

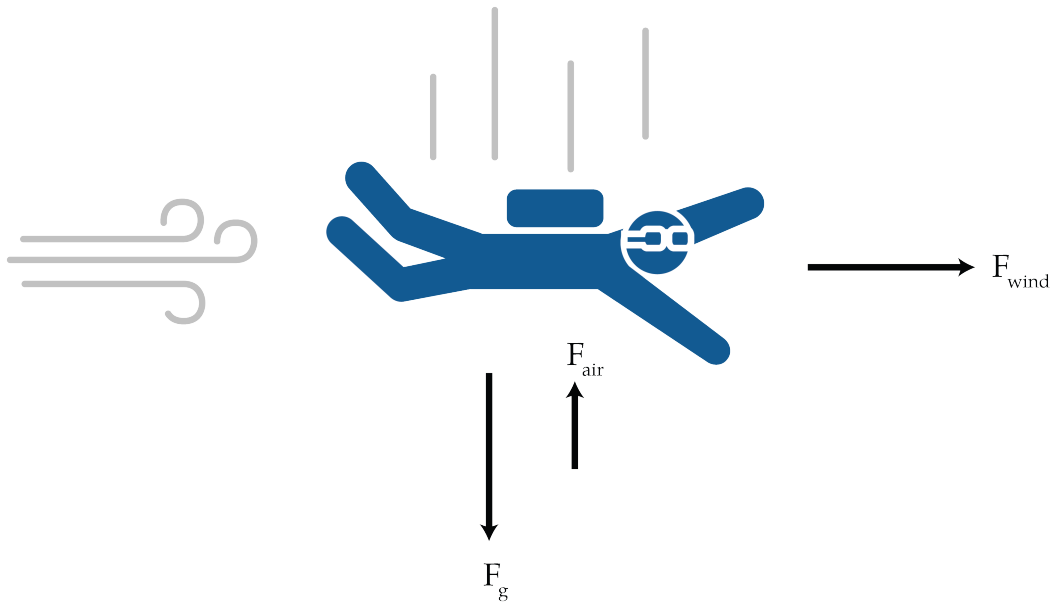


CHAPTER 1

Vectors

We have talked a some about forces, but in the calculations that we have done, we have only talked about the magnitude of a force. It is equally important to talk about its direction. To do the math on things with a magnitude and a direction (like forces), we need vectors.

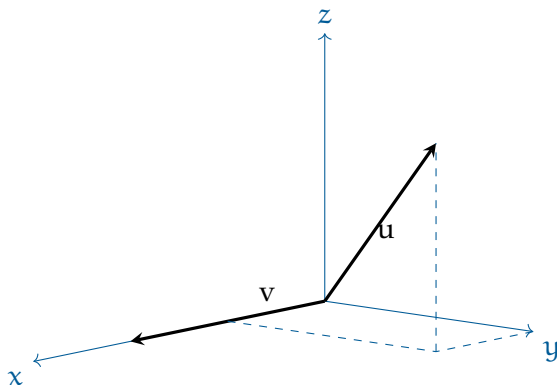
For example, if you jump out of a plane (hopefully with a parachute), several forces with different magnitudes and directions will be acting upon you. Gravity will push you straight down. That force will be proportional to your weight. If there were a wind from the west, it would push you toward the east. That force will be proportional to the square of the speed of the wind and approximately proportional to your size. Once you are falling, there will be resistance from the air that you are pushing through – that force will point in the opposite direction from the direction you are moving and will be proportional to the square of your speed.



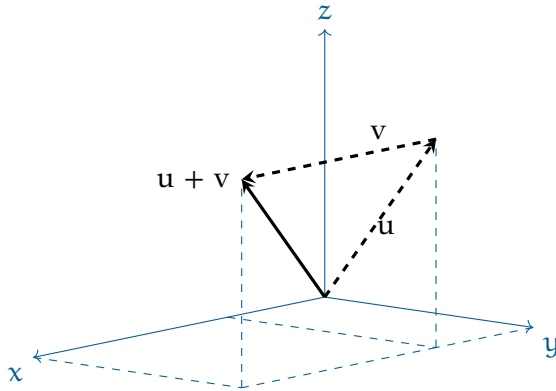
To figure out the net force (which will tell us how we will accelerate), we will need to add these forces together. So we need to learn to do math with vectors.

1.1 Adding Vectors

A vector is typically represented as a list of numbers, with each number representing a particular dimension. For example, if I am creating a 3-dimensional vector representing a force, it will have three numbers representing the amount of force in each of the three axes. For example, if a force of one newton is in the direction of the x -axis, I might represent the vector as $v = [1, 0, 0]$. Another vector might be $u = [0.5, 0.9, 0.7]$



Thinking visually, when we add to vectors, we put the starting point second vector at the ending point of the first vector.



If you know the vectors, you will just add them element-wise:

$$\mathbf{u} + \mathbf{v} = [0.5, 0.9, 0.7] + [1.0, 0.0, 0.0] = [1.5, 0.9, 0.7]$$

These vectors have 3 components, so we say they are *3-dimensional*. Vectors can have any number of components. For example, the vector $[-12.2, 3, \pi, 10000]$ is 4-dimensional.

You can only add two vectors if they have the same dimension.

$$[12, -4] + [-1, 5] = [11, 1]$$

Addition is commutative: If you have two vectors \mathbf{a} and \mathbf{b} , then $\mathbf{a} + \mathbf{b}$ is the same as $\mathbf{b} + \mathbf{a}$.

Addition is also associative: If you have three vectors \mathbf{a} , \mathbf{b} , and \mathbf{c} , it doesn't matter which order you add them in. That is, $\mathbf{a} + (\mathbf{b} + \mathbf{c}) = (\mathbf{a} + \mathbf{b}) + \mathbf{c}$.

A 1-dimensional vector is just a number. We say it is a *scalar*, not a vector.

Exercise 1 Adding vectors

Add the following vectors:

- $[1, 2, 3] + [4, 5, 6]$
- $[-1, -2, -3, -4] + [4, 5, 6, 7]$
- $[\pi, 0, 0] + [0, \pi, 0] + [0, 0, \pi]$

Working Space

Answer on Page 11

Exercise 2 Adding Forces

You are adrift in space. You are near two different stars. The gravity of one star is pulling you towards it with a force of $[4.2, 5.6, 9.0]$ newtons. The gravity of the other star is pulling you towards it with a force of $[-100.2, 30.2, -9.0]$ newtons. What is the net force?

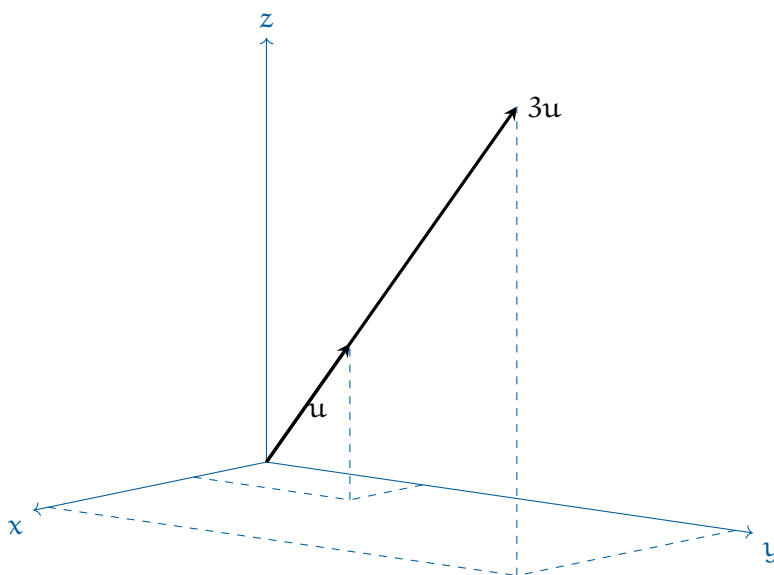
Working Space

Answer on Page 11

1.2 Multiplying a vector with a scalar

It is not uncommon to multiply a vector by a scalar. For example, a rocket engine might have a force vector v . If you fire 9 engines in the exact same direction, the resulting force vector would be $9v$.

Visually, when we multiply a vector u by a scalar a , we get a new vector that goes in the same direction as u but has a magnitude a times as long as u .



When you multiply a vector by a scalar, you just multiply each of the components by the scalar:

$$3 \times [0.5, 0.9, 0.7] = [1.5, 2.7, 3.6]$$

Exercise 3 Multiplying a vector and a scalar

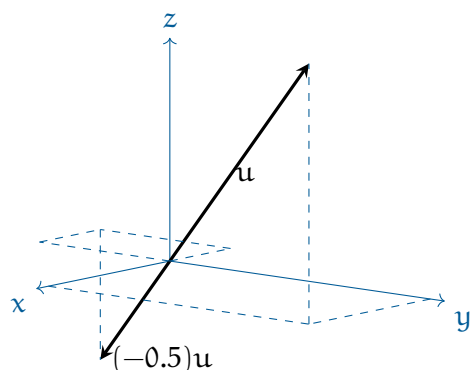
Simplify the following expressions:

- $2 \times [1, 2, 3]$
- $[-1, -2, -3, -4] \times -2$
- $\pi[\pi, 2\pi, 3\pi]$

Working Space

Answer on Page 12

Note that when you multiply a vector times a negative number, the new vector points in the opposite direction.



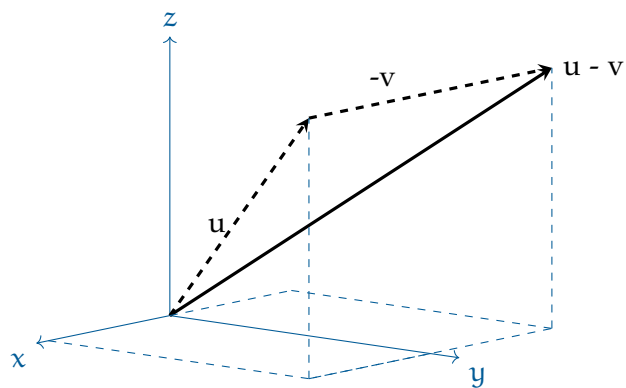
1.3 Vector Subtraction

As you might guess, when you subtract one vector from another, you just do element-wise subtraction:

$$[4, 2, 0] - [3, -2, 9] = [1, 4, -9]$$

So, $u - v = u + (-1v)$.

So visually, you reverse the one that is being subtracted:



1.4 Magnitude of a Vector

The *magnitude* of a vector is just its length. We write the magnitude of a vector v as $|v|$.

We compute the magnitude using the pythagorean theorem. If $v = [3, 4, 5]$, then

$$|v| = \sqrt{3^2 + 4^2 + 5^2} = \sqrt{50} \approx 7.07$$

(You might notice that the notation for the magnitude is exactly like the notation for absolute value. If you think of a scalar as a 1-dimensional vector, the absolute value and the magnitude are the same. For example, the absolute value of -5 is 5. If you take the magnitude of the one-dimensional vector $[-5]$, you get $\sqrt{25} = 5$.)

Notice that if you scale up a vector, its magnitude scales by the same amount. For example:

$$|7[3, 4, 5]| = 7\sqrt{50} \approx 7 \times 7.07$$

The rule then is: If you have any vector v and any scalar a :

$$|av| = |a||v|$$

Exercise 4 Magnitude of a Vector

Find the magnitude of the following vectors:

- $[1, 1, 1]$
- $[-5, -5, -5]$ (that is the same as $-5 \times [1, 1, 1]$)
- $[3, 4, -4] + [-2, -3, 5]$

Working Space

Answer on Page 12

1.5 Vectors in Python

NumPy is a library that allows you to work with vectors in Python. You might need to install it on your computer. This is done with `pip`. `pip3` installs things specifically for Python 3.

```
pip3 install NumPy
```

We can think of a vector as a list of numbers. There are also grids of numbers known as *matrices*. NumPy deals with both in the same way, so it refers to both of them as arrays.

The study of vectors and matrices is known as *Linear Algebra*. Some of the functions we need are in a sublibrary of NumPy called `linalg`.

As a convention, everyone who uses NumPy, imports it as *np*.

Create a file called `first_vectors.py`:

```
import NumPy as np

# Create two vectors
v = np.array([2,3,4])
u = np.array([-1,-2,3])
print(f"u = {u}, v = {v}")

# Add them
w = v + u
print(f"u + v = {w}")

# Multiply by a scalar
w = v * 3
print(f"v * 3 = {w}")

# Get the magnitude
# Get the magnitude
mv = np.linalg.norm(v)
mu = np.linalg.norm(u)
print(f"|v| = {mv}, |u| = {mu}")
```

When you run it, you should see:

```
> python3 first_vectors.py
u = [-1 -2  3], v = [2 3 4]
u + v = [1 1 7]
v * 3 = [ 6  9 12]
|v| = 5.385164807134504, |u| = 3.7416573867739413
```

1.5.1 Formatting Floats

The numbers 5.385164807134504 and 3.7416573867739413 are pretty long. You probably want it rounded off after a couple of decimal places.

Numbers with decimal places are called *floats*. In the placeholder for your float, you can specify how you want it formatted, including the number of decimal places.

Change the last line to look like this:

```
print(f"|v| = {mv:.2f}, |u| = {mu:.2f}")
```

When you run the code, it will be neatly rounded off to two decimal places:

```
|v| = 5.39, |u| = 3.74
```

This is a draft chapter from the Kontinua Project. Please see our website (<https://kontinua.org/>) for more details.



APPENDIX A

Answers to Exercises

Answer to Exercise 1 (on page 4)

- $[1, 2, 3] + [4, 5, 6] = [5, 7, 9]$
- $[-1, -2, -3, -4] + [4, 5, 6, 7] = [3, 3, 3, 3]$
- $[\pi, 0, 0] + [0, \pi, 0] + [0, 0, \pi] = [\pi, \pi, \pi]$

Answer to Exercise 2 (on page 4)

To get the net force, you add the two forces:

$$\mathbf{F} = [4.2, 5.6, 9.0] + [-100.2, 30.2, -9.0] = [-96, 35.8, 0.0] \text{ newtons}$$

Answer to Exercise 3 (on page 5)

- $2 \times [1, 2, 3] = [2, 4, 6]$
- $[-1, -2, -3, -4] \times -3 = [3, 6, 9, 12]$
- $\pi[\pi, 2\pi, 3\pi] = \pi^2, 2\pi^2, 3\pi^2]$

Answer to Exercise 4 (on page 7)

- $||[1, 1, 1]|| = \sqrt{3} \approx 1.73$
- $||[-5, -5, -5]|| = |-5 \times [1, 1, 1]| = 5\sqrt{3} \approx 8.66$
- $||[3, 4, 5] + [-2, -3, -4]|| = |[1, 1, 1]| = \sqrt{3} \approx 1.73$



INDEX

floats
 formatting, [9](#)

linalg, [8](#)

np, [8](#)
NumPy, [8](#)

vectors, [1](#)
 adding, [2](#)
 in python, [7](#)
 magnitude of, [6](#)
 multiplying by a scalar, [4](#)
 subtraction, [6](#)