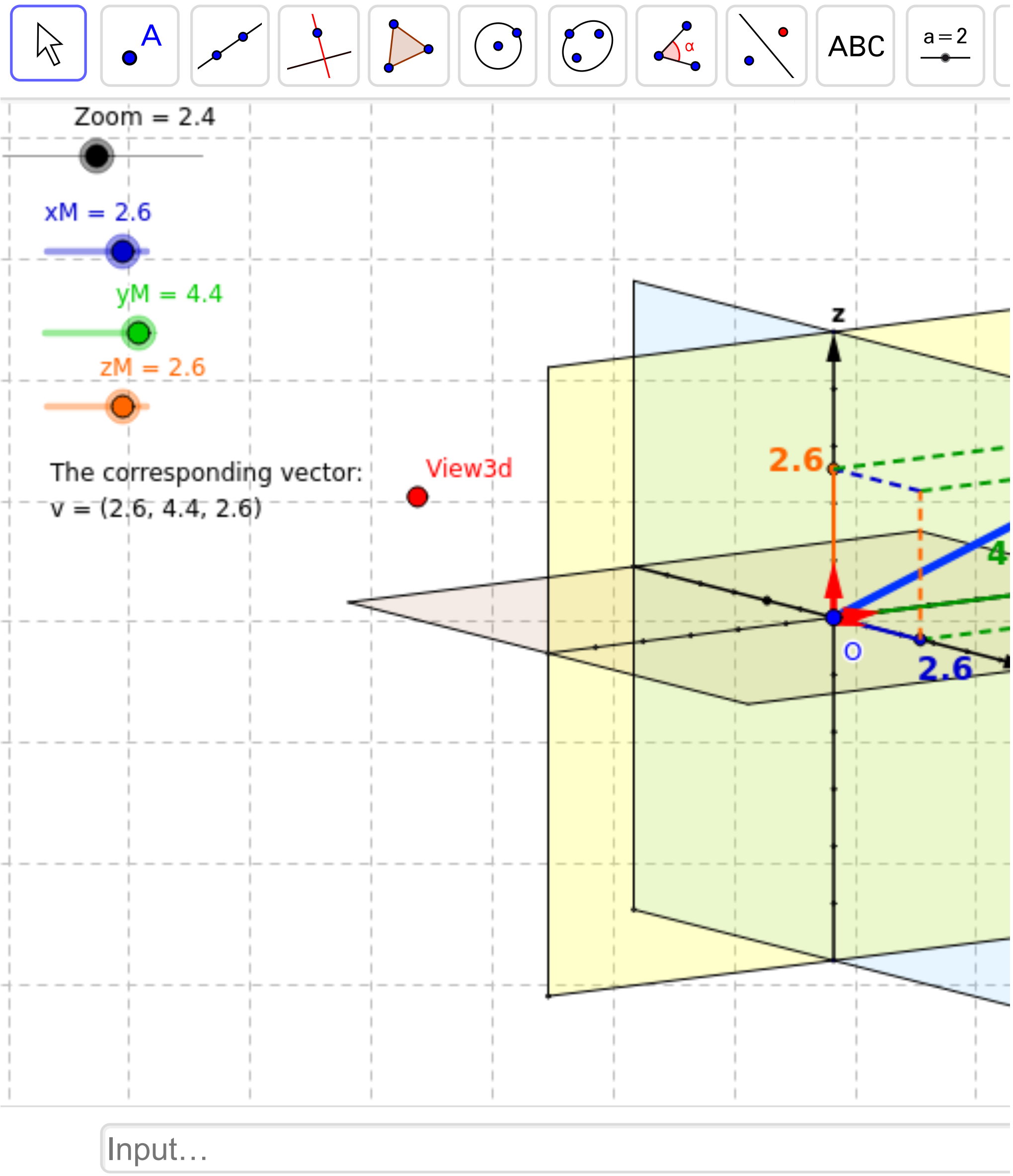
Out[14]:



Experiment with the 3-dimensional vector in the interactive window below.

```
In [18]: IFrame("https://www.geogebra.org/classic/meg4scuj", 800, 600)
```

Out[18]:



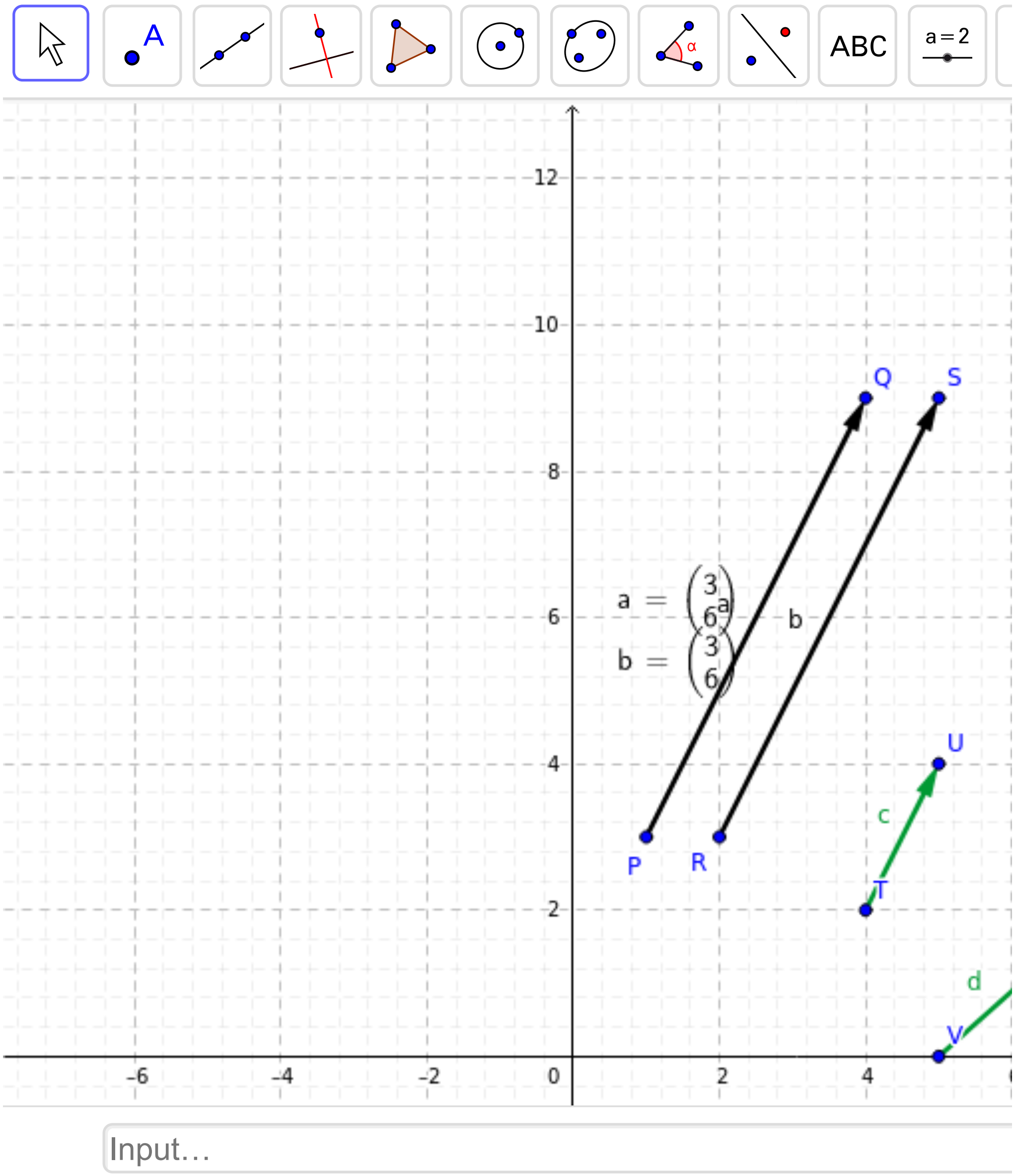
The corresponding vector:
v = (2.6, 4.4, 2.6)

The following interactive window explains when are two vectors equal.

In [23]:

```
IFrame("https://www.geogebra.org/classic/fkkbkvuj", 800, 60
0)
```

Out[23]:


ABC
a = 2
12
10
8
6
4
2
0
-6
-4
-2
2
4
Q
S
a = $\begin{pmatrix}3\\6\end{pmatrix}$
b = $\begin{pmatrix}3\\6\end{pmatrix}$
a
b
U
c
P
R
T
d
V
Input…

2.2.2. Vector addition

Definition of vector addition

Vector addition happens component-wise, namely the sum of the vectors $x = (x_1, x_2, \cdots, x_n)$ and $y = (y_1, y_2, \cdots, y_n)$ is:

$$x + y = (x_1 + y_1, x_2 + y_2, \ldots, x_n + y_n)$$

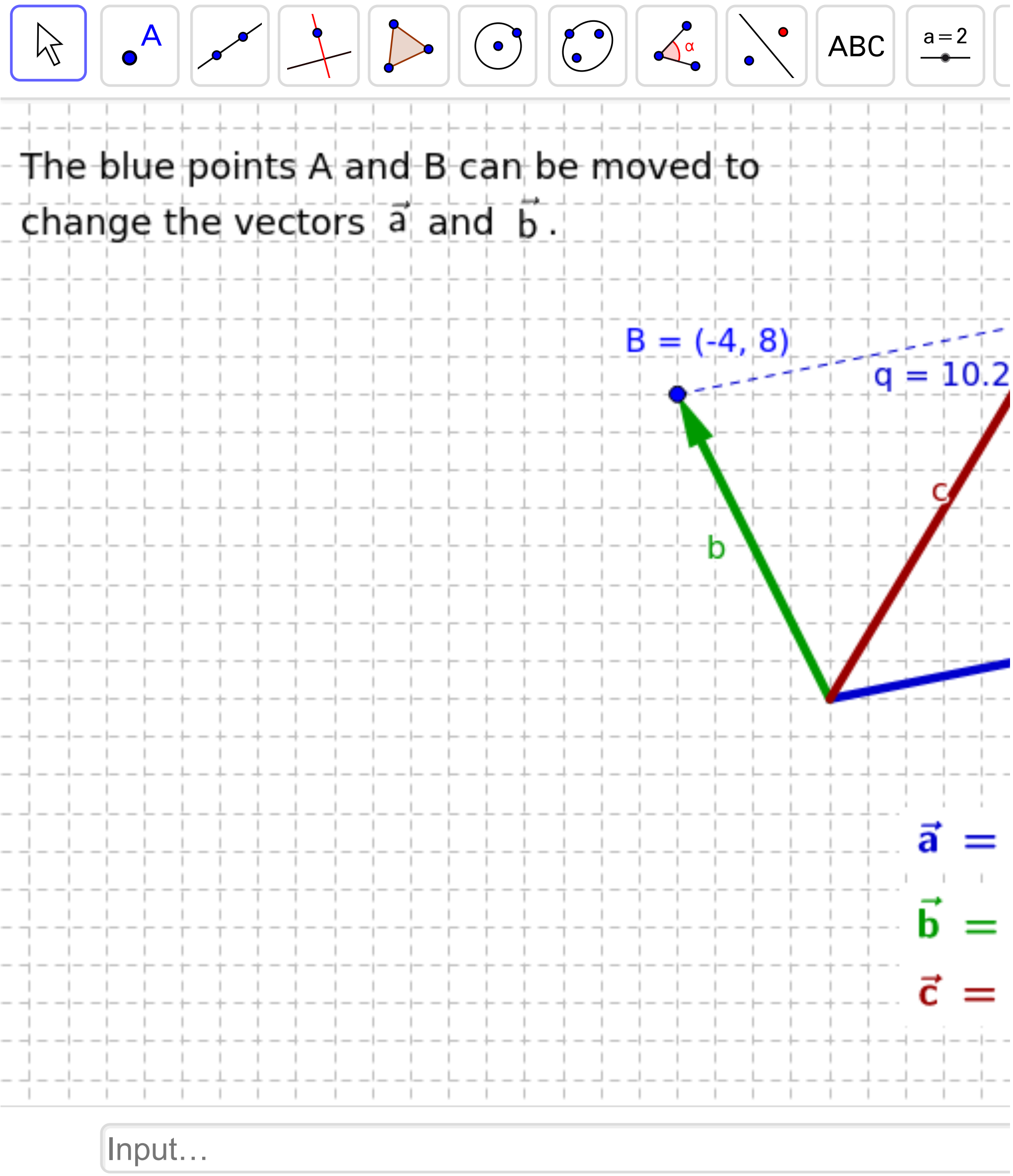
There exist two approaches to visualise vector addition

1. parallelogram method
2. triangle method

Both are visualised below.

In [25]:
```
IFrame("https://www.geogebra.org/classic/jnchvrhg", 800, 60
0)
```

Out[25]:



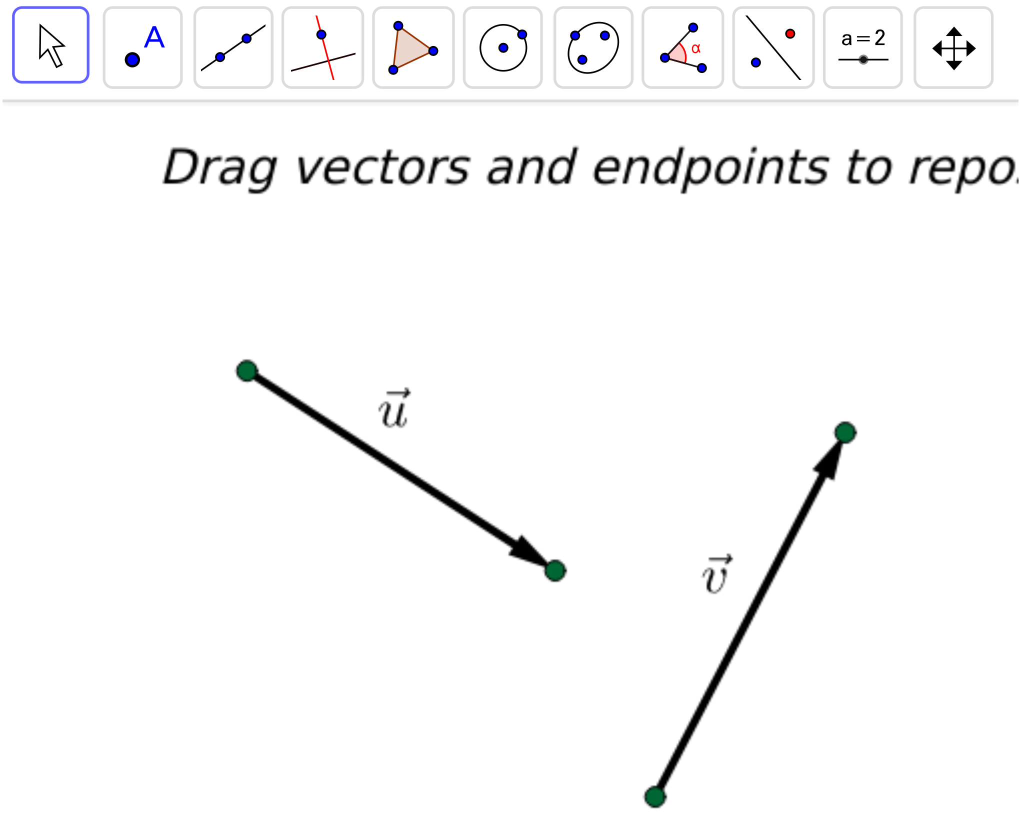
The blue points A and B can be moved to change the vectors $\vec{a}$ and $\vec{b}$.

Below you can see another approach to vector addition.

In [62]:

```
IFrame("https://www.geogebra.org/classic/mzgchv22", 1200, 6
00)
```

Out[62]:



2.2.3. Multiplication of vectors by a scalar

Definition of the multiplication by a scalar

This happens also component-wise exactly as addition, namely

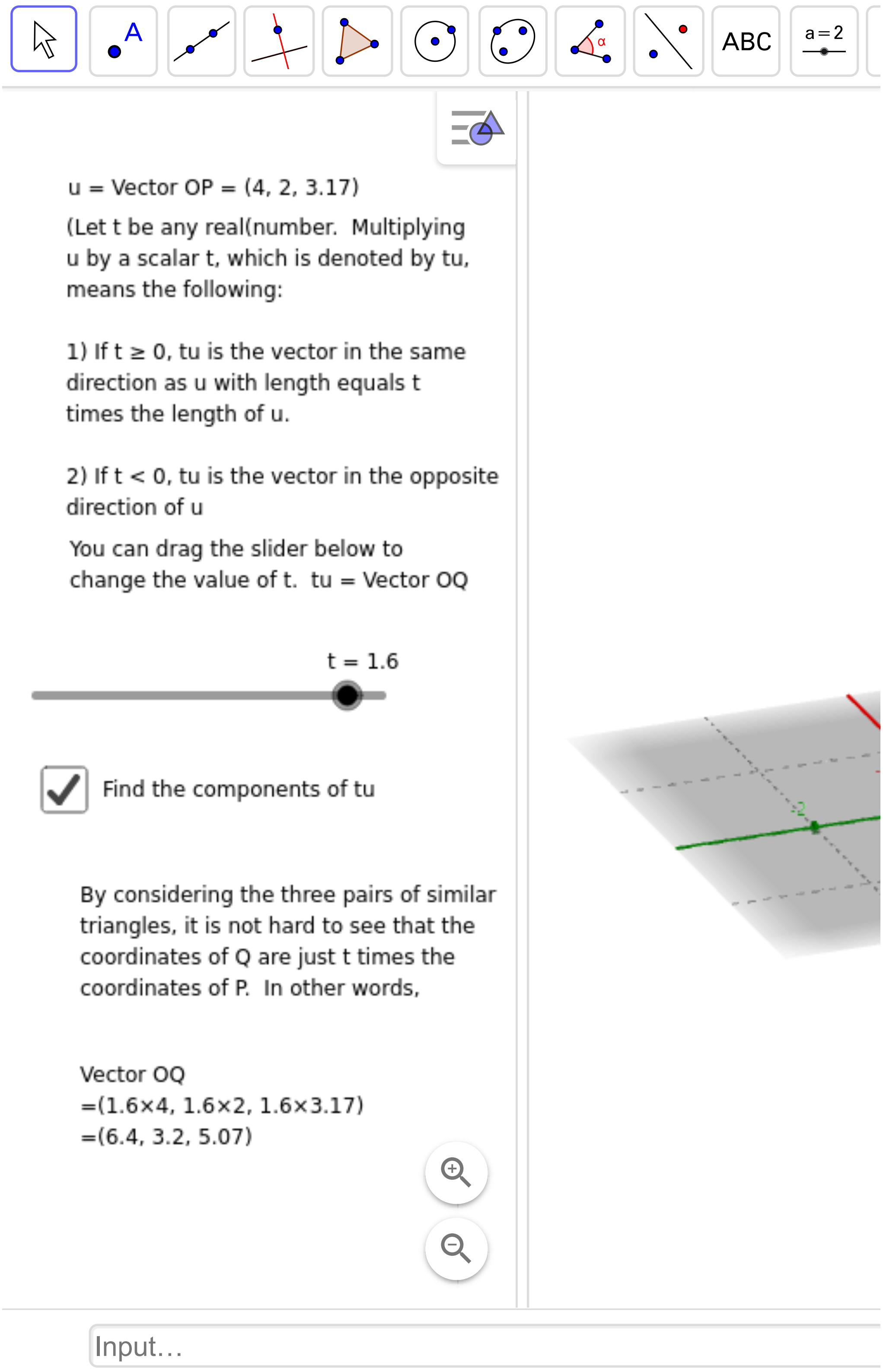
$$\lambda(x_1, x_2, \ldots, x_n) = (\lambda x_1, \lambda x_2, \ldots, \lambda x_n).$$

This operation is illustrated below

In [61]:

```
IFrame("https://www.geogebra.org/classic/gxhsev8k", 800, 80
0)
```

Out[61]:



u = Vector OP = (4, 2, 3.17)

(Let t be any real(number. Multiplying u by a scalar t, which is denoted by tu, means the following:

1) If t ≥ 0, tu is the vector in the same direction as u with length equals t times the length of u.

2) If t < 0, tu is the vector in the opposite direction of u

You can drag the slider below to change the value of t. tu = Vector OQ

t = 1.6

☑ Find the components of tu

By considering the three pairs of similar triangles, it is not hard to see that the coordinates of Q are just t times the coordinates of P. In other words,

Vector OQ
=(1.6×4, 1.6×2, 1.6×3.17)
=(6.4, 3.2, 5.07)

2.2.4. Vector substraction

Definition of vector substraction

Vector substraction happens component-wise, namely the difference of the vectors $x = (x_1, x_2, \cdots, x_n)$ and $y = (y_1, y_2, \cdots, y_n)$ is:

$$x - y = (x_1 - y_1, x_2 - y_2, \ldots, x_n - y_n)$$

Observe that vector substraction is not commutative, i.e. $x - y \neq y - x$ in general.

2.2.5. Abstract linear algebra terminology

1. For any two elements $x, y \in \mathrm{R}^n$ it holds that
$$x + y \in \mathrm{R}^n.$$
This property is called **closedness** of R^n w.r.t. addition.
2. Observe the **commutativity** of the addition on R^n is inherited by the vectors in R^n, i.e.
$$x + y = y + x$$
for any $x, y \in \mathrm{R}^n$.
3. Observe that addition is also **associative** on R^n, i.e.
$$x + (y + z) = (x + y) + z, \qquad \text{for any } x, y, z \in \mathrm{R}^n$$
4. If we add the zero vector $\mathbf{0} = (0, 0, \ldots, 0) \in \mathrm{R}^n$ to any other vector $x \in \mathrm{R}^n$ it holds that
$$\mathbf{0} + x = x + \mathbf{0} = x.$$
The single element with the above property is called the **neutral element** w.r.t. addition.
5. For a vector $x = (x_1, x_2, \ldots, x_n)$ the vector x^* for which
$$x + x^* = x^* + x = \mathbf{0}$$
is called the **inverse vector** of x w.r.t. addition.

What is the inverse of the vector $x = (2, 3, -1)$? Inverse: $-x = (-2, -3, 1)$

What is the inverse of a vector $x = (x_1, x_2, \ldots, x_n)$? Inverse: $-x = (-x_1, -x_2, \ldots, -x_n)$

As every vector of R^n possesses an inverse, we introduce the notation $-x$ for its inverse w.r.t addition.

A set V with an operation $\circ$ that satsifies the above properties is called a **commutative or Abelian group** in linear algebra. For us $V = \mathrm{R}^n$ and $\circ = +$.

The scalar mutiplication, that we have introduced, has the following properties

1. **associativity** of multiplication: $(\lambda_1\lambda_2)x = \lambda_1(\lambda_2 x)$,
2. **distributivity**: $(\lambda_1 + \lambda_2)x = \lambda_1 x + \lambda_2 x$ and $\lambda(x + y) = \lambda x + \lambda y$,
3. **unitarity**: $1x = x$, for all $x, y \in \mathrm{R}^n$ and $\lambda, \lambda_1, \lambda_2$ scalars.

Our scalars are elements of R. This set is a **field**, i.e. the operations $\lambda_1 + \lambda_2, \lambda_1 - \lambda_2, \lambda_1 \cdot \lambda_2$ make sense for any $\lambda_1, \lambda_2 \in \mathrm{R}$ and the $\lambda_1/\lambda_2 = \lambda_1 \cdot \lambda_2^{-1}$ can be performed also when $\lambda_2 \neq 0$.

A **vector space** consists of a set V and a field F and two operations:

- an operation called vector addition that takes two vectors $v, w \in V$, and produces a third vector, written $v + w \in V$,

- an operation called scalar multiplication that takes a scalar $\lambda \in F$ and a vector $v \in V$, and produces a new vector, written $cv \in V$, which satisfy all the properties enlisted above (5+3).

Remark
Observe that

$$x - y = x + (- y),$$

which means that the difference of x and y can be visualised as a vector addition of x and $-y$.

$-y$ is here the inverse of the vector y w.r.t. addition. Geometrically $-y$ can be represented by the same oriented segment as y, just with opposite orientation.

2.2.6. Modulus of a vector, length of a vector, size of a vector

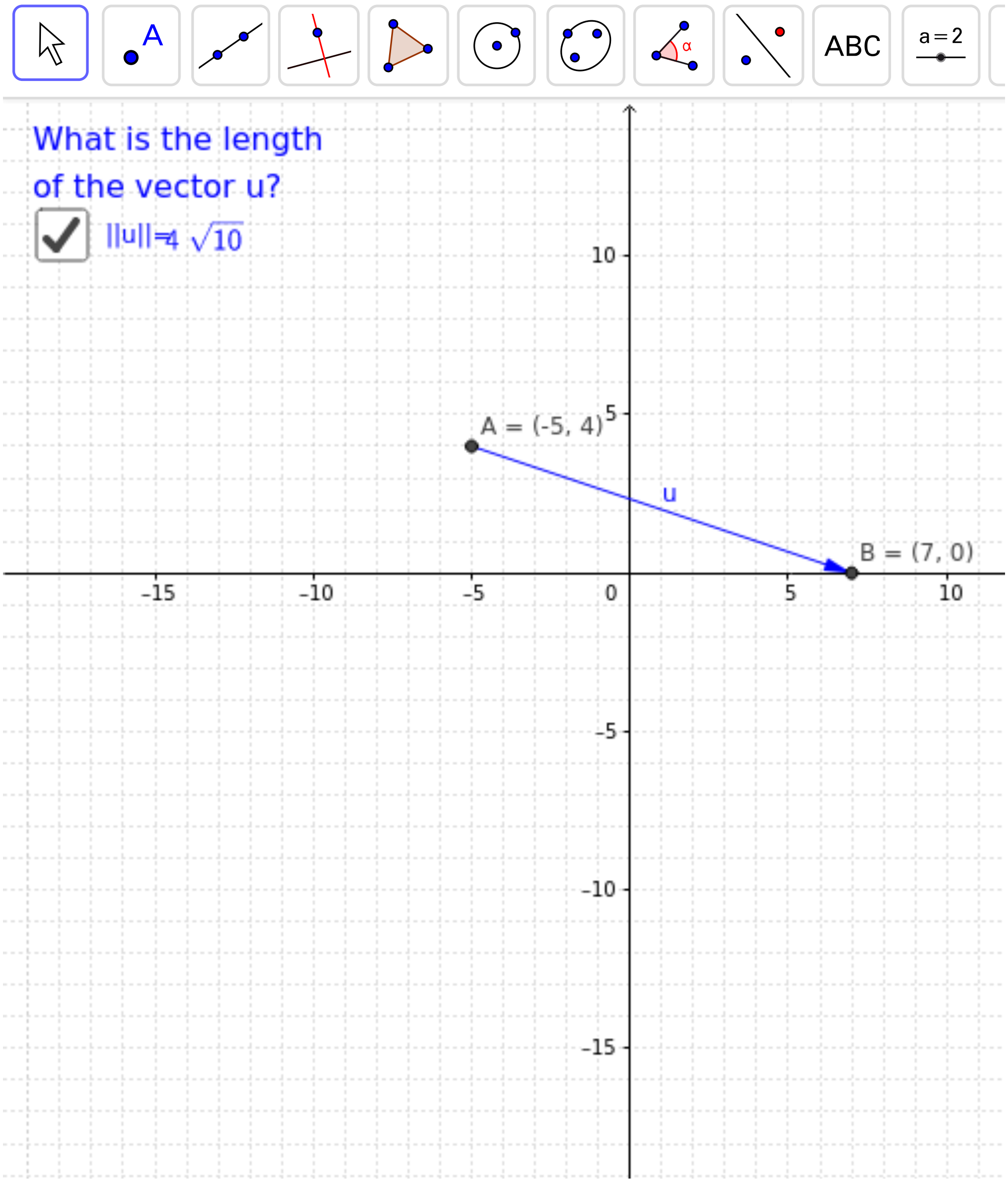
The length of a vector or norm of a vector $x = (x_1, x_2, \cdots, x_n)$ is given by the formula

$$||x|| = \sqrt{x_1^2 + x_2^2 + \cdots x_n^2}$$

Experiment with the interactive window below and derive the missing formula.

In [63]:
```
IFrame("https://www.geogebra.org/classic/mfzdes3n", 800, 600)
```

Out[63]:



What is the length of the vector u?

☑ ||u||=4 $\sqrt{10}$

Each vector $x = (x_1, x_2, \ldots, x_n) \in \mathrm{R}^n$ is uniquely determined by the following two features:

- its magnitude / length / size / norm: $r(x) = ||x|| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$,
- its direction: $e(x) = \frac{x}{||x||} = \frac{1}{\sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}}(x_1, x_2, \ldots, x_n)$.

If the maginute $r \in \mathrm{R}$ and the direction $e \in \mathrm{R}^n$ of a vector is given, then this vector can be written as re.

Observe that $\frac{x}{||x||}$ has length 1.

2.2.7. Dot product / inner product / scalar product

Definition of the **dot product**

The **dot product** / **inner product** / **scalar product** of two vectors $x = (x_1, x_2)$ and $y = (y_1, y_2)$ is denoted by $\langle x, y \rangle$ and it is equal to the scalar $x_1 \cdot y_1 + x_2 \cdot y_2$.

This can be generalised to the vectors $x = (x_1, x_2, \cdots, x_n) \in \mathrm{R}^n$ and $y = (y_1, y_2, \cdots, y_n) \in \mathrm{R}^n$ as $\langle x, y \rangle = x_1 \cdot y_1 + x_2 \cdot y_2 + \cdots + x_n \cdot y_n$.

Observe that as a consequence of the definition distributivity over addition holds, i.e. $\langle x, y + z \rangle = \langle x, y \rangle + \langle x, z \rangle$.

Furthermore $\lambda \langle x, y \rangle = \lambda \langle x, y \rangle = \langle x, \lambda y \rangle$.

The last two properties together are called also **bilinearity** of the scalar product.

Observe also that the scalar product is **commutative**, i.e. $x \cdot y = y \cdot x$.

Question: What's the relation between the length of a vector and the dot product?

Length of a vector: $||x|| = \sqrt{x_1^2 + x_2^2 + \cdots x_n^2}$

The dot product of two vectors: $\langle x, y \rangle = x_1 \cdot y_1 + x_2 \cdot y_2 + \cdots + x_n \cdot y_n$.

Substituting x in the last equation instead of y, we obtain

$$\langle \mathbf{x}, \mathbf{x} \rangle = x_1^2 + x_2^2 + \cdots x_n^2 = ||\mathbf{x}||^{\mathbf{2}}$$

Convention
When talking exclusively about vectors, for the simplicity of writing, we often think of them as row vectors. However, when matrices appear in the same context and there is a chance that we will multiply a matrix by a vector, it is important to specify also whether we talk about a row or column vector. In this extended context a vector is considered to be a column vector by default.

From now on we are going to follow also this convention and we are going to think of a vector always as a column vector.

Relationship of dot product and matrix multiplication

Even if we didn't define formally the matrix product yet, we mention its relationship with the dot product, because in mathematical formulas it proves to be handy to have an alternative way for writing the dot multiplication.

The following holds for any vectors $x, y \in \mathrm{R}^n$

$$\langle \mathbf{x}, \mathbf{y} \rangle = x_1 \cdot y_1 + x_2 \cdot y_2 + \cdots x_n \cdot y_n = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \mathbf{x}^{\mathbf{T}} \cdot \mathbf{y},$$

where $\mathbf{x}^{\mathbf{T}} \cdot \mathbf{y}$ denotes the matrix product of the row vector x_T and the column vector y.

Furthermore, observe that due to the commutativity of the dot product

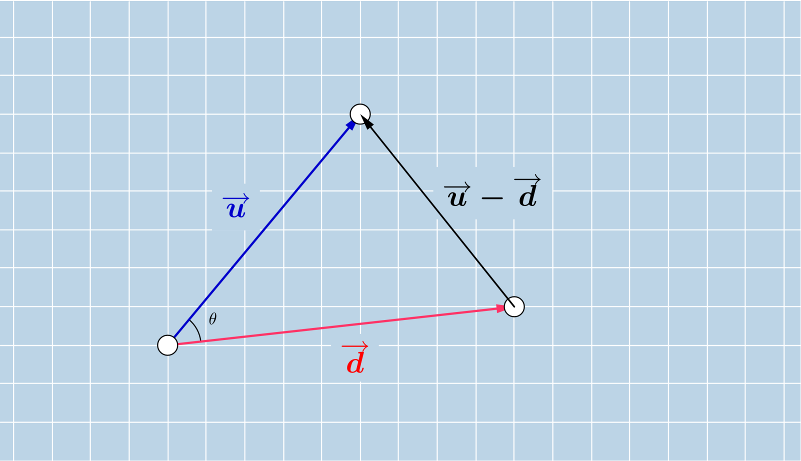
$$x^T \cdot y = \langle x, y \rangle = \langle y, x \rangle = y^T \cdot x$$

2.2.8. The dot product and the cosine rule

Let us consider two vectors u and d and denote their angle by θ. We construct the triangle having as sides the vectors u, d and $u - d$. In the forthcoming we derive the formula

$$\langle u, d \rangle = ||u|| \cdot ||d|| \cdot cos(\theta)$$

from the law of cosines.



The **law of cosines** is a generalisation of the **Pythagorean theorem** in a triangle, which holds not just for right triangles. As the Pythagorean theorem, this formulates also a relationship between the lengths of the three sides. In a triangle with side lengths a, b and c and an angle θ opposite to the side with length a, the law of cosines claimes that

$$a^2 = b^2 + c^2 - 2bc\cos(\theta).$$

In our setting we can write for the side lengths the norm / length of the vectors u, d, respectively $u - d$. In this way we obtain

$$||u - d||^2 = ||u||^2 + ||d||^2 - 2 \cdot ||u|| \cdot ||d|| \cdot \cos(\theta).$$

On the other hand using the relationship between the length of a vector and the dot product, we can write the following

$$||u - d||^2 = \langle u - d, u - d \rangle$$

Using the bilinearity and commutativity of the dot product we can continue by

$$||u - d||^2 = \langle u - d, u - d \rangle = \langle u, u \rangle - \langle d, u \rangle - \langle u, d \rangle - \langle d, d \rangle = ||u||^2 + ||d||^2 - 2\langle d, u \rangle$$

Summing up what did we obtain until now

$$\{ \;||u-d||^2 = ||u||^2 + ||d||^2 - 2 \cdot ||u|| \cdot ||d|| \cdot \cos(\theta) \quad |$$

2.2.9. Scalar and vector projection

Scalar projection: length of the resulting projection vector, namely

$$||\pi_d(u)|| = cos(\theta) \cdot ||u||$$

where θ is the angle of the vectors d and u.

Due the cosine rule that we have derived for the scalar product, we can substitute $\cos(\theta)$ by $\frac{\langle d,u \rangle}{||u|| \cdot ||d||}$ and we obtain the following formula for the length of the projection

$$||\pi_d(u)|| = \frac{\langle u, d \rangle}{||u|| \cdot ||d||} \cdot ||u|| = \frac{\langle u, d \rangle}{||d||}$$

Vector projection: We have determined the magnite of the projection vector, the direction is given by the one of the vector d. These two characteristics do uniquely define the projection vector, thus we can write

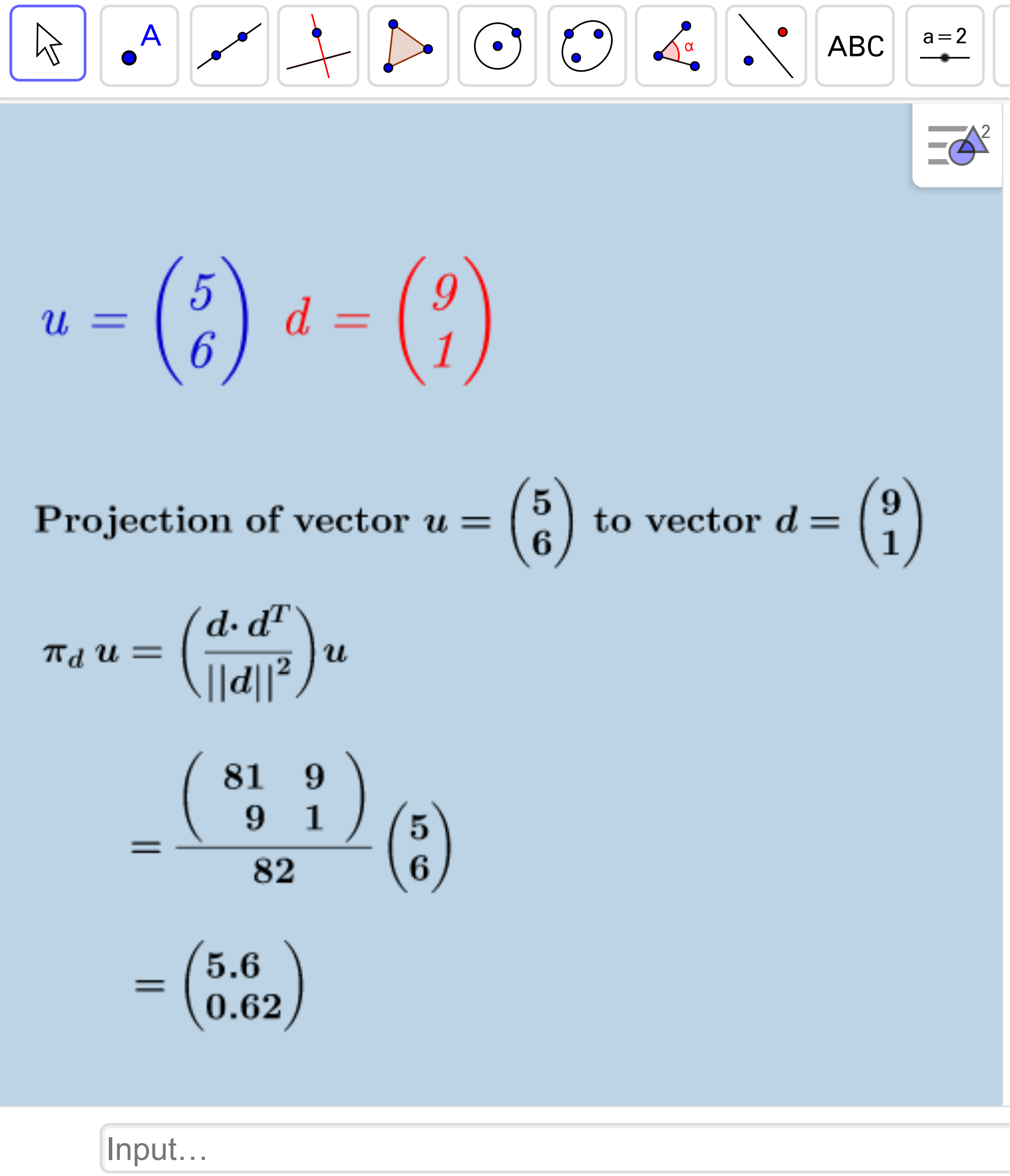
$$\pi_d(u) = ||\pi_d(u)|| \frac{d}{||d||} = \frac{\langle \mathbf{u}, \mathbf{d} \rangle}{||\mathbf{d}||} \frac{\mathbf{d}}{||\mathbf{d}||} = \frac{d\langle u, d \rangle}{||d||^2} = \frac{d\langle d, u \rangle}{||d||^2} = \frac{d \cdot (d^T \cdot u)}{||d||^2} = \frac{(d \cdot d^T) \cdot u}{||d||^2} = \frac{\mathbf{d} \cdot \mathbf{d^T}}{||\mathbf{d}||^\mathbf{2}} \cdot$$

The projection matrix $\frac{d \cdot d^T}{||d||^2}$ in R^n is an $n \times n$-dimensional matrix.

Exersize
Change the canonical basis to another orthogonal basis by scalar projection.

```
In [70]: IFrame("https://www.geogebra.org/classic/qhhqpmrt", 1000, 6
         00)
```

Out[70]:



$$u = \begin{pmatrix} 5 \\ 6 \end{pmatrix} \quad d = \begin{pmatrix} 9 \\ 1 \end{pmatrix}$$

Projection of vector $u = \begin{pmatrix} 5 \\ 6 \end{pmatrix}$ **to vector** $d = \begin{pmatrix} 9 \\ 1 \end{pmatrix}$

$$\pi_d \, u = \left(\frac{d \cdot d^T}{||d||^2} \right) u$$

$$= \frac{\begin{pmatrix} 81 & 9 \\ 9 & 1 \end{pmatrix}}{82} \begin{pmatrix} 5 \\ 6 \end{pmatrix}$$

$$= \begin{pmatrix} 5.6 \\ 0.62 \end{pmatrix}$$

2.2.10. Basis of a vectorspace, linear independence of vectors

Definition of linear combination

A linear combination of the vectors $x^{(1)}, x^{(2)}, \ldots, x^{(m)} \mathrm{R}^n$ is a vector of R^n, which can be written in the form of

$$\lambda_1 x^{(1)} + \lambda_2 x^{(2)} + \cdots \lambda_m \cdot x^{(m)},$$

where $\lambda_1, \lambda_2, \ldots, \lambda_m$ are real valued coefficients.

Example

Consider the vectors $x^{(1)} = \begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix}$ and $x^{(2)} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$.

Then

$$2x^{(1)} + 3x^{(2)} = 2\begin{pmatrix} 1 \\ 0 \\ 2 \end{pmatrix} + 3\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix}$$

is a linear combination of $x^{(1)}$ and $x^{(2)}$.

Definition of linear dependence

Let us consider a set of m vectors $x^{(1)}, x^{(2)}, \ldots, x^{(m)}$ in R^n. They are said to be linearly dependent if and only if there exist the not all zero factors $\lambda_1, \lambda_2, \ldots, \lambda_m \in \mathrm{R}$ such that

$$\lambda_1 x^{(1)} + \lambda_2 x^{(2)} + \cdots + \lambda_m x^{(m)} = \mathbf{0}$$

Remark

Observe that if $x^{(1)}, x^{(2)}, \ldots, x^{(m)}$ are dependent, then for some not all zero factors $\lambda_1, \lambda_2, \ldots, \lambda_m \in \mathrm{R}$ it holds that

$$\lambda_1 x^{(1)} + \lambda_2 x^{(2)} + \cdots + \lambda_m x^{(m)} = \mathbf{0}$$

We know that at least one of the factors is not zero, let us assume that λ_i is such a factor. This means that from the above equation we can express the vector x_i as a linear combination of the others.

Definition of linear independence

m vectors $x^{(1)}, x^{(2)}, \ldots, x^{(m)}$ in R^n are linearly independent if there exist no such factors $\lambda_1, \lambda_2, \ldots, \lambda_m \in \mathrm{R}$ for which

$$\lambda_1 x^{(1)} + \lambda_2 x^{(2)} + \cdots + \lambda_m x^{(m)} = \mathbf{0}$$

and where at least one factor is different of zero.

Alternative definition of linear independence

Equivalently $x^{(1)}, x^{(2)}, \ldots, x^{(m)}$ in R^n are linearly independent if and only if the equation

$$\lambda_1 x^{(1)} + \lambda_2 x^{(2)} + \cdots + \lambda_m x^{(m)} = \mathbf{0}$$

holds just for $\lambda_1 = \lambda_2 = \cdots = \lambda_m = 0$.

Remark

If we write the vectors from the above equation by their components, the above equation can be equivalently transformed to

$$\lambda_1 \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \\ \vdots \\ x_n^{(1)} \end{pmatrix} + \lambda_2 \begin{pmatrix} x_1^{(2)} \\ x_2^{(2)} \\ \vdots \\ x_n^{(2)} \end{pmatrix} + \cdots + \lambda_m \begin{pmatrix} x_1^{(m)} \\ x_2^{(m)} \\ \vdots \\ x_n^{(m)} \end{pmatrix} = \mathbf{0}$$

$$\Leftrightarrow \quad \begin{pmatrix} \lambda_1 x_1^{(1)} + \lambda_2 x_1^{(2)} + \cdots + \lambda_m x_1^{(m)} \\ \lambda_1 x_2^{(1)} + \lambda_2 x_2^{(2)} + \cdots + \lambda_m x_2^{(m)} \\ \vdots \\ \lambda_1 x_n^{(1)} + \lambda_2 x_n^{(2)} + \cdots + \lambda_m x_n^{(m)} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$\Leftrightarrow \quad \begin{pmatrix} x_1^{(1)} & x_1^{(2)} & \cdots & x_1^{(m)} \\ x_2^{(1)} & x_2^{(2)} & \cdots & x_2^{(m)} \\ \vdots & & & \\ x_n^{(1)} & x_n^{(2)} & \cdots & x_n^{(m)} \end{pmatrix} \cdot \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$\Leftrightarrow \quad \begin{pmatrix} \uparrow & \uparrow & \cdots & \uparrow \\ x^{(1)} & x^{(2)} & \cdots & x^{(m)} \\ \downarrow & \downarrow & \cdots & \downarrow \end{pmatrix} \cdot \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{pmatrix} = \mathbf{0}$$

In the process of the above transformation we used tacitly the definition of the matrix product.

2.2.11. Quiz

In [3]:
```
from ipywidgets import widgets, Layout, Box, GridspecLayout
from File4MCQ import create_multipleChoice_widget
```

In [5]:

```
#test = create_multipleChoice_widget("1. Question: What day of the week is it today the 1st of September 2020?", ['a. Monday', 'b. Tuesday', 'c. Wednesday'],'b. Tuesday','[Hint]:')
#test
```

2.3. Matrices

Definition of matrices, matrix addition, multiplication by a scalar of matrices

$n \times m$-dimensional matrices are elements of the set $\mathrm{R}^{n \times m}$.

We organise the elements of an $n \times m$-dimensional matrix in n rows and m columns.

For the notation of matrices we use often capital letters of the alphabet.

For a matrix $X \in \mathrm{R}^{n \times m}$ let us denote the element at the intersection of the ith row and jth column by $x_{i,j}$. Then we can define the matrix by its compenents in the following way

$$X = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,m} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,m} \\ \vdots & & & \\ x_{n,1} & x_{n,2} & \cdots & x_{n,m} \end{pmatrix}$$

Matrix addition and **multiplication by a scalar** happens component-wise, exactly as in the case of vectors. The sum of the matrices $X = (x_{i,j})_{i=\overline{1,n},j=\overline{1,m}} \in \mathrm{R}^{n \times m}$ and $Y = (y_{i,j})_{i=\overline{1,n},j=\overline{1,m}} \in \mathrm{R}^{n \times m}$ is the matrix

$$X + Y = (x_{i,j} + y_{i,j})_{i=\overline{1,n},j=\overline{1,m}},$$

that is

$$X + Y = \begin{pmatrix} x_{1,1} + y_{1,1} & x_{1,2} + y_{1,2} & \cdots & x_{1,m} + y_{1,m} \\ x_{2,1} + y_{2,1} & x_{2,2} + y_{2,2} & \cdots & x_{2,m} + y_{2,m} \\ \vdots & & & \\ x_{n,1} + y_{n,1} & x_{n,2} + y_{n,2} & \cdots & x_{n,m} + y_{n,m} \end{pmatrix}$$

For $\lambda \in \mathrm{R}$ and $X = (x_{i,j})_{i=\overline{1,n},j=\overline{1,m}} \in \mathrm{R}^{n \times m}$ we define the mutiplication of X by the scalar λ as the matrix

$$\lambda X = (\lambda x_{i,j})_{i=\overline{1,n},j=\overline{1,m}}$$

that is

$$\lambda X = \begin{pmatrix} \lambda x_{1,1} & \lambda x_{1,2} & \cdots & \lambda x_{1,m} \\ \lambda x_{2,1} & \lambda x_{2,2} & \cdots & \lambda x_{2,m} \\ \vdots & & & \\ \lambda x_{n,1} & \lambda x_{n,2} & \cdots & \lambda x_{n,m} \end{pmatrix}$$

2.3.1. Matrix multiplication

Matrix multiplication of a vector as a linear transformation that transforms basis vectors of the original space to basis vectors of the image space.

2.3.2. Gauss elimination to solve a system of linear equations

2.3.3. Inverse of a matrix by Gaussian elimination

2.3.4. The determinant of a 2×2 matrix

as the volume of the paralellogram spanned by the colmn vectors of the original matrix.

What means if the determinant is 0?

Ex. for a 3x3 system with a multiple solution.

2.3.5. Rotation in a different coordinate system than the canonical one

2.3.6. Orthogonal matrices

2.3.7. Gram-Schmidt orthogonalisation

2.3.8 Reflection in $\mathbf{R}^3$ w.r.t. a plane

2.3.9. Eigenvectors, eigenvalues

"eigen" = "characteristic" Eigenvectors are the vectors, which are just scaled by a factor when appliying the matrix operation on them. Eigenvalues are the solutions of characteristic polynomial.

Rotation in R^2: no eigenvector

Rotation in R^3: the only eigenvector is the axis of rotation

Scaling along one axis: 2 eigenvectors

Identity matrix: Every vector is an eigenvector

2.3.10. Diagonalisation by changing the basis to the eigenvectors

Application to calculating the nth power of a matrix. We can be interested in this question when T is s transition matrix encorporating the change that happens in one time unit. Then T^n shows the change happening in n time units. If we can find a basis, where the matrix is diagonal, then calculate its nth power and afterwards transform it back, it is easier than calculating the nth power of the original matrix.

In [1]:

```
test = create_multipleChoice_widget("1. Question: What day
of the week is it today?", ['a. Monday', 'b. Tuesday', 'c.
Wednesday'],'b. Tuesday','[Hint]:')
test
```

```
------------------------------------------------------------
----------------
NameError                                 Traceback (most r
ecent call last)
<ipython-input-1-f79ed3bf0b3b> in <module>
----> 1 test = create_multipleChoice_widget("1. Question: W
hat day of the week is it today?", ['a. Monday', 'b. Tuesda
y', 'c. Wednesday'],'b. Tuesday','[Hint]:')
      2 test

NameError: name 'create_multipleChoice_widget' is not defin
ed
```

2.3.11. Eigenvectors, eigenvalues

"eigen" = "characteristic" Eigenvectors are the vectors, which are just scaled by a factor when appliying the matrix operation on them. Eigenvalues are the solutions of characteristic polynomial.

Please associate the following transformations in R^2 to the number of eigenvectors that they have:

1. Rotation in R^2
2. Rotation in R^3
3. Scaling along one axis
4. Scaling along 2 axis
5. Multiplication by the identity matrix

Number of eigenvectors:
a. every vector is an eigenvector
b. the transformation has exactly 2 eigenvectors
c. the transformation has exactly one eigenvectors
d. the transformation can have none or two eigenvectors

2.3.9. Diagonalisation by changing the basis to the eigenvectors

Application to calculating the nth power of a matrix. We can be interested in this question when T is s transition matrix encorporating the change that happens in one time unit. Then T^n shows the change happening in n time units. If we can find a basis, where the matrix is diagonal, then calculate its nth power and afterwards transform it back, it is easier than calculating the nth power of the original matrix.

3. Homework

Read through the material of the first day and summarise the relevant formulas, notions on a cheat sheet.