Analytic Geometry

Introduction

Euclidian space is a fundamental concept in mathematics. It is defined by a set of axioms (attributed to ancient Greek mathematician Euclid, 300 BC).

- [Euclidean geometry](#)
- [Hilbert's axioms](#)

In Euclidean space, we have few basic concepts:

Primitive terms

- Points
- Lines
- Planes

Primitive relations

- Betweenness
- Lies on
- Congruence

Using just axioms of Euclidean geometry it is very hard proving new theorems. Thanks to [Decartes](#) we can introduce a new concept called Cartesian space. This new approach allows us to use numbers and algebra to describe the objects and relations in the Euclidean space.

Analytic geometry is a branch of mathematics that deals with the study of geometry concepts like points, lines, curves, surfaces, and many more, in the context of a Cartesian coordinate system. Analytic geometry allows us to describe geometric objects using algebraic equations. It provides a powerful tool for solving geometric problems by transforming them into algebraic problems.

Now we will build a [Cartesian space](#) step by step.
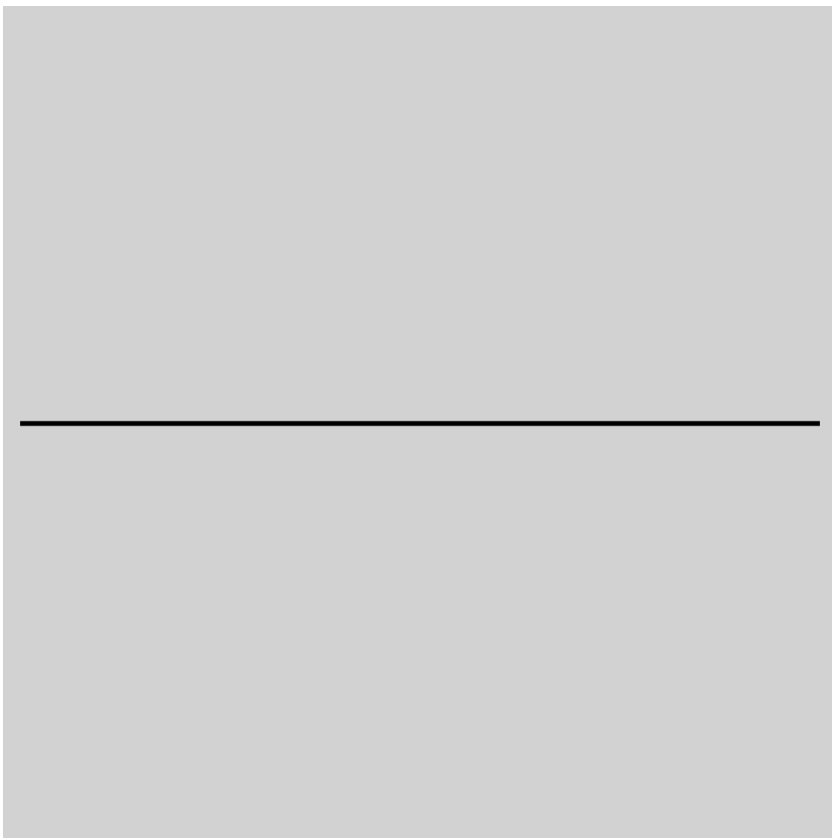
Building a Cartesian space

Let's take a Euclidean plane

This is pure Euclidean plane, without any structures (lines, circles, etc.).
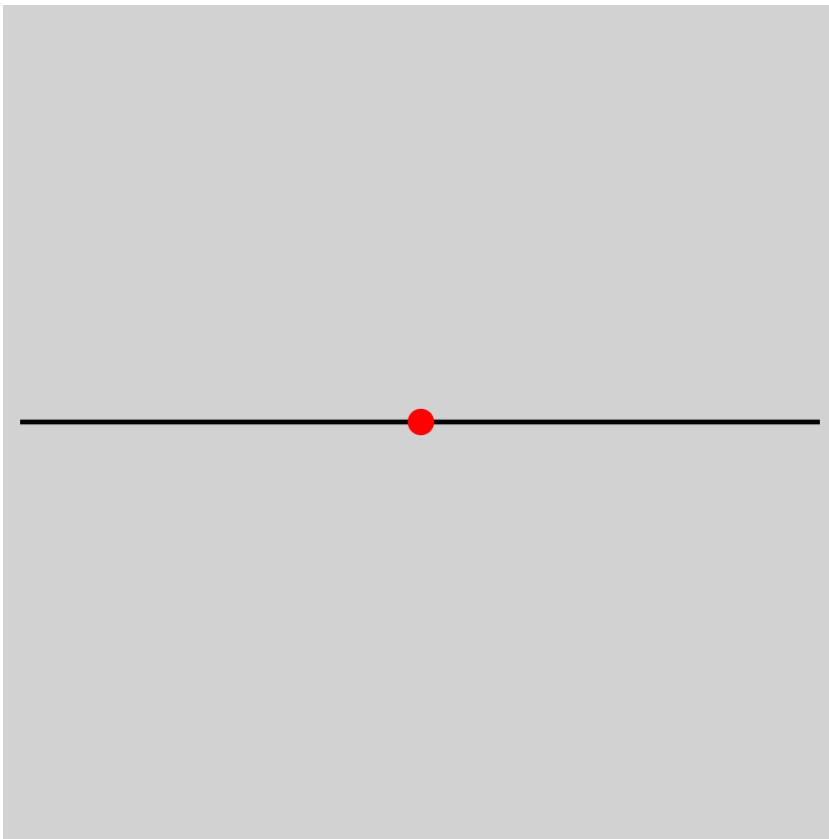
Draw a line

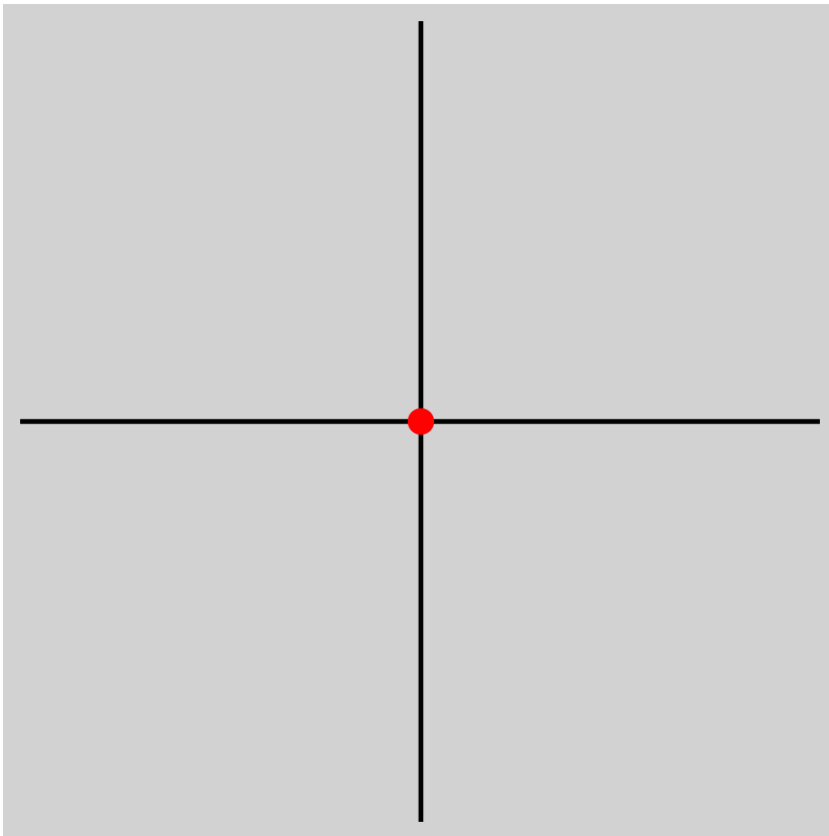Let's draw a line. We will call it x.

Draw a point on the line

You can choose any point on the line.

Draw a perpendicular line

By axioms of Euclidean geometry, you can draw a perpendicular line to the first line which passes through the chosen point. This line is unique and it passes through the chosen point. We will call this line y.

A red point is called the origin of the Cartesian space. These lines are called the axes of the Cartesian space.

Defining the unit length

Now, you can choose second point on x-axis different from the origin. Distance between these two points will be called the unit length of the Cartesian space.

By Euclidean axioms, you can fill the whole x-axis with points separated by the unit length.

Also, by the axioms of Euclidean geometry, you can find a point on the y-axis such that the distance between this point and the origin is equal to the unit length, and further points on the y-axis are separated by the unit length.

By doing this, we introduce numbers to the Euclidian space.

Those numbers describe the position of the points on the each axis.
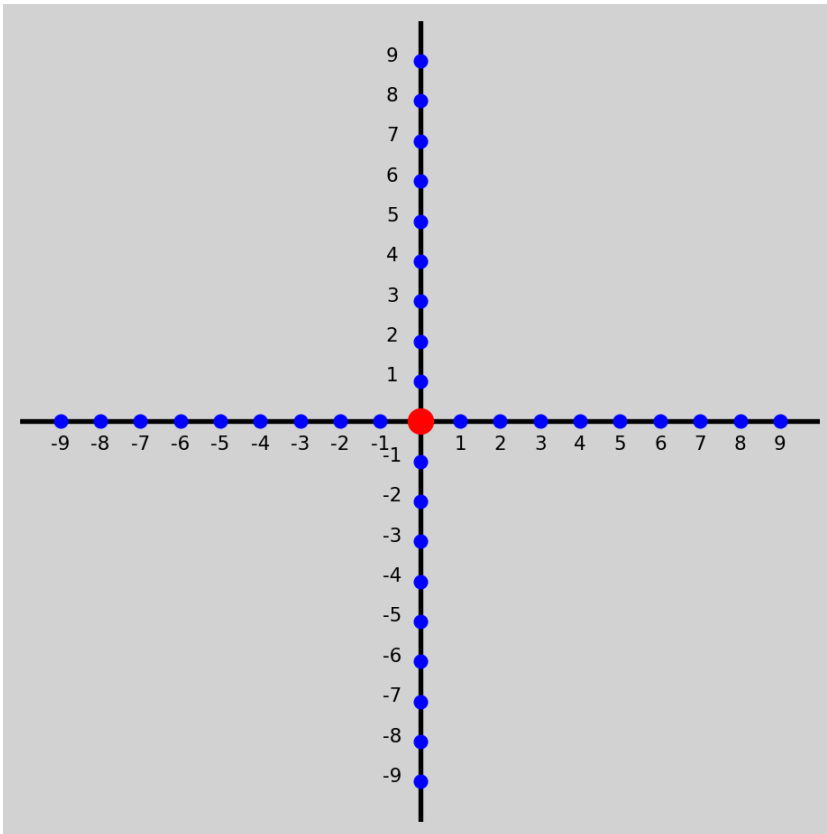
Describing the position of any point

Let's consider any point P on the plane. From the axioms of Euclidean geometry, we know that there exists a unique line passing through this point that is perpendicular to the first axis. Similarly, there is a unique line that is perpendicular to the second axis and passes through point P. These lines intersect the axes at unique points A and B, respectively. Since, for any point P, there are unique points A and B on the axes, we can define the position of P by these numbers. We call these numbers the coordinates of point P.

We have just defined the **Cartesian coordinate system**.
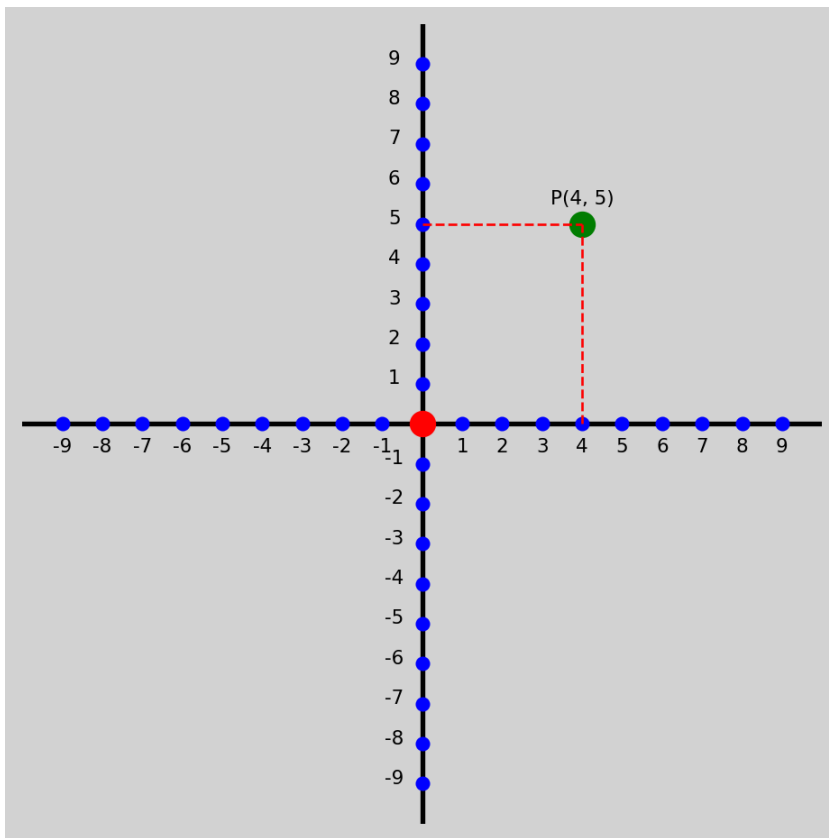
Summary

We have built a Cartesian plane. We have introduced numbers to the Euclidean plane. We can describe the position of any point on a plane by two numbers.

Now we will investigate how we can use algebra to describe the relations between points in the Euclidean space.

Cartesian plane

Below we present some basic terms and operations in the Cartesian space.

Distance between two points

The distance between two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ is given by the formula:

$$d(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Midpoint of a line segment

The midpoint of a line segment with endpoints $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ is given by the formula:

$$M(P_1, P_2) = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2} \right)$$

Addition of two points

The sum of two points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ is given by the formula:

$$P_1 + P_2 = (x_1 + x_2, y_1 + y_2)$$

Subtraction of two points

The difference of two points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ is given by the formula:
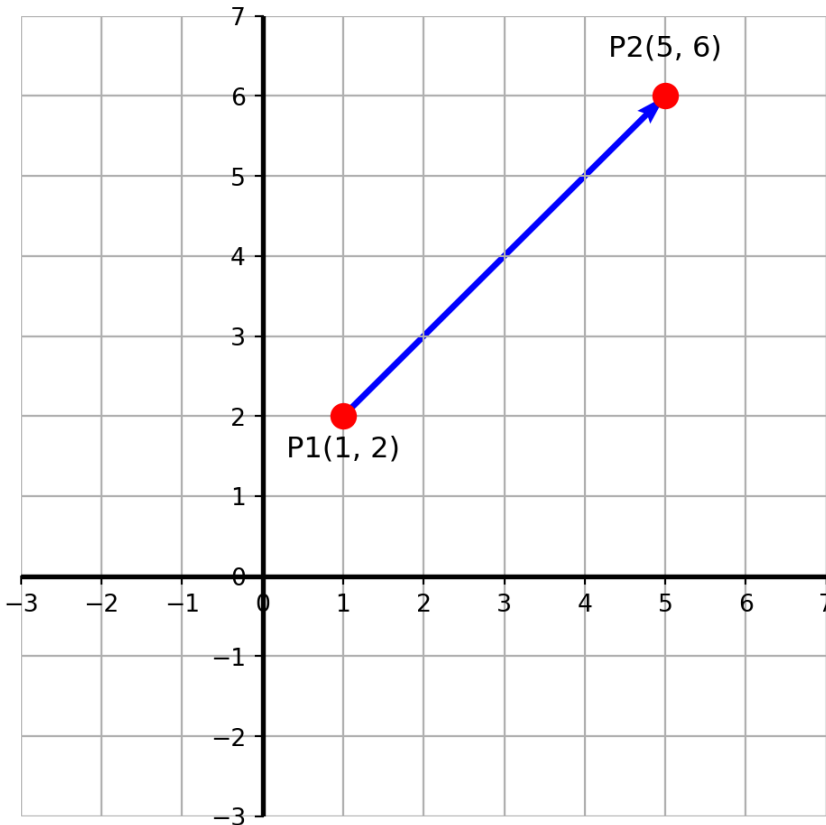
$$P_1 - P_2 = (x_1 - x_2, y_1 - y_2)$$

Multiplication of a point by a scalar

The multiplication of a point $P(x, y)$ by a scalar k is given by the formula:

$$k \cdot P = (k \cdot x, k \cdot y)$$

Vector

Ordered two points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ define a **vector**. On plane vector is represented by an arrow pointing from P_1 to P_2.

Vectors are very useful in geometry. They can also be used to represent physical quantities like velocity, force, acceleration, electric field, etc.

Higher dimensional Cartesian space

In the case of a plane, we observe that two numbers are required to describe the position of any point, which is why we refer to it as two-dimensional space. We can extend this concept further by introducing a new axis perpendicular to the existing ones. In this way, we define three-dimensional Cartesian space.

By continuing this process, we can define four-dimensional Cartesian space, and so on. Consequently, in n-dimensional space, n numbers are needed to describe the position of any point.

We will be discussing two-dimensional and three-dimensional Cartesian spaces in more detail. Conventionally, we denote:

- in 2D, each point can be written as (x, y).
- in 3D, points are of the form (x, y, z).

Vectors

A vector can be represented with an arrow between points. It is denoted by arrow above the symbol $\vec{a}$ or in bold font $\mathbf{a}$.

From the definition of a vector, we can describe its atributes:

- **Magnitude** is a segment length between the points defining the vector. It is denoted by $|\mathbf{a}|$.
- **Direction** is encoded in the order of the points defining the vector. We can also use angles to describe the direction of the vector.
- **Coordinates** are the differences between the coordinates of the points defining the vector. For example, for a vector $\mathbf{a}$ defined by points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$, the coordinates of the vector are $[x_2 - x_1, y_2 - y_1]$.

Unit vector is a vector with a magnitude equal to 1. For convenience, the unit vector is often denoted by "hat", i.e. $\hat{\mathbf{a}}$ (pronounced "a-hat").

Any non-zero vector $\mathbf{a}$ can be transformed into a unit vector $\hat{\mathbf{a}}$ by dividing it by its magnitude
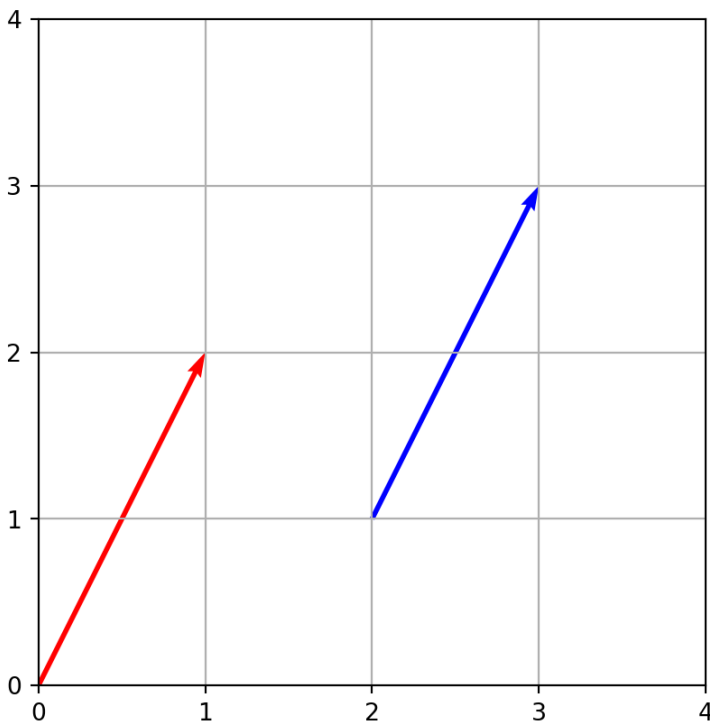
$$\mathbf{a} \quad \rightarrow \quad \hat{\mathbf{a}} = \frac{\mathbf{a}}{|\mathbf{a}|}$$

Example:

The vector $\mathbf{a} = [3, 4]$ has a magnitude $|\mathbf{a}| = 5$, and the corresponding unit vector (the unit vector along the direction of vector $\mathbf{a}$) is expressed as $\hat{\mathbf{a}} = \left[\frac{3}{5}, \frac{4}{5}\right]$.

Affine Space

Two vectors can have the same direction and magnitude but originate from different points.

In terms of their coordinates, these vectors will appear identical.

- Red vector: $\mathbf{a} = [1 - 0, 2 - 0] = [1, 2]$

- Blue vector: $\mathbf{b} = [3 - 2, 3 - 1] = [1, 2]$

Literature, often 'hand waves' here and uses the following terms:

- A **fixed vector** originates from a specific point, typically the origin, and is defined by its endpoint coordinates.

- A **free vector** is defined solely by its magnitude and direction, independent of its starting point.

To properly address this subtle distinction, one has to introduce the concept of an affine space. In an affine space, vectors can be freely translated without altering their intrinsic properties. It is quite challenging, and goes beyond the scope of this course. It is crucial to be cautious when working with definitions that rely on vectors anchored at the origin of the coordinate system. This might be the source of many errors while working with vectors in computer systems.

Basis

The **basis** of vectors is a set of vectors that spans a vector space. In Cartesian space, the standard basis vectors are $\hat{\mathbf{i}}, \hat{\mathbf{j}}, \hat{\mathbf{k}}$.

For 2D:

$$\hat{\mathbf{i}} = [1, 0], \quad \hat{\mathbf{j}} = [0, 1]$$

For 3D:

$$\hat{\mathbf{i}} = [1, 0, 0], \quad \hat{\mathbf{j}} = [0, 1, 0], \quad \hat{\mathbf{k}} = [0, 0, 1]$$

Note that basis vectors are unit vectors along the axes and are perpendicular to each other.

Vectors can be represented using components in the Cartesian coordinate system, where each component corresponds to the distance of the point from the respective coordinate axis.

- For a vector in two-dimensional space: $\mathbf{a} = [a_x, a_y] = a_x\hat{\mathbf{i}} + a_y\hat{\mathbf{j}}$
- For a vector in three-dimensional space: $\mathbf{a} = [a_x, a_y, a_z] = a_x\hat{\mathbf{i}} + a_y\hat{\mathbf{j}} + a_z\hat{\mathbf{k}}$
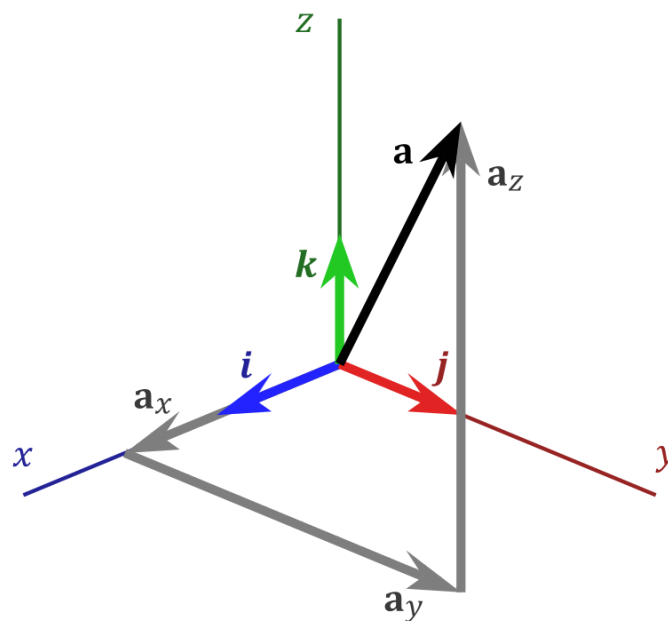
Different coordinate systems that can be used to represent vectors:

- **Cartesian Coordinate System** is used in 2D and 3D and defines a point by its distance from the origin along the x, y, and z axes.

- **Polar Coordinate System** is used in 2D and defines a point by its distance from the origin (r) and the angle (θ) relative to the reference axis. Link: Polar Coordinate System

- **Spherical Coordinate System** is used in 3D and defines a point by its distance from the origin (r), azimuth angle (θ), and elevation angle (φ). Link: Spherical Coordinate System

To find the coordinates of vector $\mathbf{v}$ in a new basis $\{\mathbf{b_1}, \mathbf{b_2}, \mathbf{b_3}\}$, express $\mathbf{v}$ as a linear combination of the basis vectors.

$$\mathbf{v} = c_1\mathbf{b_1} + c_2\mathbf{b_2} + c_3\mathbf{b_3}$$

The coefficients c_1, c_2, c_3 are the coordinates of vector $\mathbf{v}$ in the new basis.

Vector Addition

Vector addition involves adding the corresponding components of vectors of the same length.

For two vectors $\mathbf{a}$ and $\mathbf{b}$ in the space $\mathbb{R}^n$, where:

$$\mathbf{a} = [a_1, a_2, \ldots, a_n], \quad \mathbf{b} = [b_1, b_2, \ldots, b_n]$$

Vector addition is defined as follows:

$$\mathbf{a} + \mathbf{b} = [a_1 + b_1, a_2 + b_2, \ldots, a_n + b_n]$$

Scalar Multiplication of a Vector

Scalar multiplication of a vector involves multiplying each component of the vector by the same number.

For a vector $\mathbf{a}$ and a scalar k:

$$k\mathbf{a} = [ka_1, ka_2, \ldots, ka_n]$$

Vector's Length

Length of a vector is the magnitude of the vector. It is calculated as the square root of the sum of the squares of the components of the vector.

$$|\mathbf{a}| = \sqrt{\sum_{i=1}^{n} a_i^2} = \sqrt{a_1^2 + a_2^2 + \cdots + a_n^2}$$

Vectors in 2D:

If $\mathbf{a} = [a_x, a_y]$, then the magnitude is given by the formula $|\mathbf{a}| = a = \sqrt{a_x^2 + a_y^2}$.

Vectors in 3D:

If $\mathbf{a} = [a_x, a_y, a_z]$, then the magnitude of the vector is $|\mathbf{a}| = a = \sqrt{a_x^2 + a_y^2 + a_z^2}$.

Plotting a Vector

```python
import matplotlib.pyplot as plt # import the matplotlib.pyplot module

plt.figure() # create a new figure
plt.quiver(0, 0, 3, 4, angles='xy', scale_units='xy', scale=1, color='r') # quick way to plot a
plt.xlim(0, 5) # set x-axis range
plt.ylim(0, 5) # set y-axis range
```

```
plt.grid(True) # enable grid
plt.gca().set_aspect('equal') # set equal scaling on x and y axes
plt.show()
```
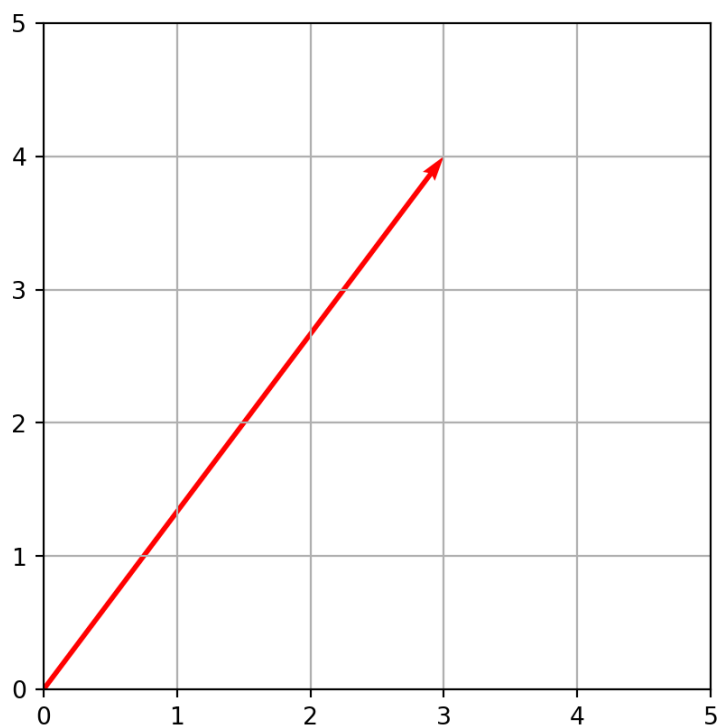

More complicated example:

```
import matplotlib.pyplot as plt # import the matplotlib.pyplot module #example of vector additi

plt.figure() # create a new figure
plt.quiver(0, 0, 2, 1, angles='xy', scale_units='xy', scale=1, color='r') # quick way to plot v
plt.quiver(0, 0, 1, 2, angles='xy', scale_units='xy', scale=1, color='b') # quick way to plot v
plt.quiver(0, 0, 3, 3, angles='xy', scale_units='xy', scale=1, color='g') # quick way to plot v
plt.quiver(1, 2, 2, 1, angles='xy', scale_units='xy', scale=1, color='y') # quick way to plot v
plt.quiver(2, 1, 1, 2, angles='xy', scale_units='xy', scale=1, color='m') # quick way to plot v
plt.xlim(-1, 4) # set x-axis range
plt.ylim(-1, 4) # set y-axis range
plt.grid(True) # enable grid
plt.gca().set_aspect('equal') # set equal scaling on x and y axes
plt.show()
```
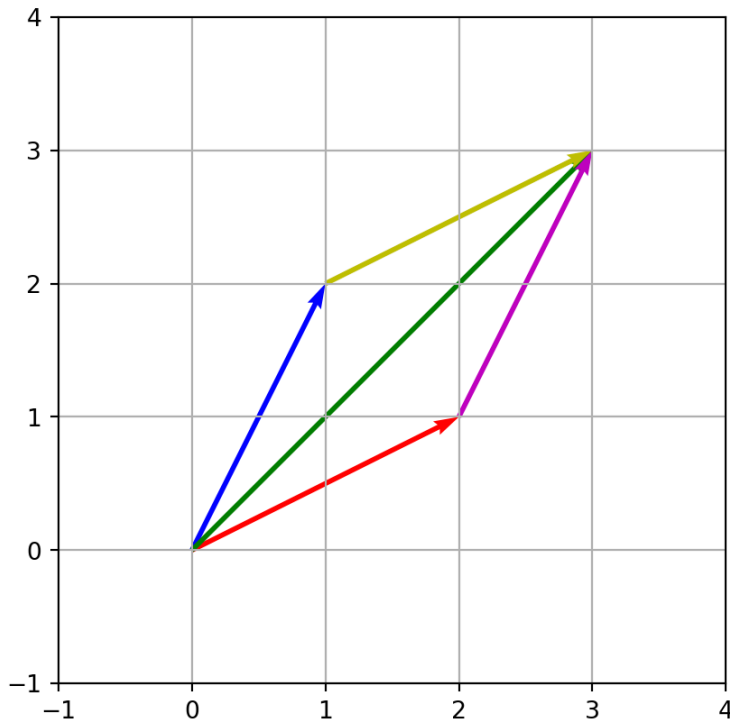
Scalar Product

The scalar product (also called the dot product) is an operation that takes two vectors and returns a scalar value. It is defined as the sum of the products of the corresponding components of two vectors.

For two vectors $\mathbf{a}$ and $\mathbf{b}$ in the space $\mathbb{R}^n$, where:

$$\mathbf{a} = [a_1, a_2, \ldots, a_n], \quad \mathbf{b} = [b_1, b_2, \ldots, b_n]$$

The scalar product is defined as follows

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n = \sum_{i=1}^{n} a_i b_i$$

In three-dimensional space ($\mathbb{R}^3$) the scalar product of vectors $\mathbf{a} = [a_1, a_2, a_3]$ and $\mathbf{b} = [b_1, b_2, b_3]$ is defined as:

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + a_3 b_3$$

The scalar product returns a number that measures the "similarity of direction" between two vectors.

If $\mathbf{a} \cdot \mathbf{b} = 0$, the vectors are perpendicular, which is denoted as $\mathbf{a} \perp \mathbf{b}$.

Alternatively, the scalar product can be expressed as:

$$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}||\mathbf{b}| \cos(\theta),$$

which is the product of the magnitudes of vectors $\mathbf{a}$ and $\mathbf{b}$ multiplied by the cosine of the angle between them.

Scalar product is commutative, i.e., $\mathbf{a} \cdot \mathbf{b} = \mathbf{b} \cdot \mathbf{a}$.

The scalar product can be used to find the length of a vector:

$$|\mathbf{a}| = \sqrt{\mathbf{a} \cdot \mathbf{a}}$$

Cross Product (Vector Product)

The cross product is an operation that returns a new vector perpendicular to the two input vectors. It is defined only in three-dimensional space $\mathbb{R}^3$ (and ... seven-dimensional space).

For two vectors $\mathbf{a} = [a_1, a_2, a_3]$ and $\mathbf{b} = [b_1, b_2, b_3]$, the vector product $\mathbf{a} \times \mathbf{b}$ is defined as:

$$\mathbf{a} \times \mathbf{b} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix}$$

Where $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are unit vectors along the x, y, z axes. Interestingly, to calculate the vector product and obtain the resulting vector, we use the determinant (a scalar) of the matrix (a second-order tensor) whose rows are the basis vectors (vectors) and the components of the vectors (scalars).

The vector product can be expressed in expanded form:

$$\mathbf{a} \times \mathbf{b} = (a_2 b_3 - a_3 b_2)\mathbf{i} - (a_1 b_3 - a_3 b_1)\mathbf{j} + (a_1 b_2 - a_2 b_1)\mathbf{k}$$

The result of the vector product is a new vector:

$$\mathbf{a} \times \mathbf{b} = \mathbf{c} = [c_1, c_2, c_3] = [a_2 b_3 - a_3 b_2, a_3 b_1 - a_1 b_3, a_1 b_2 - a_2 b_1]$$

The vector product returns a vector $\mathbf{c}$ perpendicular to the plane defined by $\mathbf{a}$ and $\mathbf{b}$, and its magnitude is equal to the area of the parallelogram whose sides are defined by the vectors $\mathbf{a}$ and $\mathbf{b}$

$$|\mathbf{c}| = |\mathbf{a}||\mathbf{b}| \sin\theta$$

where θ is the angle between the vectors. The direction of vector $\mathbf{c}$ is determined by the right-hand rule.

Mixed Product

The mixed product of three vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ is defined as:

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$$

The mixed product is equal to zero if the vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ are collinear or parallel.

The volume of the parallelepiped spanned by three vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ is equal to the absolute value of the mixed product of these vectors.

Applications

Calculating the Angle Between Vectors

$$\cos\theta = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}||\mathbf{b}|}$$

Checking Vector Perpendicularity

Vectors $\mathbf{a}$ and $\mathbf{b}$ are perpendicular if their dot product is zero:

$$\mathbf{a} \cdot \mathbf{b} = 0 \quad \Leftrightarrow \quad \mathbf{a} \perp \mathbf{b}$$

Finding Projections of Vectors onto Other Vectors

$$\text{Projection} = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{b}|}$$

Calculating Work Done by a Force in Physics

For a force $\mathbf{F} = (F_x, F_y, F_z)$ acting along a displacement $\mathbf{d} = (d_x, d_y, d_z)$, work is calculated as:

$$W = \mathbf{F} \cdot \mathbf{d} = |\mathbf{F}||\mathbf{d}| \cos \theta = F_x d_x + F_y d_y + F_z d_z$$

Torque

In physics: torque is the vector product of the force vector and the arm of the force:

$$\mathbf{M} = \mathbf{r} \times \mathbf{F}$$

Calculating the Area of a Parallelogram

The area of a parallelogram spanned by two vectors:

$$\text{Area}_p = |\mathbf{a} \times \mathbf{b}| = |\mathbf{a}||\mathbf{b}| \sin \theta$$

Calculating the Area of a Triangle

The area of a triangle spanned by two vectors $\mathbf{a}$ and $\mathbf{b}$:

$$\text{Area}_t = \frac{1}{2}|\mathbf{a} \times \mathbf{b}|$$

Calculating the Volume of Parallelipeds

The mixed product of vectors is used to calculate the volume of parallelepipeds. For three vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$, the volume of the parallelepiped is given by:

$$V = |\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})|$$

(Note: **| |** here denotes the absolute value, not the length!)

Calculating the Volume of Tetrahedrons

The volume of a tetrahedron spanned by three vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$ is given by:

$$V = \frac{1}{6}|\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})|$$

(Note: **| |** here denotes the absolute value, not the length!)

Checking Coplanarity of three 3D vectors

If the mixed product of three 3Dvectors is equal to zero, it means that the vectors are coplanar (lie in the same plane, which make them linearly dependent).

Introduction to GeoGebra

This document introduces essential GeoGebra functionalities for creating and manipulating objects such as points, segments, vectors, functions, curves, parameterizations, sliders, intersections, perpendicularity, and tangency.

`Text with purple font color indicates the input to be entered in the GeoGebra input field.`

Points and Their Components

Creating a Point

- **Using the Point Tool**: Select the **Point** tool and click on the Graphics View to place a point.
- **Using the Input Field**: Enter `A = (3, 4)` to create point A at coordinates (3, 4). You can define point just by specifying its coordinates like `(2,3)` Geogebra will automatically assign a name to the point. Try to write `B =`

`(h,k)` in the input field and see what happens.

- **X-Coordinate**: Use `x(A)` to get the x-coordinate of point A.
- **Y-Coordinate**: Use `y(A)` to get the y-coordinate of point A.

Segments

Creating a Segment

- **Using the Segment Tool**: Select the **Segment** tool and click on two points to create a segment between them.
- **Using the Input Field**: Create two points `A=(1,4)` and `B=(-2,2)`, then enter `Segment[A, B]` to create a segment between points A and B.

Vectors

Creating a Vector

- **Using the Vector Tool**: Select the **Vector** tool, then click on two points to create a vector directed from the first point to the second.
- **Using the Input Field**:
  - Create two points, `A=(1,4)` and `B=(-2,2)`, and then enter `Vector(A, B)` to create a vector from point A to point B.
  - Define a vector by its components, for example, `v = (3, 2)`. This will create a vector from the **origin** `(0,0)` to the point `(3,2)`. Note that we use a lowercase `v` to define the vector; using an uppercase `V` will create a point instead.

Accessing Vector Components

- **X-Component**: Use `x(v)` to get the x-component of vector v.
- **Y-Component**: Use `y(v)` to get the y-component of vector v.

Length

- **Length**: Use `Length(v)` to get the length of vector v.

Matrices

Creating a Matrix

- **Using the Input Field**: Enter `M = {{1, 2}, {3, 4}}` to create a 2x2 matrix M. If you don't specify the name of the matrix, GeoGebra will assign a default name like `m1` ect.

Operations

- **Element Access**: Use `M(i, j)` to access the element in the i-th row and j-th column of matrix M.
- **Multiplication**: Use `M * N` to multiply matrices M and N.
- **Row Access**: Use `M(i)` to access the i-th row of matrix M.
- **Determinant**: Use `Det(M)` to calculate the determinant of matrix M.
- **Transpose**: Use `Transpose(M)` to find the transpose of matrix M.

Apply Matrix on a Vector

- **Using the Input Field**: Define a vector `v = (1, 2)` and a matrix `M = {{1, 2}, {3, 4}}`.
- Entering `M * v` will create a **point** `(5, 11)`.
- If you use `ApplyMatrix(M, v)`, this will create a **vector** from the origin to the point `(5, 11)`.

Functions

Defining a Function

- Enter `f(x) = x^2 + 3x + 5` to define a function f.

- **Value at a Point**: Use `f(a)` to find the value of function f at point a.
- **Extremum**: Use `Extremum(f,a,b)` to find the extremum of function f in the interval [a, b].
- **Derivative**: Use `f'(x)` to calculate the derivative of function f.
- **Indefinite Integral**: Use `Integral(f)` to calculate the indefinite integral of function f with respect to x.
- **Definite Integral**: Use `Integral(f, a, b)` to find the integral of function f from a to b.
- **Intersection with Axis x**: Use `Root(f)` to find the x-intercepts of function f.
- **Intersection with Axis y**: Use `f(0)` to find the y-intercept of function f.
- **Tangent Line**: Use `Tangent(a,f)` to create the tangent line to function f at point `(a,f(a))`.
- **Point of Intersection between Two Functions**: Use `Intersect(f, g, a,b)` to find the intersection points of functions f and g in the interval [a, b].

Sliders

- **Creating a Slider**: Use the **Slider** tool to create a slider.
- **Using a Slider**: Sliders can control parameters in functions, vectors, or any variable, enabling interactive models.
  - Example: Define a parameter `a` as a slider, then use `f(x) = a * x^2` to observe how changing `a` affects the graph.

Curves

Creating a Parametric Curve

- Use the **Curve** tool or enter `Curve(expression, parameter, start, end)` in the input field.
- Example: `c=Curve(3 * cos(t), 2 * sin(t), t, 0, 2π)` creates an ellipse.

Simulate points on a curve

- Create slider `s` by entering just `s` in the input field.
- Define a point `P` on the curve using `P = (x(c(s)), y(c(s)))`, where **x(c(s))** and **y(c(s))** are the x and y components of the curve **c** at parameter **s**.

Intersections of functions

- **Finding Intersection Points**:
  - Use the **Input Field** to enter the functions `f(x)` and `g(x)` and `Intersect(f, g, a, b)` where `a` and `b` are the interval limits in which to search for intersections.

Algebraic Objects in GeoGebra

Lines

In GeoGebra, lines can be represented in various forms depending on the equation and approach used. Here are some common equations for lines and how they can be implemented in GeoGebra:

1. Slope-Intercept Form

The slope-intercept form of a line is given by the equation:

$$y = mx + b$$

where m is the slope and b is the y-intercept.

- **Implementation in GeoGebra**: Enter `y = m * x + b` directly into the input field. Varible `m` represents the slope, and `b` represents the y-intercept. Because we did not specify the values of `m` and `b`, GeoGebra will treat them as variables and create a dynamic slider that can be manipulated.

2. Point-Slope Form

The point-slope form of a line is given by:

$$y - y_1 = m(x - x_1)$$

where (x_1, y_1) is a point on the line and m is the slope.

- **Implementation in GeoGebra**: First define a pont A using `A = (1,2)` and then enter the equation of the line using the point-slope form: `y - y(A) = m * (x - x(A))` in the input field. Because we did not specify the value of `m`, GeoGebra will create a slider for the slope.

3. General Form

The general form of a line is:

$$Ax + By + C = 0$$

where A, B, and C are constants.

- **Implementation in GeoGebra**: Input `A * x + B * y + C = 0`, and GeoGebra will treat A, B, and C as variables that can be adjusted using sliders.

4. Parametric Form

A line can also be described using a parameter t as follows:

$$x = x_0 + t \cdot v_x$$
$$y = y_0 + t \cdot v_y$$

where (x_0, y_0) is a point on the line, and (v_x, v_y) is a vector parallel to the line (direction vector).

- **Implementation in GeoGebra**: Define a point A using `A = (1, 2)`. Use `Curve[x(A) + t * a, y(A) + t * b, t, -10, 10]`, becase we did not specify the values of `a` and `b`, GeoGebra will create sliders for them.

Second Degree Curves

In GeoGebra, second-degree curves, such as circles, ellipses, parabolas, and hyperbolas, can be represented using various forms. Here are common equations for these curves and instructions for implementing them in GeoGebra.

1. Circle

The equation of a circle with center (h, k) and radius r is:

$$(x - h)^2 + (y - k)^2 = r^2$$

- **Implementation in GeoGebra**: Define the center point using `A = (h, k)` and input `(x - x(A))^2 + (y - y(A))^2 = r^2`. If r is not defined, GeoGebra will create a slider for r to dynamically adjust the radius.

2. Ellipse

The general form of an ellipse centered at (h, k) with semi-major axis a and semi-minor axis b is:

$$\frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2} = 1$$

- **Implementation in GeoGebra**: Define the center point using `A = (h, k)`. Then input `((x - x(A))^2 / a^2) + ((y - y(A))^2 / b^2) = 1`. If a and b are not defined, GeoGebra will automatically create sliders for these values, allowing dynamic resizing of the ellipse.

3. Parabola

The equation of a parabola opening vertically with vertex (h, k) and parameter p (distance from the vertex to the focus) is:

$$(y - k) = \frac{1}{4p}(x - h)^2$$

- **Implementation in GeoGebra**: Define the vertex point using `A = (h, k)`. Enter `y - y(A) = (1 / (4 * p)) * (x - x(A))^2` into the input field. If p is undefined, GeoGebra will generate a slider for p, which allows adjusting the parabola's shape.

4. Hyperbola

The standard form of a hyperbola centered at (h, k), with transverse axis length $2a$ and conjugate axis length $2b$, is:

$$\frac{(x - h)^2}{a^2} - \frac{(y - k)^2}{b^2} = 1$$

- **Implementation in GeoGebra**: Define the center point using `A = (h, k)`. Input `((x - x(A))^2 / a^2) - ((y - y(A))^2 / b^2) = 1`. GeoGebra will create sliders for a and b if they are not defined, enabling dynamic manipulation of the hyperbola's axes.

These implementations provide flexible ways to create and explore second-degree curves in GeoGebra, offering a dynamic approach to studying the properties of circles, ellipses, parabolas, and hyperbolas.

5. General Form of Conic Sections

The general form of a conic section is `Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0`. Try to enter this equation in the input field and see what happens.

3D Lines and Planes

In GeoGebra 3D, we can also define lines and planes using equations and points in space. Below are common ways to represent these objects and instructions for implementing them in GeoGebra 3D.

1. Line in 3D

A line in 3D space can be represented parametrically with a point (x_0, y_0, z_0) on the line and a direction vector (v_x, v_y, v_z):

$$x = x_0 + t \cdot v_x$$
$$y = y_0 + t \cdot v_y$$
$$z = z_0 + t \cdot v_z$$

where t is a parameter that extends along the direction vector.

- **Implementation in GeoGebra 3D**: Define the starting point using `A = (x_0, y_0, z_0)` and direction components `v_x`, `v_y`, and `v_z`. Enter `Curve[x(A) + t * v_x, y(A) + t * v_y, z(A) + t * v_z, t, -10, 10]`. If v_x, v_y, and v_z are not specified, GeoGebra will create sliders for these components.

2. Plane

A plane can be defined by a point (x_0, y_0, z_0) on the plane and a normal vector (n_x, n_y, n_z) perpendicular to it. The equation of the plane is:

$$n_x(x - x_0) + n_y(y - y_0) + n_z(z - z_0) = 0$$

Alternatively, the plane can also be expressed in general form:

$$Ax + By + Cz + D = 0$$

where A, B, C are the components of the normal vector, and D is a constant.

- **Implementation in GeoGebra 3D**: Define a point `A = (x_0, y_0, z_0)` and the components of the normal vector `n_x`, `n_y`, and `n_z`. Enter the plane equation as `n_x * (x - x(A)) + n_y * (y - y(A)) + n_z * (z - z(A)) = 0`. Alternatively, use the general form by defining variables A, B, C, and D and entering `A * x + B * y + C * z + D = 0`. If any constants are not defined, GeoGebra will create sliders for them.

3D Curves and Surfaces

In GeoGebra 3D, second-degree surfaces such as spheres, ellipsoids, paraboloids, and hyperboloids, as well as 3D parametric curves, can be explored interactively.

1. Sphere

The equation of a sphere with center (h, k, l) and radius r is:

$$(x - h)^2 + (y - k)^2 + (z - l)^2 = r^2$$

- **Implementation in GeoGebra 3D**: Define the center point `A = (h, k, l)`, then input `(x - x(A))^2 + (y - y(A))^2 + (z - z(A))^2 = r^2`. GeoGebra will create a slider for r if it is not defined, allowing dynamic adjustment of the sphere's radius.

2. Ellipsoid

The equation of an ellipsoid centered at (h, k, l) with semi-axes a, b, and c along the x, y, and z axes respectively is:

$$\frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2} + \frac{(z - l)^2}{c^2} = 1$$

- **Implementation in GeoGebra 3D**: Define the center point `A = (h, k, l)`, then input `((x - x(A))^2 / a^2) + ((y - y(A))^2 / b^2) + ((z - z(A))^2 / c^2) = 1`. GeoGebra will automatically create sliders for a, b, and c if they are not defined, enabling dynamic resizing of the ellipsoid along each axis.

3. Paraboloid

A paraboloid with vertex at (h, k, l) and parameter p (opening along the z-axis) is given by:

$$z - l = \frac{1}{4p}\left((x - h)^2 + (y - k)^2\right)$$

- **Implementation in GeoGebra 3D**: Define the vertex point `A = (h, k, l)`. Then input `z - z(A) = (1 / (4 * p)) * ((x - x(A))^2 + (y - y(A))^2)`. GeoGebra will create a slider for p if it is not defined, allowing adjustment of the paraboloid's curvature.

4. Hyperboloid of One Sheet

The equation of a hyperboloid of one sheet centered at (h, k, l) with semi-axes a, b, and c is:

$$\frac{(x - h)^2}{a^2} + \frac{(y - k)^2}{b^2} - \frac{(z - l)^2}{c^2} = 1$$

- **Implementation in GeoGebra 3D**: Define the center point `A = (h, k, l)`. Enter `((x - x(A))^2 / a^2) + ((y - y(A))^2 / b^2) - ((z - z(A))^2 / c^2) = 1`. GeoGebra will create sliders for a, b, and c if they are not specified, enabling interactive manipulation of the hyperboloid's dimensions.

5. Parametric Curve in 3D

A 3D curve can be defined parametrically with a parameter t, as follows:

$$x = f(t)$$
$$y = g(t)$$
$$z = h(t)$$

where $f(t)$, $g(t)$, and $h(t)$ define the x, y, and z components of the curve.

- **Implementation in GeoGebra 3D**: Use `Curve[f(t), g(t), h(t), t, t_{min}, t_{max}]`, replacing $f(t)$, $g(t)$, and $h(t)$ with specific functions of t. For example, `Curve[cos(t), sin(t), t, t, -10, 10]` defines a helical curve.

These equations enable flexible creation and exploration of 3D lines, planes, curves, and surfaces in GeoGebra, allowing dynamic visualization and manipulation in three-dimensional space.

6. General Form of second order surfaces

The general form of a second-order surface is `Ax^2 + By^2 + Cz^2 + Dxy + Exz + Fyz + Gx + Hy + Iz + J = 0`. Try to enter this equation in the input field and see what happens.