Calculus tutorial

Excerpt from the **No bullshit guide to math and physics** by Ivan Savov

Abstract—aaa

CONTENTS

I. Introduction

Calculus = study of functions + operations on functions

Calculus is the study of the properties of functions. The operations of calculus are used to describe the limit behaviour of functions, calculate their rates of change, and calculate the areas under their graphs. In this section we'll learn about the `SymPy` functions for calculating limits, derivatives, integrals, and summations.

previously = numbers + operations that produce other numbers as outputs in calculus = functions + operators that produce other functions as outputs

In high school math, we learn all kinds of math procedures for solving problems using a finite number of steps of math operations. Whether you're manipulating expressions using algebra, or applying the inverse function to simplify an equation, all problems in high school math can be solved by using less than five steps, or if your teacher really doesn't like you 10 steps. In calculus, we learn a broader class of problem-solving strategies that include procedures with an infinite number of steps.

A. Definitions

- $\mathbb{R}$: the set of real numbers
- $f(x)$: a function of the form $f : \mathbb{R} \to \mathbb{R}$, which means f takes real numbers as inputs and produces real numbers as outputs
- $\lim_{\delta \to 0}$: a limit expression in which the number δ tends to zero
- $f'(x)$: the derivative of $f(x)$ is the rate of change of f at x:

$$f'(x) \stackrel{\text{def}}{=} \lim_{\delta \to 0} \frac{f(x+\delta) \; - \; f(x)}{\delta} \, .$$

  The derivative is a function of the form $f' : \mathbb{R} \to \mathbb{R}$.
- $A(a,b)$: the value of the *area* under the curve $f(x)$ from $x = a$ until $x = b$. The area $A(a,b)$ is computed as the following integral

$$A(a,b) = \int_a^b f(x) \; dx.$$

  The $\int$ sign stands for *sum*. Indeed, the integral is the "sum" of $f(x)$ for all values of x between a and b.

- $A_0(x)$: the *integral function* of $f(x)$. The integral function corresponds to the computation of the area under $f(x)$ as a function of the upper limit of integration:

$$A_0(x) \stackrel{\text{def}}{=} A(0,x) = \int_0^x f(u) \; du.$$

  The choice of $x = 0$ as the lower limit of integration is arbitrary.

- sequence
- series

B. Example 1: file download

Suppose you're downloading a large file to your computer. At $t = 0$ you click "save as" in your browser and the download starts. Let $f(t)$ represent the size of the downloaded data. At any time t, the function $f(t)$ tells you the amount of disk space taken by the partially-downloaded file. You are downloading a 720[MB] file, so the download progress at time t corresponds to the fraction $\frac{f(t)}{720[\text{MB}]}$.

Download rate: The derivative function $f'(t)$, pronounced "f prime," describes how the function $f(t)$ changes over time. In our example $f'(t)$ is the download speed. If your downloading speed is $f'(t) = 100[\text{kB/s}]$, then the file size $f(t)$ must increase by 100[kB] each second. If you maintain this download speed, the file size will grow at a constant rate: $f(0) = 0[\text{kB}]$, $f(1) = 100[\text{kB}]$, $f(2) = 200[\text{kB}]$, ..., $f(100) = 10[\text{MB}]$.

To calculate the "estimated time remaining" until the download's completion, we divide the amount of data that remains to be downloaded by the current download speed:

$$\text{time remaining } = \frac{720 - f(t)}{f'(t)} \quad [\text{s}].$$

The bigger the derivative, the faster the download will finish. If your internet connection were 10 times faster, the download would finish 10 times more quickly.

Inverse problem: Let's consider this situation from the point of view of the modem that connects your computer to the internet. Any data you download comes through the modem. The modem knows the download rate $f'(t)[\text{kB/s}]$ at all times during the download.

However, since the modem is separate from your computer, it does not know the file size $f(t)$ as the download progresses. Nevertheless, the modem can infer the file size at time t from knowing the transmission rate $f'(t)$. The integral of the download rate between $t = 0$ and $t = \tau$ corresponds to the total amount of downloaded data stored on your computer. During this download period, the change in file size is described by the integral

$$\Delta f = f(\tau) - f(0) = \int_0^\tau f'(t) \, dt \, .$$

Assuming the file size starts from zero $f(0) = 0[\text{kB}]$ at $t = 0$, the modem can use the integration procedure to find $f(\tau)$, the file size on your computer at $t = \tau$:

$$f(\tau) = \int_0^\tau f'(t) dt \, .$$

The download rate $f'(t)$ is measured in [kB/s], and each time step dt is 1[s] long, so the data downloaded during one second is $f'(t)dt[\text{kB}]$. The file size at time $t = \tau$ is equal to the sum of the data downloaded during each second from $t = 0$ until $t = \tau$.

The integral $\int_a^b q(t) \, dt$ is the calculation of the *total* of some quantity $q(t)$ that accumulates during the time period from $t = a$ to $t = b$. Integrals are necessary any time you want to calculate the total of a quantity that changes over time.

As demonstrated above, calculus is much more than the theoretical activity reserved for math specialists. Calculus relates to everyday notions you're already familiar with. Indeed, we carry out calculus-like operations in our head every day—we just don't necessarily use calculus terminology when we do so.

Derivatives and integrals are two main actors in calculus. The other supporting actors are limits and series, which we'll briefly introduce in the following example.

C. Example 2: Euler's number

Suppose you take out a loan with a 100% annual interest rate. If the interest is calculated once per year, this means at the end of the year you'll owe double the amount you borrowed. However, if the interest rate is calculated more often, you'll owe more at the end of the year. For example, if the interest rate is computed twice per year, you'll owe $(1 + \frac{1}{2})(1 + \frac{1}{2}) = 2.25$. If computed three times per year, $(1 + \frac{1}{3})(1 + \frac{1}{3})(1 + \frac{1}{3}) = 2.370$. In general, when the compounding is performed n times per year, the amount owed at the end of the year will be

$$\underbrace{\left(1 + \tfrac{1}{n}\right)\left(1 + \tfrac{1}{n}\right)\cdots\left(1 + \tfrac{1}{n}\right)}_{n \text{ times}} = \left(1 + \tfrac{1}{n}\right)^n.$$

Euler's number is defined as the *limit* of the expression $\left(1 + \frac{1}{n}\right)^n$ as n grows to infinity, which we write as

$$e \overset{\text{def}}{=} \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^n = 2.718281828\ldots.$$

This limit expression describes the annual growth rate of a loan with infinitely frequent compounding. If we borrow \$1000, we'll owe \$2718.28 at the end of one year.

Euler's number e can also be obtained from another expression:

$$e \overset{\text{def}}{=} 1 + 1 + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \frac{1}{5!} + \cdots = \sum_{k=0}^{\infty} \frac{1}{k!} = 2.718281828\ldots.$$

Note this is a math expression with an infinite number of terms. Each term is comes from a common "pattern" $\frac{1}{k!}$, where $k! = k \cdot (k-1) \cdot (k-2) \cdots 3 \cdot 2 \cdot 1$ is the factorial function. This kind of expression is called a *series*, and provides a powerful way to compute quantities by summing together a bunch of terms.

D. Applications

Learning the language of calculus will help you think more clearly about certain types of problems. Understanding the language of calculus is *essential* for learning science because many laws of nature are best described in terms of derivatives and integrals.

Using concepts to describe phenomena in: physics, engineering, finance, probability calculations (continuous random variables), optimization (used in business, statistics, machine learning, etc.).

E. Doing calculus

Symbolic calculations using pen and paper

Symbolic calculations using SymPy

Numerical calculations using NumPy and SciPy

II. Math prerequisites

A. Notation for sets and intervals

Sets are arbitrary collections of math objects. Many math ideas are expressed using the language of sets, so it's worth going over the basic definitions and notation conventions.

- S, T: the usual variable names for sets
- $s \in S$: this statement is read "s is an element of S" or "s is in S"
- { definition }: the curly brackets surround the definition of a set, and the expression inside the curly brackets describes what the set contains.
- $\mathbb{N}$: the set of natural numbers $\mathbb{N} \stackrel{\text{def}}{=} \{0, 1, 2, \ldots\}$
- $\mathbb{N}_+$: the set of positive natural numbers $\mathbb{N}_+ \stackrel{\text{def}}{=} \{1, 2, 3 \ldots\}$.
- $\mathbb{R}$: the set of real numbers.
- $\mathbb{R}_+$: the set of nonnegative real numbers. The definition of the nonnegative is written as $\mathbb{R}_+ \stackrel{\text{def}}{=}$ {all x in $\mathbb{R}$ such that $x \geq 0$}, or it can be expressed more compactly as $\mathbb{R}_+ \stackrel{\text{def}}{=} \{x \in \mathbb{R} \mid x \geq 0\}$.

We often define sets using the *set-builder* notation $\{\cdot \mid \cdot\}$. Inside the curly brackets we first describe the general kind of mathematical objects we are talking about, followed by the symbol "|" (read "such that"), followed by the conditions that must be satisfied by all elements of the set. The definitions of nonnegative real numbers $\mathbb{R}_+$ above is an example of the set-builder notation.

The *number line* is a visual representation of the set of real numbers $\mathbb{R}$, as shown in Figure 1. The real numbers correspond to all the points on the number line, from $-\infty$ to ∞.
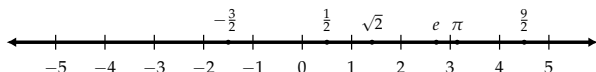

Fig. 1. The real numbers $\mathbb{R}$ cover the entire number line.

The set of real numbers includes all the rational numbers like $-\frac{3}{2}$, $\frac{1}{2}$, and $\frac{9}{2}$, as well as irrational numbers like $\sqrt{2}$, e, and π. This means any number you are likely to run into when solving math problems can be visualized as a point on the number line.

Number intervals: The number line can also be used to represent subsets of the real numbers, which we call *intervals*. Figure 2 shows an illustration of the interval $[2, 4] = \{x \in \mathbb{R} \mid 2 \leq x \leq 4\}$, which is a subset of the real numbers.

Here are some more examples of various intervals:

- $[a, b]$: the interval from a to b. This corresponds to the set of real numbers between a and b, including the endpoints a and b. The interval $[a, b]$ corresponds to the set $\{x \in \mathbb{R} \mid a \leq x \leq b\}$.
- $[a, \infty)$: the interval from a until infinity, which corresponds to the set $\{x \in \mathbb{R} \mid a \leq x\}$.
- $(-\infty, b]$: the interval from negative infinity until b, which corresponds to the set $\{x \in \mathbb{R} \mid x \leq b\}$.

The notation $[a, b]$ describes the *closed* interval from a to b, which means the endpoints a and b are included in the interval. The notation (a, b) describes the *open* interval from a to b, defined as the set $\{x \in \mathbb{R} \mid a < x < b\}$, which doesn't include the endpoints a and b. In other words, intervals defined using square brackets "[" include the endpoints (defined using less-than-or-equal conditions) while intervals defined with round brackets "(" do not include their endpoints (defined using strictly-less-than conditions). The distinction between open and closed intervals is important in general, but makes no difference in the context of probability theory, so you don't need to worry about the difference between $[a, b]$ and (a, b) in this book.
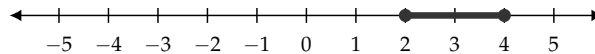

Fig. 2. The interval $[2, 4] \stackrel{\text{def}}{=} \{x \in \mathbb{R} \mid 2 \leq x \leq 4\}$.

I hope these definitions and examples made you feel more comfortable with sets, and the weird-looking curly bracket notation that mathematicians use to define sets. It might look a little complicated at first, but you'll get used to it in the rest of the book.

B. Functions

A *function* is a mathematical object that takes numbers as inputs and produces numbers as outputs. For every input x, the output value of f for that input is denoted $f(x)$. For example, the function $f(x) = \frac{1}{2}x^2$ takes any number x as input, squares it and divides the result by two to produce the output. For example, $f(3) = \frac{1}{2}3^2 = \frac{9}{2} = 4.5$. Here is the Python code that defines the function `f` and evaluates it for the input $x = 3$.

```
>>> def f(x):
        return x**2 / 2
>>> f(3)
4.5
```

We'll

Function graph: The *graph* of a function is a line that passes through all input-output pairs of a function. Imagine we take out a piece of paper and draw a coordinate system with a horizontal axis and a vertical axis. The horizontal axis describes the different input values x, while the vertical axis describes the output values $f(x)$. Each input-output pair of the function f corresponds to the point $(x, f(x))$ in the coordinate system. We obtain the graph of the function by varying the input coordinate x and plotting all the points $(x, f(x))$, as illustrated in Figure **??**.

The graph of the function f allows us to see at a glance the behaviour of the function for all possible inputs, and forms an essential visualization tool. Indeed, many phenomena and calculations related to functions can be understood geometrically as operations based on the graph of the function.

Plotting function graphs: We can use a combination of the `numpy` and `seaborn` modules to plot the graph of the function $f(x) = \frac{1}{2}x^2$, as shown in the code example below.

```
>>> import numpy as np
>>> import seaborn as sns
>>> xs = np.linspace(-3, 3, 1000)
>>> fxs = f(xs)
>>> sns.lineplot(x=xs, y=fxs)
See Figure 3 for the output.
```

We import the module `numpy` under the alias `np`. We use the function `np.linspace` to create an array (a list of numbers) `xs`,

which contains 1000 input values that range from $x = -3$ until $x = 3$. Next we apply the function f to the array of inputs `xs` and store the result in the array `fxs`, which contains all the output values of the function for the input values `xs`. At this point, the arrays `xs` and `fxs` contain 1000 input-output pairs of the form $(x, f(x))$, which is exactly what we need to plot the graph of the function. On the last line, we call the function `lineplot` to create the graph of $f(x)$, which produces the plot shown in Figure 3.
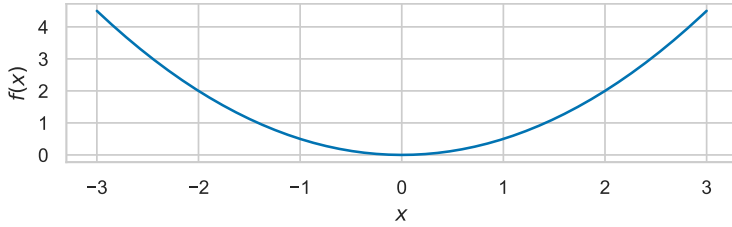

Fig. 3. Graph of the function $f(x) = \frac{1}{2}x^2$ from $x = -3$ until $x = +3$. The graph of the function f consists of all the coordinate pairs $(x, f(x))$ over some interval of x values.

Inverse functions: The inverse function $f^{-1} \colon B \to A$ performs the *inverse operation* of the function $f \colon A \to B$. If you start from some x, apply f, and then apply f^{-1}, you'll arrive—full circle—back to the original input x:

$$f^{-1}\big(\, f(x) \,\big) = x.$$

In Figure 4 the function f is represented as a forward arrow, and the inverse function f^{-1} is represented as a backward arrow that puts the value $f(x)$ back to the x it came from.
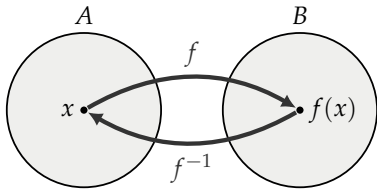

Fig. 4. The inverse f^{-1} undoes the operation of the function f.

For example, when $x \geq 0$, the inverse of the function $f(x) = \frac{1}{2}x^2$ is the function $f^{-1}(x) = \sqrt{2x}$. Earlier we computed $f(3) = 4.5$, and if we apply the inverse operation we get $f^{-1}(4.5) = \sqrt{2 \cdot 4.5} = \sqrt{9} = 3$.

```
>>> from math import sqrt
>>> sqrt(2 * 4.5)
3.0
```

The logarithmic function $\log_e(x)$ is the inverse of the exponential function e^x. If if we compute the exponential function of a number, then apply the logarithmic function, we get back the original input.

```
>>> from math import exp, log
>>> log(exp(5))
5.0
```

Function properties: It's important to keep track of the possible inputs and outputs for each function. We use the notation $f \colon A \to B$ to denote a function from the input set A to the output set B. The set of allowed inputs is called the *domain* of the function, while the set of possible outputs is called the *image* or *range* of the function. For example, the domain of the

function $f(x) = \frac{1}{2}x^2$ is $\mathbb{R}$ (any real number) and it's image is $\mathbb{R}_+$ (nonnegative real numbers), so we write it as $f \colon \mathbb{R} \to \mathbb{R}_+$.

Another important property is called *continuity*, which roughly corresponds to ability to draw a function without lifting the pen. We'll give a formal definition of continuity later in Section III-D.

C. Function inventory

Fig. 5. Graph of common functions.

D. Functions with discrete inputs

We're sometimes interested in functions with discrete inputs

$$a_k : \mathbb{N} \to \mathbb{R}$$

Sequences We can write ...

E. Geometry

The area enclosed by a circle of radius r is given by $A = \pi r^2$. A circle of radius $r = 1$ has area π.

The circumference of a circle of radius r is

$$C = 2\pi r.$$

A circle of radius $r = 1$ has circumference 2π.

The area of a rectangle of base b and height h is $A = bh$.

The area of a triangle is equal to $\frac{1}{2}$ times the length of its base times its height:

$$A = \tfrac{1}{2}ah_a.$$

Note that h_a is the height of the triangle *relative to* the side a.

(a) $A = \pi r^2$ (b) $A = bh$ (c) $A = \frac{1}{2}bh$

Fig. 6.   Area formulas for circles, rectangles, and triangles.

F. Trigonometry

III. Limits

Limit expressions provide a precise mathematical language for talking about infinitely large numbers, infinitely small lengths, and mathematical procedures with an infinite number of steps. Here are three representative examples of limit expressions:

- $\lim_{x \to \infty} f(x)$: limit expression that describes what happens to $f(x)$ when the input to the function x tends to infinity (gets larger and larger). In words, this limit expression is read as "limit of $f(x)$ as n goes to infinity."
- $\lim_{\delta \to 0} h(\delta)$: limit expression that describes what happens to $h(\delta)$ as the real number δ tends to zero. The number δ (the Greek letter delta) usually describes a small distance, and the limit as delta goes to zero ($\delta \to 0$) describes the behaviour of the expression $h(\delta)$ for an infinitely short distance δ.
- $\lim_{n \to \infty} \mathrm{proc}(n)$: limit expression that describes the value of $\mathrm{proc}(n)$ as the integer n tends to infinity. The integer n usually describes the number of steps in a given procedure, and $\mathrm{proc}(n)$ describes the output of this procedure when n steps are used.

A. Example

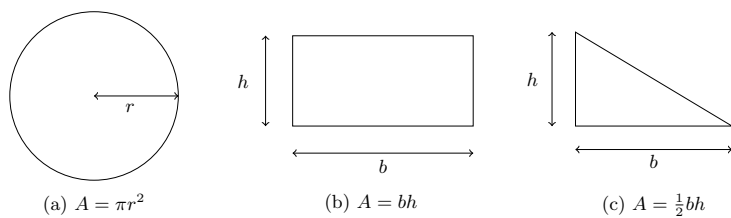The formula for the area of a regular polygon with n sides that fits inside a circle of radius r is...

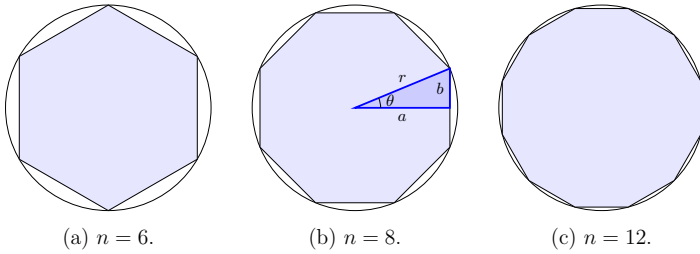(a) $n = 6$.   (b) $n = 8$.   (c) $n = 12$.

Fig. 7. A hexagon, an octagon, and a dodecagon inscribed inside a circle with radius $r = 1$.

As n goes to infinity we get πr^2, which is the formula for the area of a circle.

B. Limits at infinity

We're often interested in describing what happens to a certain function when its input variable tends to infinity. Does $f(x)$ approach a finite number, or does it keep on growing to ∞? For example, consider the limit of the function $f(x) = \frac{1}{x}$ as x goes to infinity:

$$\lim_{x \to \infty} f(x) = \lim_{x \to \infty} \frac{1}{x} = 0.$$

The function $\frac{1}{x}$ never *actually* reaches zero, so it would be wrong to write $f(x) = 0$. However, the the expression $\frac{1}{x}$ closer and closer to 0 as x goes to infinity. Limits are a useful concept because we can write $\lim_{x \to \infty} f(x) = 0$, even though $f(x) \neq 0$ for any number x.

The function $f(x)$ is said to *converge* to the number L if the function approaches the value L for large values of x:

$$\lim_{x \to \infty} f(x) = L.$$

We say "The limit of $f(x)$ as x goes to infinity is the number L." See Figure 8 for an illustration.
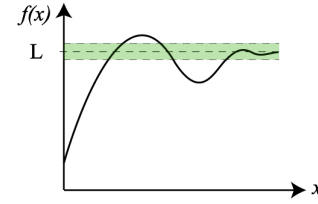

Fig. 8. A function $f(x)$ that oscillates up and down initially, but after a while "settles down" close to the value L.

Example: Calculate $\lim_{x \to \infty} \frac{2x+1}{x}$. You are given the function $f(x) = \frac{2x+1}{x}$ and must determine what the function looks like for very large values of x.

$$\lim_{x \to \infty} \frac{2x+1}{x} = \lim_{x \to \infty} \left(2 + \frac{1}{x}\right) = 2 + \lim_{x \to \infty} \left(\frac{1}{x}\right) = 2 + 0 = 2.$$

As the denominator x becomes larger and larger, the fraction $\frac{1}{x}$ becomes smaller and smaller, so $\lim_{x \to \infty} \frac{1}{x} = 0$.

C. Limits to a number

The limit of $f(x)$ approaching $x = a$ *from the right* is defined as

$$\lim_{x \to a^+} f(x) = \lim_{\delta \to 0} f(a + \delta).$$

To find the limit from the right at a, we let x take on values like $a + 0.1$, $a + 0.01$, $a + 0.001$, $a + 0.0001$, etc.

The limit of $f(x)$ when x approaches *from the left* is defined analogously,

$$\lim_{x \to a^-} f(x) = \lim_{\delta \to 0} f(a - \delta).$$

If both limits from the left and from the right of some number exist and are equal to each other, we can talk about the limit as $x \to a$ without specifying the direction of approach:

$$\lim_{x \to a} f(x) = \lim_{x \to a^+} f(x) = \lim_{x \to a^-} f(x).$$

For the two-sided limit of a function to exist at a point, both the limit from the left and the limit from the right must converge to the same number. If the function $f(x)$ obeys, $f(a) = L$ and $\lim_{x \to a} f(x) = L$, we say the function $f(x)$ is continuous at $x = a$.

D. Continuity

A function is said to be *continuous* if its graph looks like a smooth curve that doesn't make any sudden jumps and contains no gaps. If you can draw the graph of the function on a piece of paper without lifting your pen, the function is continuous.

A more mathematically precise way to define continuity is to say the function is equal to its limit for all x. We say a function $f(x)$ is *continuous* at a if the limit of f as $x \to a$ converges to $f(a)$:

$$\lim_{x \to a} f(x) = f(a).$$

Remember, the two-sided limit $\lim_{x \to a}$ requires both the left and the right limit to exist and to be equal. Thus, the definition of continuity implies the following equality:

$$\lim_{x \to a^-} f(x) = f(a) = \lim_{x \to a^+} f(x).$$

In words, this means that a function $f(x)$ is continuous at $x = a$ if the limit from the left $\lim_{x \to a^-} f(x)$ and the limit from the right $\lim_{x \to a^+} f(x)$ are both equal to the value of the function at $x = a$.

Take a moment to think about the mathematical definition of continuity at a point. Can you connect the math definition to the intuitive idea that functions are continuous if they can be drawn without lifting the pen? Most functions we'll study in calculus are continuous, but not all functions are.

E. Limit formulas

The calculation of the limit of the sum, difference, product, and quotient of two functions is computed as follows, respectively:

$$\lim_{x \to a} (f(x) + g(x)) = \lim_{x \to a} f(x) + \lim_{x \to a} g(x),$$
$$\lim_{x \to a} (f(x) - g(x)) = \lim_{x \to a} f(x) - \lim_{x \to a} g(x),$$
$$\lim_{x \to a} f(x)g(x) = \lim_{x \to a} f(x) \cdot \lim_{x \to a} g(x),$$
$$\lim_{x \to a} \frac{f(x)}{g(x)} = \frac{\lim_{x \to a} f(x)}{\lim_{x \to a} g(x)}.$$

The above formulas indicate we are allowed to *take the limit inside* of the basic arithmetic operations.

Euler's number e is defined as the limit $e \stackrel{\text{def}}{=} \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^n$, which is the compound interest calculation for an annual interest rate of 100% with compounding is performed infinitely often.

F. Computing limits using SymPy

In SymPy, we use the symbol `oo` (two lowercase os) to denote ∞. Infinity is not a number but a process: the process of counting forever. Thus, $\infty + 1 = \infty$, ∞ is greater than any finite number, and $1/\infty = 0$.

```
>>> from sympy import oo
>>> oo+1
oo
>>> 5000 < oo
True
>>> 1/oo
0
```

The SymPy function `limit` allows us to compute limit expressions. For example, here is the code for computing the limit $\lim_{x \to \infty} \frac{1}{x}$:

```
>>> import sympy as sp
>>> x = sp.symbols("x")
>>> sp.limit(1/x, x, oo)
0
```

The first line imports the `sympy` module under the alias `sp`. The second line defines the symbol `x`, which we can use to write math expressions. We provide the expression `1/x` as the first first argument to the function `limit`, then specify the variable `x` and destination `oo` as the second and third arguments.

Here is another example, that computes the limit of the fraction $\frac{2x+1}{x}$ as x goes to infinity:

```
>>> sp.limit( (2*x+1)/x , x, oo)
2
```

Recall the definition of Euler's number $e \stackrel{\text{def}}{=} \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^n$ that we showed in the introduction. We can check this definition by making SymPy compute the limit.

```
>>> n = sp.symbols("n")
>>> limit((1+1/n)**n, n, oo)
E
```

Note SymPy produced the exact value `E` $= 2.718281828\ldots$ and not an approximation.

G. Applications of limits

Limits are important in calculus because they are used in the formal definitions of derivatives, integrals, and series.

Limits for derivatives: The formal definition of a function's derivative is expressed in terms of the rise-over-run formula for an infinitely short run:

$$f'(x) = \lim_{\text{run} \to 0} \frac{\text{rise}}{\text{run}} = \lim_{\delta \to 0} \frac{f(x+\delta) - f(x)}{x + \delta - x}.$$

We'll continue the discussion of this formula in Section IV.

Limit for integrals: One way to approximate the area under the curve $f(x)$ between $x = a$ and $x = b$ is to split the area into n little rectangles of width $\Delta x = \frac{b-a}{n}$ and height $f(x)$, and then calculate the sum of the areas of the rectangles:

$$A(a,b) \approx \underbrace{\Delta x f(a) + \Delta x f(a + \Delta x) + \Delta x f(b - \Delta x)}_{n \text{ terms}}.$$

We obtain the exact value of the area in the limit where we split the area into an infinite number of rectangles with infinitely small width:

$$\int_a^b f(x)\, dx = A(a,b) = \lim_{n \to \infty} \Delta x \left[f(a) + f(a + \Delta x) + \cdots + f(b - \Delta x)\right].$$

Computing the area under a function by splitting the area into infinitely many rectangles is an approach known as a *Riemann sum*, which we'll discuss in Section V.

Limits for series: We use limits to describe the convergence properties of series. For example, the partial sum of the first n terms of the geometric series $a_k = r^k$ corresponds to the following expression:

$$S_n = \sum_{k=0}^{n} r^k = 1 + r + r^2 + r^3 + \cdots + r^n.$$

The *series* $\sum a_k$ is defined as the limit $n \to \infty$ of the above expression. For values of r that obey $|r| < 1$, the series *converges* to the a finite value:

$$S_\infty = \lim_{k \to \infty} S_k = \sum_{k=0}^{\infty} r^k = 1 + r + r^2 + r^3 + \cdots = \frac{1}{1 - r}.$$

We'll learn more about series in Section VI.

In the remainder of this tutorial we'll use limits to evaluate derivatives, integrals, and series expressions. In each of these domains, limit expressions will help us make precise statements that describe calculus procedures with infinite small lengths and infinite number of steps.

IV. Derivatives

The *derivative* function, denoted $f'(x)$, $\frac{d}{dx}f(x)$, or $\frac{df}{dx}$, describes the *rate of change* of the function $f(x)$. For example, the constant function $f(x) = c$ has derivative $f'(x) = 0$ since the function $f(x)$ does not change at all. The derivative function describes the *slope* of the graph of the function $f(x)$. The derivative of a line $f(x) = mx + b$ is $f'(x) = m$, since the slope of this line is equal to m for all values of x. In general, the slope of a function is different at different values of x, so mathematicians invented a new notation $f'(x)$ for describing "the slope of the function f at x."

The derivative function $f'(x)$ is defined as the rate of change of the function f at x, and it is computed using the formula:

$$f'(x) \stackrel{\text{def}}{=} \lim_{\delta \to 0} \frac{f(x + \delta) - f(x)}{\delta}.$$

In words, this formula describes the general rise-over-run calculation for computing the slope of a line that connects the points $(x, f(x))$ and $(x + \delta, f(x + \delta))$, with the step-length δ becoming infinitely small.

Geometrically, the derivative function computes the slope of the graph of the function $f(x)$ for all values of x. In general, the slope of a function is different for different values of x. Figure 9 shows the slope calculation for the function $f(x) = \frac{1}{2}x^2$ for two different values of x: $x = -0.5$ and $x = 1$.
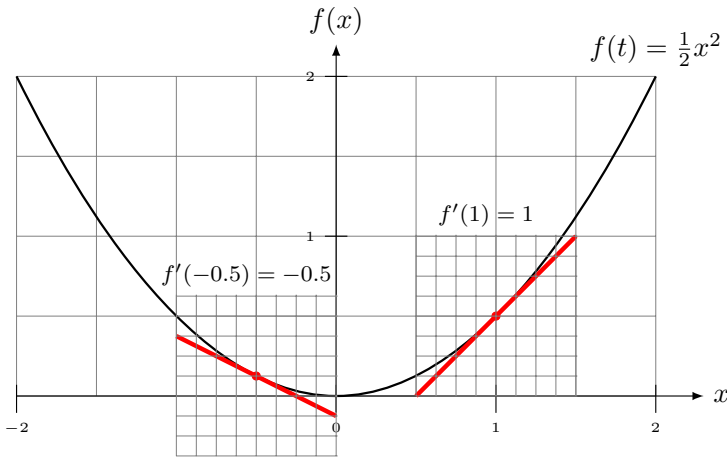

Fig. 9. The derivative of the function at $x = a$ is denoted $f'(a)$ and describes the slope function at that point.

The derivative function $f'(x)$ is a property of the function $f(x)$. Indeed, this is where the name *derivative* comes from: $f'(x)$ is not an independent function—it is *derived* from the original function $f(x)$.

The *derivative operator*, denoted $\frac{d}{dx}$ or simply D, takes as input a function $f(x)$ and produces as output the derivative function $f'(x)$, which is a function of the form $f' : \mathbb{R} \to \mathbb{R}$. For each input x_0 the derivative function tells you the slope of $f(x)$ when $x = x_0$. Applying the derivative operator to a function is also called "taking the derivative" of a function. For example, the derivative of the function $f(x) = \frac{1}{2}x^2$ is the function $f'(x) = x$. We can describe this relationship as $(\frac{1}{2}x^2)' = x$ or as $\frac{d}{dx}(\frac{1}{2}x^2) = x$. Look at Figure **??** and use the graph to prove to yourself that the slope of $f(x) = \frac{1}{2}x^2$ is described by $f'(x) = x$ everywhere on the graph.

A. Numerical derivative calculations

Here is a simple computer program for computing a numerical approximations to the derivative the function `f` at the point `x`:

```
>>> def differentiate(f, x, delta=1e-9):
        df = f(x+delta) - f(x)
        dx = delta
        return df / dx
```

The code performs the same calculation as in the definition of the derivative, but using a finite step `delta` $= 10^{-9}$ instead of the infinitely small step δ described by the limit calculation.

Consider the Python function `f` that corresponds to the math function $f = \frac{1}{2}x^2$. We can use `differentiate` to evaluate the derivative the function when $x = 1$:

```
>>> def f(x):
        return 0.5 * x**2
>>> differentiate(f, 1)
1.000000082740371
```

We obtain the approximation $f'(1) = 1.000000082740371$, which is not perfect but pretty close to the true value $f'(1) = 1$. For most practical applications, this approximation is good enough.

B. Derivative formulas

You don't need to apply the complicated derivative formula $f'(x) \stackrel{\text{def}}{=} \lim_{\delta \to 0} \frac{f(x+\delta)-f(x)}{\delta}$ every time you need to find the derivative of a function. For each function $f(x)$, it's enough to use the complicated formula once and record the formula you obtain for $f'(x)$, then you can reuse that formula whenever you need to compute $f'(x)$ in a later calculation.

The following table shows the derivative formulas for a number of commonly used functions.

$f(x)$	$-$ derivative $\to$	$f'(x)$
a	$-\frac{d}{dx} \to$	0
$\alpha f(x) + \beta g(x)$	$-\frac{d}{dx} \to$	$\alpha f'(x) + \beta g'(x)$
x	$-\frac{d}{dx} \to$	1
$mx + b$	$-\frac{d}{dx} \to$	m
x^n	$-\frac{d}{dx} \to$	nx^{n-1}
$\frac{1}{x} = x^{-1}$	$-\frac{d}{dx} \to$	$\frac{-1}{x^2} = -x^{-2}$
$\sqrt{x} = x^{\frac{1}{2}}$	$-\frac{d}{dx} \to$	$\frac{1}{2\sqrt{x}} = \frac{1}{2}x^{-\frac{1}{2}}$
e^x	$-\frac{d}{dx} \to$	e^x
$\ln(x)$	$-\frac{d}{dx} \to$	$\frac{1}{x}$
$\sin(x)$	$-\frac{d}{dx} \to$	$\cos(x)$
$\cos(x)$	$-\frac{d}{dx} \to$	$-\sin(x)$

I invite you to mentally bookmark this page so you can come back to it when derivatives come up.

C. Derivative rules

In addition to the table of derivative formulas show above, there are some important derivatives rules that allow you to find derivatives of *composite* functions.

Linearity: The derivative is a *linear* operation, which means:

$$\frac{d}{dx}[\alpha f(x) + \beta g(x)] = \alpha \frac{d}{dx} f(x) + \beta \frac{d}{dx} g(x).$$

In other words, the derivative of a linear combination of functions $\alpha f(x) + \beta g(x)$ is equal to the same linear combination of the derivatives $\alpha f'(x) + \beta g'(x)$.

Product rule: The derivative of a product of two functions is obtained as follows:

$$[f(x)g(x)]' = f'(x)g(x) + f(x)g'(x).$$

In each term, the derivative of one of the functions is multiplied by the other function.

Quotient rule: The *quotient rule* tells us how to obtain the derivative of a fraction of two functions:

$$\left[\frac{f(x)}{g(x)}\right]' = \frac{f'(x)g(x) - f(x)g'(x)}{g(x)^2}.$$

Chain rule: If you encounter a situation that includes an inner function and an outer function, like $f(g(x))$, you can obtain the derivative by a two-step process:

$$[f(g(x))]' = f'(g(x))g'(x).$$

In the first step, leave the inner function $g(x)$ alone, and focus on taking the derivative of the outer function $f(x)$. This step gives us $f'(g(x))$, the value of f' evaluated at $g(x)$. As the second step, we multiply the resulting expression by the derivative of the *inner* function $g'(x)$.

D. Higher derivatives

The second derivative of $f(x)$ is denoted $f''(x)$ or $\frac{d^2 f}{dx^2}$. It is obtained by applying the derivative operator $\frac{d}{dx}$ to $f(x)$ twice. The second derivative $f''(x)$ encodes the information about the *curvature* of $f(x)$. Positive curvature means the function opens upward, and looks like the bottom of a valley. The function $f(x) = \frac{1}{2}x^2$ shown in Figure 9 has derivative $f'(x) = x$ and second derivative $f''(x) = 1$, which means it has positive curvature. Negative curvature means the function opens downward, and looks like a mountain peak. For example, the function $g(x) = -x^2$ has negative curvature.

E. Examples

Armed with these derivative formulas and rules, you can the derivative of any function, no matter how complicated. Let's look at some examples.

Example 1: To calculate the derivative of $f(x) = e^{x^2}$, we use the chain rule: $f'(x) = e^{x^2}[x^2]' = e^{x^2}2x$.

Example 2: To find the derivative of $f(x) = \sin(x)e^{x^2}$, we use the product rule and the chain rule: $f'(x) = \cos(x)e^{x^2} + \sin(x)2xe^{x^2}$.

Example 3: The derivative of $\sin(x^2)$ requires using the chain rule: $\left[\sin(x^2)\right]' = \cos(x^2)\left[x^2\right]' = \cos(x^2)2x$.

F. Computing derivatives analytically using SymPy

The `SymPy` function `diff` computes the derivative of any expression. For example, here is how we can compute the derivative of the function $f(x) = mx + b$:

```
>>> m, x, b = sp.symbols("m x b")
>>> sp.diff(m*x + b, x)
m
```

Let's also verify the derivative formula $\frac{d}{dx}[x^n] = nx^{n-1}$:

```
>>> x, n = sp.symbols("x n")
>>> sp.diff(x**n, x)
n * x**(n - 1)
```

The exponential function $f(x) = e^x$ is special because it is equal to its derivative:

```
>>> from sympy import exp
>>> sp.diff(exp(x), x)
exp(x)
```

Let's check the derivative calculations from the above examples:

```
>>> sp.diff(sp.exp(x**2), x)
2*x*exp(x**2)
>>> sp.diff(sp.sin(x)*sp.exp(x**2), x)
2*x*exp(x**2)*sin(x) + exp(x**2)*cos(x)
>>> sp.diff(sp.sin(x**2), x)
2*x*cos(x**2)
```

G. Applications of derivatives

We use derivatives to solve problems in physics, chemistry, computing, biology, business, and many other areas of science. The derivative operator comes up whenever we study the rate of change of a quantity. We use derivatives to obtain local liner approximations to functions (tangent lines).

Tangent lines: The *tangent line* to the function $f(x)$ at $x = x_0$ is the line that passes through the point $(x_0, f(x_0))$ and has the same slope as the function at that point. The tangent line to the function $f(x)$ at the point $x = x_0$ is described by the equation

$$T_1(x) = f(x_0) + f'(x_0)(x - x_0).$$

For example, the tangent line to $f(x) = \frac{1}{2}x^2$ at $x_0 = 1$ is $T_1(x) = f(1) + f'(1)(x-1) = \frac{1}{2} + 1 \cdot (x-1) = x - \frac{1}{2}$. Look back at Figure 9 for an illustration.

The tangent line T_1 is also called a *first order approximation* to the function f, since it has the same value and the same derivative as the function f, $T_1(1) = f(1)$ and $T_1'(1) = f'(1)$. In Section VI-D, we'll learn how to build order-n approximations $T_n(x)$.

H. Optimization

Derivatives are very useful for solving optimization problems, which consist of finding the maximum or minimum value of some function $f(x)$. For example, look the graph of the function $f(x) = \frac{1}{2}x^2$ in Figure 9. The minimum of the function occurs when $x = 0$ where the slope of the function is zero $f'(0) = 0$. Note also the second derivative of is positive at that point $f''(0) > 0$, which tells us the function locally looks like bottom of a bowl.

Analytical optimization: The values of x^* where the derivative is zero are called the *critical points* of the function. Optimum values (maximum or minimum) occurs at a critical point of the function. We identify a critical point x^* that corresponds to minimum if the second derivative is positive at that point $f''(x^*) > 0$ (positive curvature). In contrast, a critical point x^* here $f''(x^*) < 0$ (negative curvature) is a maximum. This observation suggests a general procedure for finding minima and maxima:

(1) Solve the equation $f'(x) = 0$ to find the critical points x^*.
(2) For each critical point x^*, check to see if it is a maximum or a minimum by evaluating $f''(x^*)$:
- If $f''(x^*) < 0$ then x^* is a max (mountain top)
- If $f''(x^*) > 0$ then x^* is a min (bottom of a valley).

We can also perform the check in step (2) visually by looking at the graph of the function, or by evaluating the slope of the function near the critical point. If $f'(x^* - 0.1)$ is negative and $f'(x^* + 0.1)$ is positive, the point x^* is a minimum (as in Figure 9). If $f'(x^* - 0.1)$ is positive and $f'(x^* + 0.1)$ is negative, then the point x^* is a maximum. If $f'(x^* - 0.1)$ and $f'(x^* + 0.1)$ have the same sign, the value x^* is a *saddle point*, which is neither a minimum or a maximum.

Example: Let's use the two-step procedure to find the minimum and the maximum of the function $f(x) = x^3 - 2x^2 + x$. First we calculate its derivative $f'(x) = 3x^2 - 4x + 1 = 3(x-1)(x - \frac{1}{3})$. Next we find the critical points by solving the equation $f'(x) = 0$, which gives us two critical points $x_1^* = \frac{1}{3}$ and $x_2^* = 1$. The second derivative of the function is $f''(x) = 6x - 4$. For the critical value $x_1^* = \frac{1}{3}$, we find $f''(\frac{1}{3}) = -2 < 0$, which tells us x_1^* is a maximum. For $x_2^* = 1$, we find $f''(1) = 2$, which tells us x_1^* is a minimum.

Numerical optimization: Consider the shape of the function near a minimum value. The function is decreasing just before it reaches its minimum, and the function increases just after its minimum. This means we can start at any point $x = x_0$ and take "downhill" steps following the descending direction of the function, we'll end up at the minimum value. This simple procedure that repeatedly takes steps in the direction where the function is decreasing turns out to be a very powerful tool that can find the minimum of any function. This procedure is called the *gradient descent algorithm*, where the name *gradient* refers to the derivative operation for multivariable functions.

```
>>> TODO derivative_descent

>>> def f(x):
        return x**2 / 2
>>> TODO use derivative_descent start at x=10
```

Numerical optimization using SciPy: Here is a quick code example that shows how to use the function `minimize` defined in the module `scipy.optimize` to find the minimum value of the function $f(x) = (x - 5)^2$.

```
>>> from scipy.optimize import minimize
>>> res = minimize(f, x0=0)
>>> res["x"][0]    # = argmin f(x)
4.99999997455944
```

The `minimize` function takes two arguments: the function to minimize, and a initial value x_0 where to start the minimization process.

V. Integrals

Integration is process of computing the "total" of some quantity that varies over time. The integral sign $\int$ used to denote integrals is an elongated letter S, as a reminder that we're summing together some quantity.

There are actually two different tasks that are both called integration. The *definite integral* of $f(x)$ between $x = a$ and $x = b$ is denoted $\int_{x=a}^{x=b} f(x)\,dx = A(a,b)$ and correspond to the computation of the area under graph of $f(x)$ between a and b. The definite integral is a number $A(a,b) \in \mathbb{R}$. In contrast, the *indefinite integral* of $f(x)$ is denoted $\int_{x=0}^{x=b} f(x)\,dx = F_0(b)$ is a *function* that describes the are-under-the-graph-of-$f(x)$ with a variable upper limit of integraiton. The two integration operations are related. The area under the curve $A(a,b)$ can be computed as the *change* integral function: $A(a,b) = F_0(b) - F_0(a)$. Both integration tasks are important, and we'll discuss each of them in turn.

A. Integrals as area calculations

Figure 10 illustrates the calculation of the *area* enclosed between the graph of $f(x)$ and the x-axis between the vertical lines at $x = a$ and $x = b$. In calculus, we refer to this area calculation as definite integral and denote it:

$$A_f(a,b) = \int_{x=a}^{x=b} f(x)\,dx.$$

The numbers a and b are called the *limits of integration*. We often use the notation $\int_a^b f(x)\,dx$ as shorthand for $\int_{x=a}^{x=b} f(x)\,dx$. We read this expression as "the integral of $f(x)$ between a and b."
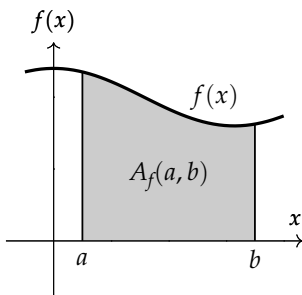

Fig. 10. The integral of the function $f(x)$ between $x = a$ and $x = b$ corresponds to the shaded area $A_f(a,b)$.

If this is the first time you're seeing the notation $\int_a^b f(x)\,dx$, you're probably freaking out, but bear with me for two more pages, and you'll see this intimidating-looking math notation is nothing to worry about! This's just a fancy way to denote a particular calculation that involves some function $f(x)$. You can think of $\int_a^b$ `<f>` dx as a "template" that you can fill in by replacing `<f>` with any function $f(x)$ to denote the area-under-the-graph-of-$f(x)$ calculation, which is also denoted $A_f(a,b)$.

Let's look at some examples.

Example 1: integral of a constant function: Consider the constant function $f(x) = 3$. No matter what the input x is, the output is always 3. We can easily find the area under the graph of this function between any two points, since the region under the graph has a rectangular shape. See Figure 11 for an illustration.

The area under $f(x)$ between $x = 0$ and $x = 5$ corresponds to the following integral calculation:

$$A_f(0,5) = \int_0^5 f(x)\,dx = 3 \cdot 5 = 15.$$

The area under the graph of $f(x)$ is a rectangle with height 3 and width 5, so its area is $3 \cdot 5 = 15$.

Fig. 11. The area of a rectangle of height 3 and width 5 is equal to 15.

Example 2: integral of a linear function: Consider now the area under the graph of the line $g(x) = x$ between $x = 0$ and $x = 5$, as shown in Figure 12. This area is described by the following integral calculation:

$$A_g(0,5) = \int_0^5 g(x)\,dx = \tfrac{1}{2}5 \cdot 5 = \tfrac{1}{2}5^2 = \frac{25}{2} = 12.5.$$

The region under the graph of $g(x)$ has a triangular shape, so we can compute its area using the formula for the area of a triangle, which is "base times height divided by 2."

Fig. 12. The area of a triangle with base 5 and height 5 is $\frac{1}{2}5^2 = \frac{25}{2} = 12.5$.

I hope these two examples are starting to convince you that the scary-looking integral sign is not that complicated after all. It's just a fancy way to describe "area under the graph of a function" calculations.

Example 3: integral of a polynomial: Consider now the function $h(x) = x^3 - 5x^2 2 + x + 10$. Suppose we want to know the area under the graph of $h(x)$ between $x = -1$ and $x = 4$, as illustrated in Figure XX. We need to calculate the following integral:

$$A_h(0,5) \stackrel{\text{def}}{=} \int_{-1}^4 h(x)\,dx = ?.$$

Looking at the graph of $h(x)$, we see it doesn't have a recognizable geometric shape with a known area formula. How could we compute the area then?

One way to approximate the area under $h(x)$ is to split it up into bunch of thin vertical rectangular strips of some fixed width, which we'll denote Δx. The height of each rectangular strip will vary depending on $f(x)$. Look ahead to Figure 13 to see where we're going with this. We can calculate the area of the kth strip using the "base times height" formula for the area of a rectangle $\Delta x f(x_k)$, where x_k is the left endpoint of the kth strip. For example, we splitting up the area $A_h(0,5)$ into $n = 25$ strips, calculating the area of the each strip, and summing them together produces the approximation $A_h(0,5) \approx \dots$. If we split the area $A_h(0,5)$ into $n = 50$ strips, we obtain the more accurate approximation $A_h(0,5) \approx \dots$. The integral $\int_{-1}^{4} h(x)\,dx$ is defined as the limit where the number of vertical strips n goes to infinity.

Relax, we won't be doing the $n = 25$ and $n = 50$ calculation by hand, let alone the exact version as n goes to infinity! Instead, we'll write a computer program and that performs the integration procedure for us. Computer are really proficient at this stuff, and this is not a coincidence—computers were originally invented precisely for computer integrals.

B. Computing integrals numerically

Computing integral of $f(x)$ "numerically" means we're splitting the region of integration into many (think thousands or millions) of strips, computing the areas of these strips, then adding up the areas to obtain the total area under the graph of $f(x)$. The approximation to the area under $f(x)$ between $x = a$ and $x = b$ using n rectangular strips corresponds to the following formula:

$$A_f(a,b) \approx \sum_{k=1}^{n} f(a + k\Delta x)\,\Delta x,$$

where $\Delta x = \frac{b-a}{n}$ is the width of the rectangular strips. The right endpoint of the kth is located at $x_k = a + k\Delta x$, so the height of the rectangular strips $f(x_k)$ varies as k goes from between $k = 1$ (first strip) and $k = n$ (last strip).

Let's check that the formula $f(a + k\Delta x)\Delta x$ correctly describes the area of the kth rectangular strip. Since we're using choosing the height of the rectangles according to their right endpoints, the area of the first rectangle is $f(x_1)\Delta x = f(a+\Delta x)\Delta x$, which is the height of the function f at $a + \Delta x$ times the width Δx. The second rectangle has height $f(a + 2\Delta x)$, the third rectangle will have height $f(a + 3\Delta x)$, and so on until the last one which has height $f(a + n\Delta x) = f(a + n\frac{b-a}{n}) = f(b)$.

Instead of manually commuting all the n area calculations in the summaiton, we can write Python code that takes an arbitrary function f as input and performs the n-rectangle area approximation calculation for us:

```
>>> def integrate(f, a, b, n=10000):
        """
        Computes the area under the graph of `f`
        between `x=a` and `x=b` using `n` rectangles.
        """
        dx = (b - a) / n
        xs = [a + k*dx for k in range(1,n+1)]
        fxs = [f(x) for x in xs]
        area = sum([fx*dx for fx in fxs])
        return area
```

The code implements word-for-word the summation that we defined in the above equation. We first compute the width

of the rectangles $\mathtt{dx} = \Delta x$. We then create the list $\mathtt{xs}$ that contains the x-coordinates of the right endpoints of the $\mathtt{xs} = [a + \Delta x, a + 2k\Delta x, a + 3k\Delta x, \dots, b]$, and evaluate the function $\mathtt{f}$ at these x-values to obtain $\mathtt{fxs} = [f(a + \Delta x), f(a + 2k\Delta x), f(a + 3k\Delta x), \dots, f(b)]$. We obtain the total area by multiplying each $\mathtt{fx}$ in $\mathtt{fxs}$ by the width $\mathtt{dx}$ and summing of them together.

Example 3 continued: Let's use the $\mathtt{integrate}$ procedure to compute the integral of the function $h(x) = x^3 - 5x^2 2 + x + 10$ that we saw in Example 3. First we define a Python function that implements h:

```
>>> def h(x):
        return x**3 - 5*x**2 + x + 10
```

We can now calculate the $n = 25$ approximation to the area under the graph of $h(x)$ between $x = -1$ and $x = 4$ as follows:

```
>>> integrate(h, -1, 4, n=25)
12.399999999999997
```

Then you can compute S_{25} by calling $\mathtt{integrate(f, -1, 4, n=25)}$, which returns $S_{25} = 12.4$.

Calling $\mathtt{integrate(f, -1, 4, 50)}$ you'll obtain $S_{50} = 12.6625$.

```
>>> integrate(h, -1, 4, n=50)
12.6625
```


(a) $n = 25$ (b) $n = 50$

Fig. 13. An approximation to the area under the graph of the function $f(x) = x^3 - 5x^2 + x + 10$ using $n = 25$ and $n = 50$ rectangles.

The approximations will get better and better if we increase the number of rectangles n. For $n = 100$, the sum of the rectangles' areas is $S_{100} = 12.7906$, for $n = 1000$ the approximation gives us $S_{1000} = 12.9041562$, which is accurate to the first decimal.

```
>>> integrate(h, -1, 4, n=1000)
12.90415625
>>> integrate(h, -1, 4, n=100000)
12.916654166656249
```

The exact value of the area $A_h(-1,4)$ is $\frac{155}{12} = 12.91\overline{6}$. We'll see how to compute this in the next section, when we learn about symbolic integration, which is a math "shortcut" that allows us compute exact integral for certain functions. For now, we content ourselves with numerical approximations, which are pretty good already. The approximation with $n = 100K$ is accurate to four decimals!

Examples 1 and 2 revisited: We can also use $\mathtt{integrate}$ to compute the integral of a constant function $f(x) = 3$ and the linear function $g(x) = x$ that computed geometrically earlier.

```
>>> def f(x):
        return 3
>>> integrate(f, a=0, b=5, n=100000)
```

```
15.000000000000002
>>> def g(x):
        return x
>>> integrate(g, a=0, b=5, n=100000)
12.500125
```

The numerical approximations we obtain are very close to the exact answers $\int_0^5 f(x)\,dx = 3 \cdot 5 = 15$ and $\int_0^5 g(x)\,dx = \frac{1}{2}5 \cdot 5 = 12.5$.

C. Formal definition of integral

In the limit as the number of rectangles n approaches ∞, the approximation to the area under the curve becomes *arbitrarily close* to the true area. The notion of applying the a rectangular-strip approximation to the area of a function, where the number of rectangles grows to infinity is known as the *Riemann sum* and is the basis for the definitions of the integral:

The integral between $x = a$ and $x = b$ is *defined* as the limit as n goes to infinity:

$$\int_a^b f(x)\,dx \equiv \lim_{n \to \infty} \sum_{k=1}^n f(a + k\Delta x)\Delta x \equiv A(a, b).$$

This is sometimes called the *Riemann sum* formula.

D. Computing integrals numerically using SciPy

The Python function `quad` in the module `scipy.integrate` allows us to compute the integral of any function. The name `quad` is short for "quadrature" which is the historical math term used for find-the-area procedures. Let's start by importing the `quad` function.

To compute the integral $\int_0^5 f(x)dx$ we call the function `quad` with inputs `f` as the first argument, and the limits of integration $a = 0$ and $b = 5$ as the second and third arguments.

```
>>> from scipy.integrate import quad
>>> quad(f, 0, 5)
(15.0, 1.1102230246251565e-13)
```

The function `quad` returns a tuple (a pair of numbers) as output: (A, ϵ). The first number in the tuple is the value of the area calculation. The second number ϵ tells us the accuracy of the procedure used to calculate the area. In the above calculation, the output tells us the integral $\int_0^5 f(x)dx$ is equal to 15.0 up to a precision on the order of 10^{-13}.

Since we're usually only interested in the value of the area A and not the precision ϵ, we often select the first number in the output of `quad`. This is why you'll often see the expression `quad(...)[0]` in code examples.

```
>>> quad(f, 0, 5)[0]   # extract A
15.0
```

As a second example, let's calculate the area under the graph of the function $g(x) = x$ between $x = 0$ and $x = 5$.

```
>>> def g(x):
        return x
>>> quad(g, 0, 5)[0]
12.5
```

The answer we obtained matches the results of the general formula we obtained above, $A_g(0, 5) = \frac{1}{2}b^2$, when the upper limit of integration is $b = 5$.

The main takeaway message is that the `quad` function is your friend whenever you need to compute integrals. All the scary-looking math equations that contain the $\int$ symbol can be computed using one or two lines of Python code.

TODO h

```
>>> quad(h, 0, 5)[0]   # extract A
10.416666666666675
```


E. Integrals as functions

The *integral function* $F_0(b)$ corresponds to the area calculation with a variable upper limit of integration $A_f(0, b)$. The variable b, which serves as the input for the integral function F_0, corresponds to the upper limit of integration in the following calculation:

$$F_0(b) \quad \overset{\text{def}}{=} \quad A_f(0, b) = \int_{x=0}^{x=b} f(x)\,dx\,.$$

There are two variables and one constant in this formula. The input variable b describes the upper limit of integration. The *integration variable* x performs a sweep from $x = 0$ until $x = b$. The constant 0 describes the lower limit of integration. As a matter of convention, we'll always denote the integral function using the capital letter of the same letter as the original function.

Note that choosing $x = 0$ for the starting point of the integral function was an arbitrary choice, and we obtain another integral function if we use $x = a$ as the starting point, $F_a(b) = \int_a^b f(x)\,dx$. Two integral functions differ only by a constant term. For example, $F_0(b) = F_a(b) + C$, where $C = \int_{x=0}^{x=a} f(x)\,dx$.

The integral function $F(b)$ contains the "precomputed" information about the area under the graph of $f(x)$. The area under $f(x)$ between $x = a$ and $x = b$ can be obtained by calculating the *change* in the integral function as follows:

$$A_f(a, b) = \int_a^b f(x)\,dx = F(b) - F(a).$$

Intuitively, this formula computes the area $A_f(a, b)$ as the difference between the areas of two regions: the area until $x = b$ minus the area until $x = a$, as illustrated in Figure 14.

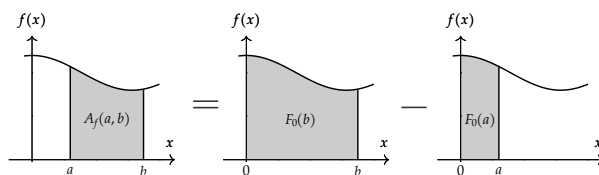Fig. 14. The area under $f(x)$ between $x = a$ and $x = b$ is computed using the formula $A_f(a, b) = F_0(b) - F_0(a)$, which describes the change in the output of $F_0(x)$ between $x = a$ and $x = b$.

TODO: warn there is no general F for any f only for certain special cases have exact symbolic formula for all other cases we're forced to do the split-into-vertical-strips — i.e. there is no analytical shortcut.

Example 1 revisited: We can easily find the integral function for the constant function $f(x) = 3$, because the region under the curve is rectangular. Choosing $x = 0$ as the starting point, we obtain the integral function $F_0(b)$ that corresponds to the area under $f(x)$ between $x = 0$ and $x = b$ as follows:

$$F_0(b) = A_f(0, b) = \int_0^b f(x)\, dx = 3b.$$

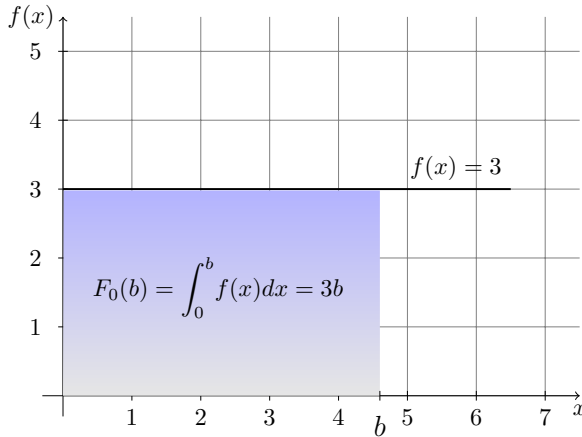The area is equal to the rectangle's height times its width, as illustrated in Figure 15.

Fig. 15. The area of a rectangle of height 3 and width b is equal to $3b$.

Knowing the function $F_0(b)$ allows us to compute the area under the graph of $f(x)$ between any two points $x = a$ and $x = b$ using the formula $A_f(a, b) = F_0(b) - F_0(a) = 3(b - a)$.

Example 2 revisited: Consider now the area under the graph of the line $g(x) = x$, starting from $x = 0$. Since the region under the curve is triangular, we can compute its area using the formula for the area of a triangle: base times height divided by two.

The general formula for the area under $g(x)$ from $x = 0$ until $x = b$ is described by the following integral calculation:

$$G_0(b) = A_g(0, b) = \int_0^b g(x)\, dx = \frac{1}{2}(b \times b) = \frac{1}{2}b^2.$$
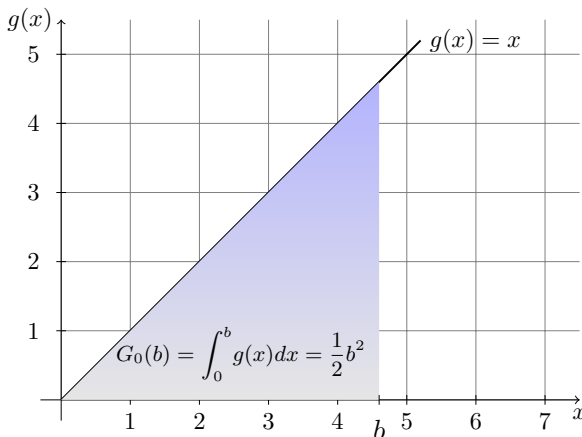

Fig. 16. The area of a triangle with base b and height b is equal to $\frac{1}{2}b^2$.

Knowing the function $G_0(b)$ allows us to compute the area under the graph of $g(x)$ between $x = a$ and $x = b$ as the difference $A_g(a, b) = G_0(b) - G_0(a) = \frac{1}{2}b^2 - \frac{1}{2}a^2$.

Example 3 revisited: We

$$H_{-1}(b) = A_h(-1, b) = \int_{-1}^b h(x)\, dx = ?$$

one of the special cases where there IS a shotctu

We were able to compute the above integrals thanks to the simple geometries of the areas under the graphs.

F. Computing integrals using SymPy

We can also use Python to do *symbolic* integration using variables (symbols) instead of numbers. Symbolic integration allows us to obtain exact formulas for integrals that are valid for *any* limits of integration $x = a$ and $x = b$. The Python module `sympy` provides the functionality for doing symbolic math calculations similar to the calculation you could do using pen and paper.

We can literally do the same calculations as in previous examples and obtain the same answers.

The following code block imports the SymPy function `symbols`, which is used to define new symbolic variables, and the function `integrate` that we'll use for computing integrals.

```
>>> from sympy import symbols, integrate
```

Next we define four symbols x, a, b, and c, which we'll use to denote mathematical variables and constants in the following code examples.

```
>>> x, a, b, c = symbols('x a b c')
```

Example 1S: constant function: Consider the constant function $f(x) = c$. The symbolic expression that represents the value of this function is simply the constant c, which we can define as follows:

```
>>> fx = c
>>> fx
c
```

The variable `fx` is defined as the constant `c`, one of the SymPy symbols we defined earlier, which we assume corresponds to some unspecified constant value.

To compute the integral $\int_a^b f(x)dx$, we call the SymPy function `integrate`, passing in the expression we want to integrate as the first argument. The second argument is a triple (x, a, b), which specifies the variable of integration x, the lower limit of integration a, and the upper limit of integration b.

```
>>> integrate(fx, (x,a,b))   # = A_f(a,b)
c*(b-a)
```

Since a, b, and c are arbitrary constants, the expression we obtain for $A_f(a, b) = \int_a^b f(x)dx$ is a general purpose formula that works for all functions $f(x) = c$ and all possible integration intervals $[a, b]$. Geometrically speaking, this is just the height-times-width formula for the area of a rectangle.

To compute the specific integral between $a = 0$ and $b = 5$ under the graph of $f(x) = 3$, we can use the method `subs` (short for substitute) on the SymPy expression we obtained as a result of the integration. The `subs` method expects as inputs a Python dictionary whose keys are symbols, and whose values represent the numbers we want to "plug" into the expression. In our case, we want to make the substitutions $c = 3$, $a = 0$, and $b = 5$.

```
>>> integrate(fx, (x,a,b)).subs({c:3, a:0, b:5})
15
```

This result matches the value we obtained using the intuitive geometrical calculation (see Figure 11) and the value we obtained using numerical integration, `quad(f,0,5) = 15`.

We can also use SymPy to compute the integral function $F_0(b)$, which is defined as $F_0(b) \stackrel{\text{def}}{=} \int_0^b f(x)dx$, for the function $f(x) =$ `fx`.

```
>>> integrate(fx, (x,0,b))    # = F_0(b)
b*c
```

Recall that the integral function F_0 is simply the area-under-the-graph calculation with a variable upper limit of integration b. See Figure 15 for an illustration of the integral function $F_0(b)$.

Example 2S: linear function: Let's now compute some integrals of the function $g(x) = x$. First we'll define a SymPy expression that corresponds to the function.

```
>>> gx = 1*x
>>> gx
x
```

We can now compute the integral $\int_a^b g(x)dx$ by calling the function `integrate` with arguments `gx`, followed by the triple specifying the variable of the integration and the limits of integration.

```
>>> integrate(gx, (x,a,b))    # = A_g(a,b)
b**2/2 - a**2/2
```

To obtain the numerical value for the integral $\int_0^5 g(x)dx$, we call the method `subs` on the result of the integration.

```
>>> integrate(gx, (x,a,b)).subs({a:0, b:5})
25/2
```

SymPy computed the exact answer for us as a fraction $\frac{25}{2}$, but we sometimes want to force the answer to be computed as a floating-point number (a Python `float`), which we can do by calling the `.evalf()` method on the SymPy expression.

```
>>> integrate(gx, (x,a,b)).subs({a:0, b:5}).evalf()
12.5
```

This result matches the value we obtained earlier using numerical integration, `quad(g,0,5) = 12.5`.

If we use the symbol `b` for the upper limit of integration, we can obtain an expression for the integral function $G_0(b) \stackrel{\text{def}}{=} \int_0^b g(x)dx$.

```
>>> integrate(gx, (x,0,b))    # = G_0(b)
b**2 / 2
```

Note the expression for $G_0(b)$ we obtain from SymPy is identical to the formula we obtained earlier using a geometrical calculation (the area of a triangle with base b and height b). See Figure 16.

```
>>> hx = x**3 - 5*x**2 + x + 10
>>> hx
x**3 - 5*x**2 + x + 10
```

TODO code

In `SymPy` we use `integrate(f, x)` to obtain the integral function $F(x)$ of any function $f(x)$: $F(x) = \int_0^x f(u)\,du$.

```
>>> integrate(x**3, x)
x**4/4
>>> integrate(sin(x), x)
-cos(x)
>>> integrate(ln(x), x)
x*log(x) - x
```

This is known as an *indefinite integral* since the limits of integration are not defined.

In contrast, a *definite integral* computes the area under $f(x)$ between $x = a$ and $x = b$. Use `integrate(f, (x,a,b))` to compute the definite integrals of the form $A(a,b) = \int_a^b f(x)\,dx$:

```
>>> integrate(x**3, (x,0,1))
1/4                   # the area under x^3 from x=0 to x=1
```

We can obtain the same area by first calculating the indefinite integral $F(c) = \int_0^c f(x)\,dx$, then using $A(a,b) = F(x)\big|_a^b = F(b) - F(a)$:

```
>>> F = integrate(x**3, x)
>>> F.subs({x:1}) - F.subs({x:0})
1/4
```

Unfortunately, it's not always possible to use symbolic manipulations to find integrals. We can only use `sympy.integrate` for certain simple examples where it is possible to obtain exact expressions for integral functions. For most practical calculations in probability and statistics, we'll need to rely on the `scipy.integrate` function `quad(f,a,b)`, which computes the integral $\int_a^b f(x)dx$ for *any* function $f(x)$ expressed as a Python function `f`.

G. Properties of integrals

We'll now state some properties of integrals that follow from their interpretation as area calculations.

- **Additivity.** The integral from a to b plus the integral from b to c is equal to the integral from a to c:

$$\int_a^b f(x)\,dx + \int_b^c f(x)\,dx = \int_a^c f(x)\,dx.$$

- **Linearity.** Integration is a *linear* operation: it preserves linear combinations. The integral of the linear combination of two functions $\alpha f(x) + \beta g(x)$, is equal to the same linear combination of the integrals of the two functions:

$$\int [\alpha f(x) + \beta g(x)]\, dx = \alpha \int f(x)\, dx \ + \ \beta \int g(x)\, dx,$$

where α and β are two arbitrary constants.
- **Integral at a single point.** Integrals over intervals with zero length have zero value for any function $f(x)$:

$$\int_a^a f(x)\, dx = 0.$$

Thinking geometrically, this integral defines a region with height $f(x)$ and width 0, so it corresponds to zero area.

H. Fundamental theorem of calculus

The fundamental theorem of calculus (FTC) is a deep insight about the inverse relation that exists between the operations of integration $\int \cdot dx$ and differentiation $\frac{d}{dx}[\cdot]$.

A priori, there is no reason to suspect the integral function would be related to the derivative operation. The integral corresponds to the computation of an area, whereas the derivative operation computes the slope of a function. The fundamental theorem of calculus describes the relationship between derivatives and integrals.

Theorem (fundamental theorem of calculus): Let $f(x)$ be a continuous function on the interval $[a, b]$, and let $\alpha \in \mathbb{R}$ be a constant. Define the function $A_\alpha(x)$ as follows:

$$A_\alpha(x) \equiv A(\alpha, x) = \int_\alpha^x f(u)\, du.$$

Then, the derivative of $A_\alpha(x)$ with respect to x is equal to $f(x)$:

$$\frac{d}{dx}\big[A_\alpha(x)\big] = f(x),$$

for any $x \in (a, b)$.

Differential calculus and integral calculus are two sides of the same coin. If you understand why the theorem is true, you will understand something very deep about calculus. Differentiation is the inverse operation of integration.

In order to understand the inverse relationship between integration and differentiation, we can draw an analogy with the inverse relationship between a function f and its inverse function f^{-1}, which *undoes* the effects of f. See Figure 4 on page 5. Given some initial value x, if we apply the function f to obtain the number $f(x)$, and apply the inverse function f^{-1} on the number $f(x)$, then the result will be the initial value x we started from:

$$f^{-1}(f(x)) = x.$$

Similarly, the derivative operator is the "inverse operator" of the integral operator. If you perform the integral operation followed

by the derivative operation on some function, you'll get back to original function:

$$\frac{d}{dx} \int_c^x f(u)\, du = f(x).$$

We can use SymPy to verify the fundamental theorem of calculus. First we construct a function `f` and compute its integral function `F` using `integrate`:

```
>>> from sympy import diff, integrate, log, exp, sin
>>> f =  log(x) + exp(x) + sin(x)
>>> F = integrate(f)
>>> F
x*log(x) - x + exp(x) - cos(x)
```

If we now take the derivative of the function `F`, we get back the original function `f`.

```
>>> diff(F)
log(x) + exp(x) + sin(x)
>>> diff(integrate(f)) == f   # FTC part 1
True
```

The integral is the "inverse operation" of the derivative. If you perform the integral operation followed by the derivative operation on some function, you'll obtain the same function:

$$\left(\frac{d}{dx} \circ \int dx \right) f(x) = \frac{d}{dx} \int_c^x f(u)\, du = f(x).$$

```
>>> f = x**2
>>> F = integrate(f, x)
>>> F
x**3/3            # + C
>>> diff(F, x)
x**2
```

Alternately, if you compute the derivative of a function followed by the integral, you will obtain the original function $f(x)$ (up to a constant):

$$\left(\int dx \circ \frac{d}{dx} \right) f(x) = \int_c^x f'(u)\, du = f(x) + C.$$

```
>>> f = x**2
>>> df = diff(f, x)
>>> df
2*x
>>> integrate(df, x)
x**2     # + C
```

The inverse relationship also holds for the opposite order of application: if we take the derivative of some function, then compute the integral of the derivative, then we arrive back at the original function (up to an additive constant factor).

$$\int_c^x f'(u)\, du = f(x) + C.$$

```
>>> integrate(diff(f)) == f   # FTC part 2
True
```

The fundamental theorem of calculus gives us an alternative way for computing integrals. You can find integral functions using a table of derivative formulas (see page **??**) and some "reverse engineering" thinking. To find an integral function of the function $f(x)$, we can look for a function function $F(x)$ such that $F'(x) = f(x)$.

Example: Suppose you're given a function $f(x)$ and asked to find its integral function $F(x) = \int f(x)\, dx$. This fundamental theorem of calculus tells us this problem is equivalent to finding a function $F(x)$ whose derivative is $f(x)$: $F'(x) = f(x)$. For example, suppose you want to find the indefinite integral $\int x^2\, dx$. We can rephrase this problem as the search for some function $F(x)$ such that $F'(x) = x^2$. Remembering the derivative formulas we saw above, you guess that $F(x)$ must contain an x^3 term. Taking the derivative of a cubic term results in a quadratic term. Therefore, the function you are looking for has the form $F(x) = cx^3$, for some constant c. Pick the constant c that makes this equation true: $F'(x) = 3cx^2 = x^2$. Solving $3c = 1$, we find $c = \frac{1}{3}$ and so the integral function is $F(x) = \int x^2\, dx = \frac{1}{3}x^3 + C$. In other words, the area under the graph of $f(x) = x^2$ is described by the family of functions $F(x) = \frac{1}{3}x^3 + C$.

LEAD OUT: what do we do when there is no simple formula?

I. Techniques of integration

There are a bunch of tricks that extend the reach of analytical integration methods (anti-differentiation) to more complicated functions. There many such tricks an we don't have room to discuss all of them here, but I'll show you the two most important ones.

Substitution trick: Suppose the function we want to integrate has the structure $f(u(x))u'(x)$, which consists of inner function wrapped in an outer function multiplied by the derivative of the inner function. We can use the *substitution trick* to rewrite this integral in terms of the function $f(u)$ using u as the variable of integration:

$$\int_{x \in \mathcal{X}} f(u(x))\, u'(x)dx = \int_{u \in \mathcal{U}} f(u)\, du.$$

The substitution trick is "change of variable" operation from the variable x to the variable u, similar to a search-and-replace operation when editing text. Because we're doing the substitution "inside" an integral operation, we must also change the region of integration ($\mathcal{X}$ to $\mathcal{U}$) and change of the "step" parameter (dx to du).

Follow these three steps to apply the substitution trick:
1) Replace dx with $\frac{1}{u'(x)}du$.
2) Replace all occurrences of $u(x)$ with u.
3) Replace the x limits of integration with u limits of integration.

For example, let's compute the integral $\int_a^b \frac{1}{x-\sqrt{x}}\, dx$ by applying the substitution $u = \sqrt{x}$, which implies $u'(x) = \frac{1}{2\sqrt{x}}$.

Performing the three steps of the substitution trick gives

$$\int_{x=a}^{x=b} \frac{1}{x - \sqrt{x}}\, dx = \int_{x=a}^{x=b} \frac{1}{x - \sqrt{x}} \frac{1}{2\sqrt{x}}du$$

$$= \int_{x=a}^{x=b} \frac{1}{u^2 - u}\, 2u\, du$$

$$= \int_{u(a)}^{u(b)} \frac{1}{u^2 - u}\, 2u\, du = \int_{u(a)}^{u(b)} \frac{2u}{u^2 - u}\, du$$

$$= \int_{u(a)}^{u(b)} \frac{2}{u - 1}\, du = 2\ln(u-1)\Big|_{u(a)}^{u(b)}$$

$$= 2\ln(\sqrt{x} - 1)\Big|_{x=a}^{x=b} = 2\ln(\sqrt{b} - 1) - 2\ln(\sqrt{a} - 1).$$

In the fourth line, we recognized the general form of the function inside the integral, $f(u) = \frac{2}{u-1}$, to be similar to the function $f(u) = \frac{1}{u}$ whose integral function is $\ln(u)$. Accounting for the -1 horizontal shift and the factor of 2 in the numerator, we obtain the answer $2\ln(u-1)$. In the last step, we changed back from u-variables to x-variables to compute the final answer.

The substitution trick for integrals comes from the chain rule for derivatives $[f(u(x))]' = f'(u(x))u'(x)$. Unlike the chain rule which you can apply to *all* functions of the form $f(u(x))$, the substitution rule only works when you're computing integrals where the function you're integrating has the special structure $f'(u(x))u'(x)$.

Integration by parts: Integration by parts is useful whenever the function we're integrating has the special structure $f(x)g'(x)$.

$$\int f(x)\, g'(x)\, dx = f(x)g(x) - \int f'(x)g(x)\, dx.$$

It is easier to remember the integration by parts formula in its shorthand notation, $\int u\, dv = uv - \int v\, du$. You can think of integration by parts as a form of "double substitution," where you simultaneously replace u and dv. For definite integrals, the integration by parts rule must account for evaluation at the function's limits:

$$\int_a^b u\, dv = (uv)\Big|_a^b - \int_a^b v\, du.$$

Let's see how we can calculate $\int_0^5 xe^x\, dx$ using the integration by parts procedure. We apply the substitutions $u = x$ and $dv = e^x dx$, which means $du = dx$ and $v = e^x$. Applying the formula for integration by parts gives us

$$\int_0^5 x\, e^x dx = (xe^x)\Big|_0^5 - \int_0^5 e^x\, dx$$

$$= (xe^x)\Big|_0^5 - e^x\Big|_0^5$$

$$= \left[5e^5 - 0e^0\right] - \left[e^5 - e^0\right]$$

$$= 5e^5 - e^5 + 1$$

$$= 4e^5 + 1.$$

Other techniques: Special tricks for trigonometric functions, square roots etc.

J. Applications of integration

Intuitively, we can use integration whenever we want to compute the "total" of some quantity that changes over time.

Kinematics: TODO

Solving differential equations: We use integrals to solve differential equations. If we have to solve for $f(x)$ in the differential equation $\frac{d}{dx}f(x) = g(x)$, we can take the integral on both sides of the equation to obtain the answer $f(x) = \int g(x)\, dx + C$.

Probability and expectation calculations: One of the key applications of integration to computing probabilities for continuous random variables. A continuous random variable X is described by its probability density function f_X and the probability of the event $\{a \leq X \leq b\}$ is defined as the following integral:

$$\Pr(\{a \leq X \leq b\}) \equiv \int_a^b f_X(x)\, dx.$$

The probability density f_X varies for different values of x, so if we want to compute the total probability of X falling between $x = a$ and $x = b$, we must compute the integral of f_X between $x = a$ and $x = b$.

We also use integration to compute *expectations* for quantities that depend on continuous random variables. The expected value of the quantity $G = g(X)$ under the randomness of a

continuous random variable X is defined as the following integral calculation:

$$\mathbb{E}_X[G] \equiv \mathbb{E}_X[g(X)] \equiv \int_{x \in \mathcal{X}} g(x) f_X(x) \, dx.$$

The expected value is computed by "weighing" each value of $g(x)$ by the corresponding probability density for the event $\{X = x\}$, summed over all possible values for the random variable X.

The mean $\mu = \mathbb{E}_X[X]$ and the variance $\sigma^2 = \mathbb{E}_X[(X - \mu)^2]$ are two central concepts in probability theory and statistics that are computed as expectation integrals. Every time we use the $\mathbb{E}_X$ notation in Section **??**, there will be some integral calculation going on behind the scenes, so if you want know what's going on you need to know a thing or two about integrals.

VI. Sequences and series

Sequences are functions that take whole numbers as inputs. Instead of continuous inputs $x \in \mathbb{R}$, sequences take natural numbers $k \in \mathbb{N}$ as inputs. We denote sequences as a_k instead of the usual function notation $a(k)$.

A sequence is a function of the form $a : \mathbb{N} \to \mathbb{R}$. The sequence's input variable is usually denoted k or n, and it corresponds to the *index* or number in the sequence. We describe sequences either by specifying the formula a_k for the kth term in the sequence or by listing all the values of the sequence:

$$ a_k, k \in \mathbb{N} \quad \Leftrightarrow \quad (a_0, a_1, a_2, a_3, a_4, \dots). $$

Note the new notation for the input variable as a subscript. This is the standard notation for describing sequences, and is used instead of the standard function notation $a(k)$.

Examples

We're often interested in computing the sum of all the values in this given a sequence a_k. To describe the sum of 3rd, 4th, and 5th elements of the sequence a_k, we turn to summation notation:

$$ a_3 + a_4 + a_5 \equiv \sum_{3 \le k \le 5} a_k \equiv \sum_{k=3}^{5} a_k. $$

The capital Greek letter *sigma* stands in for the word *sum*, and the range of index values included in this sum is denoted below and above the summation sign.

The partial sum of the sequence values a_k ranging from $k=0$ until $k=n$ is denoted as

$$ S_n = \sum_{k=0}^{n} a_k = a_0 + a_1 + a_2 + \cdots + a_{n-1} + a_n. $$

In calculus, the notion of a *series* describes the sum of *all* the values in the sequence a_k:

$$ \sum a_k \equiv S_\infty = \lim_{n \to \infty} S_n = \sum_{k=0}^{\infty} a_k = a_0 + a_1 + a_2 + a_3 + a_4 + \cdots. $$

Note if the sequence a_k continues indefinitely, computing the sum requires an infinite number of addition operations.

A. Exact sums

Formulas exist for calculating the sum of certain series, even series with infinite number of terms.

The formulas for the sum of the first n positive integers is

$$ \sum_{k=1}^{n} k = \frac{n(n+1)}{2}. $$

The the sum of the squares of the first n positive integers is

$$ \sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}. $$

There is another nice series for powers of 2:

$$ \sum_{k=0}^{n} 2^k = 1 + 2 + 4 + 8 + \cdots + 2^n = 2^{n+1} - 1. $$

The Binomial series

$$ \sum_{k=0}^{n} \binom{n}{k} a^{n-k} b^k = (a+b)^n $$

special case when one of the terms is 1:

$$ \sum_{k=0}^{n} \binom{n}{k} x^k = (1+x)^n $$

We define a sequence by specifying an expression for its nth term:

```
>>> k = sp.symbols("k")
>>> a_k = 1 / k
>>> b_k = 1 / sp.factorial(k)
```

Substitute the desired value of n to see the value of the nth term:

```
>>> a_k.subs({k:5})
1/5
```

The Python list comprehension syntax `[item for item in list]` can be used to print the sequence values for some range of indices:

```
>>> [ a_k.subs({k:i}) for i in range(1,8) ]
[1, 1/2, 1/3, 1/4,  1/5,   1/6,   1/7]
>>> [ b_k.subs({k:i}) for i in range(0,8) ]
[1,  1, 1/2, 1/6, 1/24, 1/120, 1/720, 1/5040]
```

Observe that a_k is not defined for $k=0$ since $\frac{1}{0}$ is a division-by-zero error. In other words, the domain of a_k is the nonnegative natural numbers $a_k : \mathbb{N}_+ \to \mathbb{R}$. Observe how quickly the 'factorial' function $k! = 1 \cdot 2 \cdot 3 \cdots (k-1) \cdot k$ grows: $7! = 5040$, $10! = 3628800$, $20! > 10^{18}$.

We're often interested in calculating the limits of sequences as $k \to \infty$. What happens to the terms in the sequence when k becomes large?

```
>>> sp.limit(a_k, k, sp.oo)
0
>>> sp.limit(b_k, k, sp.oo)
0
```

Both $a_k = \frac{1}{k}$ and $b_k = \frac{1}{k!}$ *converge* to 0 as $k \to \infty$.

B. Series

Suppose we're given a sequence a_k and we want to compute the sum of all the values in this sequence $\sum_{k=-}^{\infty} a_k$. Series are sums of sequences. Summing the values of a sequence $a_k : \mathbb{N} \to \mathbb{R}$ is analogous to taking the integral of a function $f : \mathbb{R} \to \mathbb{R}$.

The formula for the geometric sequence is $a_k = r^k$. The sum of the first n terms in the geometric sequence is

$$ S_n = \sum_{k=0}^{n} r^k = 1 + r + r^2 + \cdots + r^n = \frac{1 - r^{n+1}}{1 - r}. $$

If $|r| < 1$, taking the limit $n \to \infty$ in the above expression leads to

$$ S_\infty = \lim_{n \to \infty} S_n = \sum_{k=0}^{\infty} r^k = 1 + r + r^2 + r^3 + \cdots = \frac{1}{1 - r}. $$

Example: Consider the geometric series with $r = \frac{1}{2}$. Applying the above formula, we obtain

$$S_\infty = \sum_{k=0}^{\infty} \left(\frac{1}{2}\right)^k = 1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \cdots = \frac{1}{1-\frac{1}{2}} = 2.$$

You can visualize this infinite summation graphically in Figure 17.
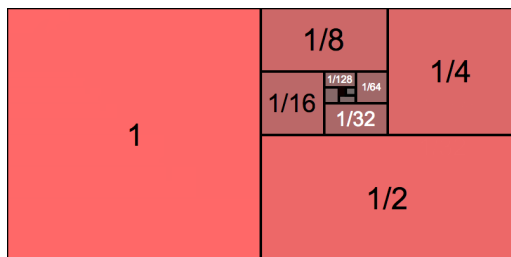

Fig. 17. A graphical representation of the infinite sum of the geometric series with $r = \frac{1}{2}$. The area of each region corresponds to one of the terms in the series. The total area is equal to $\sum_{k=0}^{\infty}(\frac{1}{2})^k = \frac{1}{1-\frac{1}{2}} = 2$.

To work with series in `SymPy`, use the `summation` function whose syntax is analogous to the `integrate` function:

```
>>> a_k = 1 / k
>>> sp.summation(a_k, (k,1,sp.oo))
oo
>>> b_k = 1 / sp.factorial(k)
>>> sp.summation(b_k, (k,0,sp.oo))
E
```

We say the series $\sum a_k$ *diverges* to infinity (or *is divergent*) while the series $\sum b_k$ converges (or *is convergent*). As we sum together more and more terms of the sequence b_k, the total becomes closer and closer to some finite number. In this case, the infinite sum $\sum_{k=0}^{\infty} \frac{1}{k!}$ converges to the number $e = 2.71828\ldots$.

The `summation` command is useful because it allows us to compute *infinite* sums, but for most practical applications we don't need to take an infinite number of terms in a series to obtain a good approximation. This is why series are so neat: they represent a great way to obtain approximations.

Using standard Python commands, we can obtain an approximation to e^5 that is accurate to six decimals by summing 10 terms in the series:

```
>>> import math
>>> def b_kf(n):
        return 1.0/math.factorial(n)
>>> sum( [b_kf(k) for k in range(0, 10)] )
2.7182815255731922
>>> sp.E.evalf()
2.718281 82845905        # true value
```

C. Taylor series

The *Taylor series* of a function $f(x)$ approximates the function by an infinitely long polynomial:

$$f(x) = \sum_{k=0}^{\infty} c_k x^k = c_0 + c_1 x + c_2 x^2 + c_3 x^3 + c_4 x^4 + \cdots.$$

Each term in the series is of the form $a_k = c_k x^k$, where the coefficient c_k depends on the properties of the function $f(x)$.

Specifically, $c_k = \frac{f^{(k)}(0)}{k!}$, where $f^{(k)}(0)$ is the kth derivative of $f(x)$ and $k!$ is the factorial function:

$$f(x) = f(0) + f'(0)x + \frac{f''(0)}{2!}x^2 + \frac{f'''(0)}{3!}x^3 + \frac{f^{(4)}(0)}{4!}x^4 + \cdots$$
$$= \sum_{k=0}^{\infty} \frac{f^{(k)}(0)}{k!}x^k.$$

Using this formula and your knowledge of derivatives, you can compute the Taylor series of any function $f(x)$.

For example, let's find the Taylor series of the function $f(x) = e^x$ at $x = 0$. The first derivative of $f(x) = e^x$ is $f'(x) = e^x$. The second derivative of $f(x) = e^x$ is $f''(x) = e^x$. In fact, all the derivatives of $f(x)$ will be e^x because the e^x is a special function that is equal to its derivative! The kth coefficient in the power series of $f(x) = e^x$ at the point $x = 0$ is equal to the value of the kth derivative of $f(x)$ evaluated at $x = 0$. In the case of $f(x) = e^x$ we have $f^{(k)}(0) = e^0 = 1$, so the coefficient of the kth term is $c_k = \frac{f^{(k)}(0)}{k!} = \frac{1}{k!}$. The Taylor series of $f(x) = e^x$ is

$$e^x = \sum_{k=0}^{\infty} \frac{1}{k!}x^k = 1 + x + \frac{x^2}{2} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!} + \cdots$$

Series are a powerful computational tool for approximating numbers and functions. As we compute more terms from the above series, our the polynomial approximation to the function $f(x) = e^x$ becomes more accurate. The exact value of the function at $x = 1$ is $f(1) = e^1 = e$. The partial sum of the first six terms (as shown above) gives us an approximation of e^1 that is accurate to three decimals. The partial sum of the first 12 terms gives us e to an accuracy of nine decimals.

A *power series* is a series whose terms contain different powers of the variable x. The kth term in a power series is a function of both the sequence index k and the input variable x.

For example, the power series of the function $\exp(x) = e^x$ is

$$\exp(x) \equiv 1 + x + \frac{x^2}{2} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!} + \cdots = \sum_{k=0}^{\infty} \frac{x^k}{k!}.$$

This is, IMHO, one of the most important ideas in calculus: you can compute the value of $\exp(5)$ by taking the infinite sum of the terms in the power series with $x = 5$:

```
>>> exp_xk = x**k / sp.factorial(k)
>>> sp.summation( exp_xk.subs({x:5}), (k,0,sp.oo)).evalf()
148.413159102577
>>> sp.exp(5).evalf()   # the true value
148.413159102577
```

Note that `SymPy` is actually smart enough to recognize that the infinite series you're computing corresponds to the closed-form expression e^5:

```
>>> sp.summation( exp_xk.subs({x:5}), (k,0,sp.oo))
exp(5)
```

The coefficients in the power series of a function (also known as the *Taylor series*) depend on the value of the higher derivatives of the function. The formula for the kth term in the Taylor series of $f(x)$ expanded at $x = c$ is $a_k(x) = \frac{f^{(n)}(c)}{k!}(x - c)^k$, where $f^{(k)}(c)$ is the value of the kth derivative of $f(x)$ evaluated at $x = c$.

The `SymPy` function `series` is a convenient way to obtain the Taylor series of any function. Calling `series(expr,var,at,nmax)` will show you the series expansion of `expr` near `var=at` up to power `nmax`:

```
>>> x = sp.symbols("x")
>>> sp.series( sp.sin(x), x, x0=0, n=8)
x - x**3/6 + x**5/120 - x**7/5040 + O(x**8)
>>> sp.series( sp.cos(x), x, x0=0, n=8)
1 - x**2/2 + x**4/24 - x**6/720 + O(x**8)
```
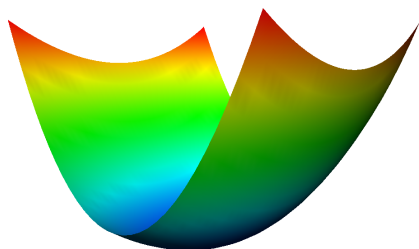
VII. Multivariable calculus

Multivariable calculus is the extension of the ideas of differential and integral calculus to functions like $f(x, y)$ that depend on multiple input variables. You can plot a function $f : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ as a *surface*, where the height z of the surface above the point (x, y) is function output $z = f(x, y)$.

If you know single-variable calculus (derivatives and integrals), then then you won't have much new math to learn in multivariable calculus: it's essentially the same concepts but with more variables.

A. Plotting multivariable functions

Suppose the height of a mountain is described by the function $f(x, y)$. The coordinates (x, y) tell us the horizontal position point in the xy-plane and the value of the function $f(x, y)$ represents the height of the mountain at those coordinate.

We identify the z coordinate with the hight of the mountain $z = f(x, y)$ and graph the function $f(x, y)$ is as a surface in 3D as illustrated in Figure 18.

TODO: redo using plt

Fig. 18. The 3D surface plot of the the function $f(x, y)$.

Three dimensional surface plots are very good for visualizing multivariable functions, but they can be difficult to draw by hand.

Another approach for representing function of the form $f(x, y)$ is to use a two-dimensional plot that shows the "view from above" of the surface $f(x, y)$. We can use colour to represent the height of the function through different sharing: darker-shading to represent large values of $f(x, y)$ and lighter-shading to represent small values of $f(x, y)$. We can also trace *level curves* in the plot, which is the same approach used for topographic maps: each level curve show the points that are at a certain height.
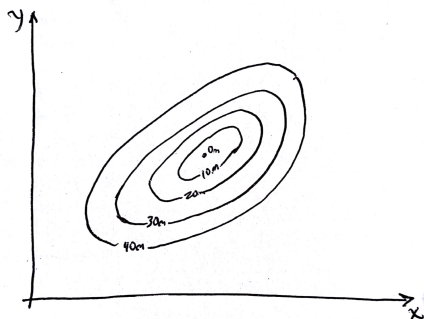
TODO: redo using plt surf

Fig. 19. The topographic map that shows the height of the function $f(x, y)$ using sharing to represent height. The level curves at each 10m intervals are also shown.

The curve labeled 30m line you see in Figure 19 represents the solution to the equation $30 = f(x, y)$, where $f(x, y)$ is the height of this hill for all coordinates (x, y) on the map.

B. Partial derivatives

For a function of two variables $f(x, y)$, there is an "x-derivative" operator $\frac{\partial}{\partial x}$ and a "y-derivative" operator $\frac{\partial}{\partial y}$. The operation $\frac{\partial}{\partial x} f(x, y)$ describes taking the derivative of $f(x, y)$ with respect to the input variable x, while keeping the input variable y constant. Taking the derivative of a multivariable function with respect to one of its input variables is called a *partial derivative* and denoted with the symbol ∂.

The partial derivative of $f(x, y)$ with respect to x is

$$\frac{\partial}{\partial x} f(x, y) = \frac{\partial f}{\partial x} \stackrel{\text{def}}{=} \lim_{\delta \to 0} \frac{f(x + \delta, y) - f(x, y)}{\delta}.$$

Similarly the partial derivative of with respect to y is

$$\frac{\partial}{\partial y} f(x, y) = \frac{\partial f}{\partial y} \stackrel{\text{def}}{=} \lim_{\delta \to 0} \frac{f(x, y + \delta) - f(x, y)}{\delta}.$$

Note that both $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial y}$ are function of x and y. Indeed, we can ask the questions "what is the slope in the x-direction" and "what is the slope in the y-direction" at any point (x, y) on the surface of the function. That's precisely the information returned by the functions $\frac{\partial f}{\partial x}(x, y)$ and $\frac{\partial f}{\partial y}(x, y)$.

TODO: example

C. Gradient

The operator ∇ is a combination of both the x and y derivatives:

$$\nabla f(x, y) \stackrel{\text{def}}{=} \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right).$$

Note that ∇ acts on a function $f(x, y)$ to produce a vector output. This vector is called the *gradient* vector and it tells you the combined x- and y-slopes of the surface. More specifically, the gradient vector tells you the direction of the function's maximum increase—the "uphill" direction at the surface of graph of $f(x, y)$ at the point (x, y). The gradient vector is always perpendicular to the *level curve* at that point.

D. Partial integration

We can perform integration with respect to one of the input variables:

$$f(y) = \int_{x \in \mathbb{R}} f(x, y) \, dx \quad \text{and} \quad f(x) = \int_{y \in \mathbb{R}} f(x, y) \, dy.$$

The result of this partial integration is a function of the variable that we didn't integrate.

E. Double integrals

The multivariable generalization of the integral $\int_{x \in I} f(x) \, dx$ that computes the "total" amount of $f(x)$ on some interval $I = [a, b]$ is the multivariable integral of the form:

$$\iint_{(x, y) \in R} f(x, y) \, dx dy,$$

where R is called the *region of integration* and corresponds to some subset of the cartesian plane $\mathbb{R} \times \mathbb{R}$. The idea behind multivariable integrals is the same as for single variable integrals—to compute the total amount of some function for some range of input values. For single-variable integrals, we split the region into thing rectangular strips of width dx. For double integrals we split the two-dimensional region of integration into small squares of area $dxdy$, and compute the total volume of a many vertical columns whose base area is $dxdy$ and whose height is given by the function $f(x, y)$.

TODO: insert graphic of 3D integral split into vertical columns

TODO: explain "sweep along x then sweep along y" idea + hint at change-of-variables techniques

F. Applications of multivariable calculus

Optimization: The notion of an uphill or downhill direction for the surface $f(x, y)$ turns out to be very useful for optimization. To find the lowest point on the surface (minimum value of $f(x, y)$), you can start at some point and keep moving downhill, that is in the opposite direction to the gradient $-\nabla f(x, y)$. Intuitively, this is the path that a water stream would take as it descends down the slope of the mountain until it reaches the minimum at the bottom of a valley. This intuitive notion of "keep moving downhill until you get to a local minimum" is the general idea behind the *gradient descent* optimization algorithm which is very important for machine learning applications.

VIII. Vector calculus

Vector calculus is the study of vector fields $\mathbf{F}$, which are functions of the form $\mathbf{F} : \mathbb{R}^3 \to \mathbb{R}^3$, which defines 3-dimensional output vector at each point (x, y, z) in space. For example, the electric field $\mathbf{E}(x, y, z)$ describes the strength and the direction of the electric force that a changed particle would experience if placed at (x, y, z).

Vector calculus is *waaaaay* out of scope for an introductory calculus tutorial, so I will just show you some simple definitions of the building blocks.

A. Definitions

- $\nabla \overset{\text{def}}{=} \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)$: the vector derivative operator (*nabla*).
- $\nabla \cdot \vec{F}(x, y, z)$: the *divergence* of the field $\vec{F}$ tells us if field $\vec{F}$ is acting as a "source" or a "sink" at the point (x, y, z).
- $\nabla \times \vec{F}(x, y, z)$: the *curl* of the field $\vec{F}$ tells us the "rotational tendency" of the vector field $\vec{F}$ at (x, y, z).

B. Path integrals

path integrals of vectors fields,

Scalar path integral.

$$\int_C f(\mathbf{r}) \, dr \equiv \int_{t_i}^{t_f} f(\mathbf{r}) \, \|\mathbf{r}'(t)\| \, dt.$$

Here the curve $C \in \mathbb{R}^3$ is described by the parametrization $\mathbf{r} : \mathbb{R} \to \mathbb{R}^3$, which assigns a coordinate $\mathbf{r} = (x, y, z)$ for each value

of the parameter (denoted t in the above). Note $dr \equiv \|\mathbf{r}'(t)\| \, dt$, which involves computing the derivative of $\mathbf{r}(t)$ then computing the length.

Vector path integral.

$$\int_C \mathbf{F}(\mathbf{r}) \cdot d\mathbf{r} \equiv \int_{t_i}^{t_f} \mathbf{F}(\mathbf{r}) \cdot \mathbf{r}'(t) \, dt.$$

This integral computes the total of the vector field $\mathbf{F}$ in the direction of the tangent line to the curve C describe by $\mathbf{r}(t)$. To obtain the component of $\mathbf{F}$ in the direction of the tangent line, we take the dot product with $d\mathbf{r} \equiv \mathbf{r}'(t)dt$ during each step.

C. Surface integrals

flux integrals of vectors fields through surfaces,

Scalar surface integral.

$$\iint_S f(\mathbf{r}) \, dS \equiv \int_{v_i}^{v_f} \int_{u_i}^{u_f} f(\mathbf{r}) \, \|\mathbf{r}'_u \times \mathbf{r}'_v\| \, dudv.$$

Here the surface $S \in \mathbb{R}^3$ is described by the parametrization $\mathbf{r} : \mathbb{R}^2 \to \mathbb{R}^3$, which assigns a coordinate $\mathbf{r} = (x, y, z)$ for each pair of the parameter u and v. Note $dS = \|\mathbf{r}'_u \times \mathbf{r}'_v\| \, dudv$, which involves computing the partial derivatives of $\mathbf{r}(u, v)$ with respect to the two parameters, taking the cross product, then computing the length.

Vector surface integral.

$$\iint_S \mathbf{F}(\mathbf{r}) \cdot d\mathbf{S} \equiv \int_{v_i}^{v_f} \int_{u_i}^{u_f} \mathbf{F}(\mathbf{r}) \cdot (\mathbf{r}'_u \times \mathbf{r}'_v) \, dudv.$$

This integral computes the total *flux* of the vector field $\mathbf{F}$ flowing perpendicularly through the surface S. To obtain the component of $\mathbf{F}$ in the direction of the surface normal, we take the dot product with $d\mathbf{S} \equiv \hat{\mathbf{n}}dS \equiv (\mathbf{r}'_u \times \mathbf{r}'_v)dudv$, for each piece of the surface.

The main thing we'll have to learn is how to parametrize regions of space. In fact, we could even say that the main purpose of this course is to get you comfortable with parameterizations of curves, surfaces, and volumes. Once you have a parametrization for a region you can perform any integral calculation over this region.

D. Vector calculus theorems

The main results in vector calculus are two theorems: *Gauss' divergence theorem* and *Stokes theorem*. Both theorems can be understood as extensions of the fundamental theorem of calculus (FTC), which relates the integral of the differential of some quantity over a region R to the value of this quantity on the boundary of a region, denoted ∂R. In the case of the fundamental theorem of calculus, the region is the interval $I = [a, b] \subseteq \mathbb{R}$ whose boundary ∂I consists of the two points a and b. The fundamental theorem of calculus is

$$\int_a^b f'(x) \, dx = \int_I f'(x) \, dx = f_{\partial I} = f(b) - f(a),$$

Gauss' Divergence Theorem relates the volume integral of the quantity $\nabla \cdot \vec{F}$, which is called the divergence of $\vec{F}$, to the

total flux of the vector field through the surface ∂V, which is the boundary of the volume V. Gauss' divergence theorem is:

$$\iiint_V \nabla \cdot \vec{F} \; dV = \iint_{\partial V} \vec{F} \cdot d\vec{S}$$

Intuitively, the *divergence* of a vector field describes how much of the vector field emanates from a given point in space. The *flux* of a vector field over a surface S accounts for the strength of the vector field flowing through the surface. In the above example, we saw Gauss' divergence theorem applied to the electric field, but the vector field $\vec{F}$ could also represent thermal flows, or fluid flows.

Stokes' Theorem uses the "other" vector derivative $\nabla \times \vec{F}$, which is called the *curl* of $\vec{F}$. The curl of a vector field, denoted $(\nabla \times \vec{F})(x, y, z)$ describes the local rotational tendency of the vector field $\vec{F}$ at the point (x, y, z). Given any surface S in space, we can cut up the surface into tiny little rectangles and calculate the total surface area as a double integral $S = \int dS$. Stokes' theorem is the application of this "splitting up into little squares" idea and the fundamental theorem of calculus, which leads us to the following equation.

$$\iint_\Sigma \nabla \times \vec{F} \cdot d\vec{S} = \int_{\partial \Sigma} \vec{F} \cdot d\vec{r},$$

The surface integral of the curl $\nabla \times \vec{F}$ over any surface Σ is equal to the circulation of $\vec{F}$ along the boundary of the surface $\partial \Sigma$. Both the left and right sides of this equation correspond to scalar numbers. The left side is the vector surface integral of a vector quantity (the curl of $\vec{F}$), the right side corresponds to a vector path integral of a vector quantity over an oriented curve $\partial \Sigma$.

E. Applications of vector calculus

Vector calculus is the math machinery used for electricity and magnetism, which is the study electric field $\mathbf{E}(x, y, z)$, the magnetic field $\mathbf{B}(x, y, z)$, and the interactions between them.

Practice problems

This means learning calculus is all about getting practical experience calculating limits, derivatives, and integrals of functions, which is best achieved by solving lots of problems.

TODO: add exercises

TODO: link to notebook for solutions

Links

I hope this tutorial helped you see as practical and useful math that allows you to do calculations—just look at the name of the thing!

[*Essence of calculus* series by 3Blue1Brown]
`https://tinyurl.com/CALCess`

[*Calculus made simple* by Silvanus P. Thompson]
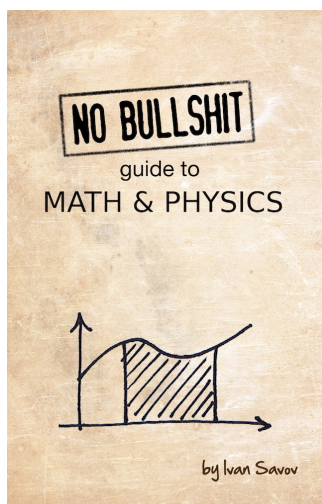`https://gutenberg.org/ebooks/33283`

If you want to learn more about calculus, I invite you to check out my book, the **No bullshit guide to math and physics**.

This book contains short lessons on mechanics, differential and integral calculus written in a style that is jargon-free and to the point. This textbook covers both subjects in an integrated manner and aims to highlight the connections between them.

Contents:

- HIGH SCHOOL MATH
- VECTORS
- MECHANICS
- DIFFERENTIAL CALCULUS
- INTEGRAL CALCULUS
- SEQUENCES AND SERIES

5½[in] × 8½[in] × 528[pages]

For more information, see the book's website `minireference.com` or you can get in touch with me by email here `ivan@minireference.com`.