Calculus tutorial

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

***Abstract*—aaa**

Contents

I. Introduction

Banking example: Consider the function $\mathrm{ba}(t)$ that represents your bank account balance at time t. Also consider the function $\mathrm{tr}(t)$, which corresponds to the transactions (deposits and withdrawals) on your account.

Suppose you have a record of all the transactions on your account $\mathrm{tr}(t)$, and you want to compute the final account balance at the end of the month $\mathrm{ba}(30)$. You can use the integration procedure on the transactions $\mathrm{tr}(t)$ to calculate the total change in the account balance at the end of the month, relative to the account balance at the beginning of the month $\mathrm{ba}(0)$. The end-of-the-month-balance calculation is described by the following equation:

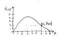
$$\mathrm{ba}(30) = \mathrm{ba}(0) + \int_0^{30} \mathrm{tr}(t)\, dt.$$

The integral $\int_0^{30} \mathrm{tr}(t)dt$ describes the process of computing the total of all the transactions that occurred between day 0 and day 30. The weird-looking integral sign "$\int$" comes from the Latin word *summa* for sum.

We use integrals every time we need to calculate the total of some quantity over a time period. 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 $t = b$.

II. Definitions

Let's start by defining all the concepts from university-level math you need to know about. Don't worry if you're seeing some of these concepts for the first time, you'll see plenty of examples using these concepts, so you'll get to know them very well by the end of this section.

- *set*: a collection of math objects. Sets are denoted using curly brackets $\{\ldots\}$. A set can be defined as a finite list of elements like $\{\texttt{heads}, \texttt{tails}\}$, by specifying a pattern $\{0, 1, 2, 3, \ldots\}$, or through some other math expression {<def'n>}.
- $f(x)$: 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. Functions are usually defined through an analytical formula like $f(x) = x^2$, which tells us how to compute the output $f(x)$ for a given input x. Functions can also be represented visually as a function graph , which is a curve that passes through all the coordinates pairs $(x, f(x))$ in the Cartesian plane.
- $A_f(a, b)$: the value of the *area* under the graph of the function $f(x)$ from $x = a$ until $x = b$. The area $A_f(a, b)$ corresponds to the following integral
$$A_f(a,b) \stackrel{\text{def}}{=} \int_a^b f(x)\, dx.$$
The $\int$ sign stands for *sum*. Indeed, the integral is the "sum" of all the values of $f(x)$ for inputs x between $x = a$ and $x = b$.
- $F_0(b) \stackrel{\text{def}}{=} A_f(0, b)$: 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:
$$F(b) \stackrel{\text{def}}{=} A_f(0,b) = \int_0^b f(x)\, dx.$$
The choice of $x = 0$ as the lower limit of integration is arbitrary. We could define any number of other integral functions $F_a(b)$ for different starting points $x = a$.

In the next few pages, we'll go into some details about each of these math concepts. Don't be intimidated by all the fancy-looking math notation—it's just a bunch of language mathematicians invented in order to describe concepts precisely and concisely. It looks weird to everyone who sees this specialized math notation for the first time (a.k.a. alien symbols), but you'll quickly get used to it.

III. 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.
- S^c: the *complement* of the set S, is defined as all elements that are not in the set S.
- $\mathbb{N}$: the set of natural numbers $\mathbb{N} \stackrel{\text{def}}{=} \{0, 1, 2, \ldots\}$
- $\mathbb{Z}$: the set of integers $\mathbb{Z} \stackrel{\text{def}}{=} \{\ldots, -2, -1, 0, 1, 2, 3, \ldots\}$
- $\mathbb{Q}$: the set of rational numbers, $\mathbb{Q} \stackrel{\text{def}}{=} \left\{ \frac{m}{n} \,\middle|\, m \in \mathbb{Z},\ n \in \mathbb{N},\ n \neq 0 \right\}$. The set $\mathbb{Q}$ consists of all numbers that can be expressed as *fractions* of the form $\frac{m}{n}$, where m is an integer, n is a natural number, and $n \neq 0$.
- $\mathbb{R}$: the set of real numbers
- $\mathbb{R}_+$: the set of nonnegative real numbers. The definition of the nonnegative is written as $\mathbb{R}_+ \stackrel{\text{def}}{=} \{\text{all } x \text{ in } \mathbb{R} \text{ such that } x \geq 0\}$, or it can be expressed more compactly as $\mathbb{R}_+ \stackrel{\text{def}}{=} \{x \in \mathbb{R} \mid x \geq 0\}$.

Note the multiple ways we use the curly-brackets notation $\{\}$ to denote sets. A *finite set* is defined by simply listing all its elements. For example, the set of possible outcomes of a coin flip is $\{\texttt{heads}, \texttt{tails}\}$. For an infinite set we can't write down all the elements, but we can show the pattern like $\mathbb{N} \stackrel{\text{def}}{=} \{0, 1, 2, 3, 4, \ldots\}$. The meaning of the three dots is "and so on, continuing the same pattern." Another way to define a set is to use 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 the rational numbers $\mathbb{Q}$ and the nonnegative real numbers $\mathbb{R}_+$ above are examples 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 ∞.

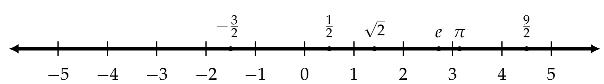


Fig. 1. The real numbers $\mathbb{R}$ cover the entire number line.

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.

A. 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.

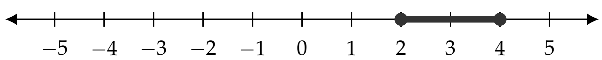


Fig. 2. The interval $[2, 4] \stackrel{\text{def}}{=} \{x \in \mathbb{R} \mid 2 \leq x \leq 4\}$.

B. *Set operations*

We use set operations like union $\cup$, intersection $\cap$, and set difference $\setminus$ to define composite sets.

- $S \cup T$: the *union* of two sets. The union of S and T corresponds to the elements in either S or T.
- $S \cap T$: the *intersection* of two sets. The intersection of S and T corresponds to the elements that are in both S and T.
- $S \setminus T$: *set difference* or *set minus*. The set difference $S \setminus T$ corresponds to the elements of S that are not in T.

Consider the overlapping intervals $A = [a, b]$ and $B = [c, d]$ illustrated in Figure 3. The union of these two intervals is the set of numbers that are *either* between a and b *or* between c and d, which corresponds to the interval $[a, d]$. The intersection of A and B is the set of numbers that are in *both* A and B, and corresponds to the interval $[c, b]$. The figure also illustrates the two set differences, $A \setminus B$ and $B \setminus A$ which correspond to numbers that are in one set, but not in the other.

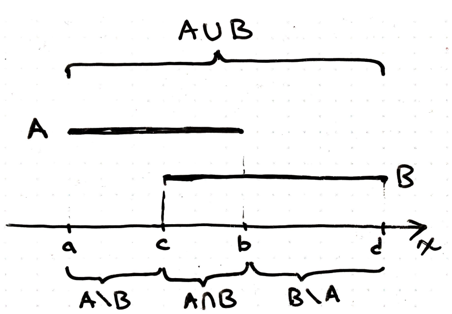


Fig. 3. Various intervals that can be obtained using set operations of the intervals A and B.

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.

In probability theory, we use finite sets and countably infinite sets like the natural numbers to represent the sample spaces of discrete random variables. We also use intervals to describe outcomes in the sample space of continuous random variables.

IV. Functions

A *function* is a mathematical object that takes numbers as inputs and produces numbers as outputs. We use the notation

$$f\colon A \to B$$

to denote a function from the input set A to the output set B. For every input x, the output value of f for that input is denoted $f(x)$.

A. 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 4.

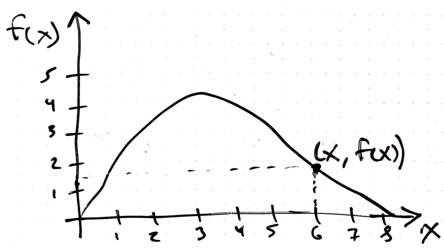


Fig. 4. The graph of the function f consists of all the points with coordinates $(x, f(x))$ over some interval of x values.

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.

In probability theory, we use functions to describe the probability distributions of random variables. Discrete random variables are described by a probability mass function of the form $f\colon \mathcal{X} \to \mathbb{R}$, where the sample space $\mathcal{X}$ is either a finite set or a countably infinite set like the natural numbers $\mathbb{N}$. Continuous random variables are described by probability density functions of the form $f\colon \mathcal{X} \to \mathbb{R}$, where the sample space $\mathcal{X}$ is some subset of the real numbers $\mathbb{R}$.

B. Plotting function graphs using NumPy and Seaborn

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

```
>>> def g(x):
        return 0.5 * x**2
>>> import numpy as np
>>> import seaborn as sns
>>> xs = np.linspace(0, 10, 1000)
>>> gxs = g(xs)
>>> sns.lineplot(x=xs, y=gxs, label="Graph of g(x)")
See Figure 5 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 $x = 0$ until $x = 10$. Next we apply the function g to the array of inputs `xs` and store the result in the array `gxs`, which contains all the output values of the function for the input values `xs`. At this point, the arrays `xs` and `gxs` contain 1000 input-output pairs of the form $(x, g(x))$, 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 $g(x)$, which produces the plot shown in Figure 5.

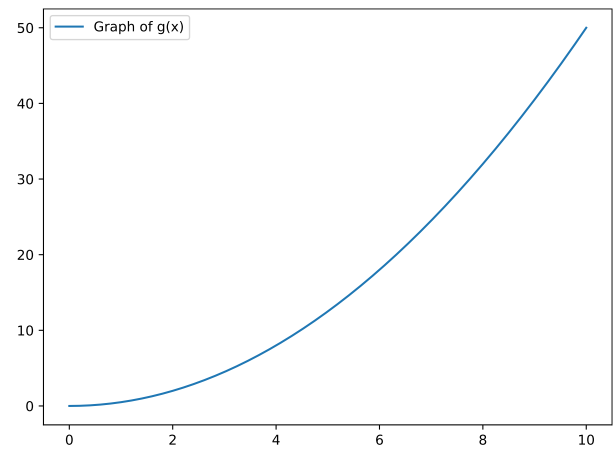


Fig. 5. Graph of the function $g(x) = x$ from $x = 0$ until $x = 10$.

C. 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 6 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.

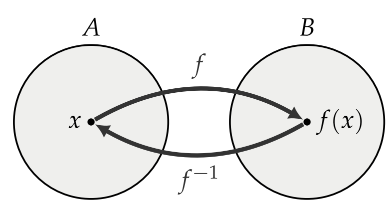


Fig. 6. The inverse f^{-1} undoes the operation of the function f.

For example, if we compute the square root of a number, then square the result, we obtain the original number, since the quadratic function x^2 is the inverse of the square-root function $\sqrt{x}$.

```
>>> np.sqrt(4)**2
4.0
```

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

```
>>> np.exp(np.log(4))
4.0
```

In probability theory, we often do calculations using the cumulative distribution function (CDF) $F_X \colon \mathcal{X} \to [0,1]$, and also use the inverse of the cumulative distribution function $F_X^{-1} \colon [0,1] \to \mathcal{X}$. Knowing about inverse functions (and the weird superscript $^{-1}$ notation used to describe them) is useful for your conceptual understanding of these concepts: instead of thinking about the inverse-CDF F_X^{-1} as some new complicated concept you have to memorize, you can think of F_X^{-1} as the "undo operation" for F_X. In other words, F_X and F_X^{-1} describe the same mapping, but used in opposite directions.

V. Limits

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.

Limit expressions provide a precise mathematical language for talking about infinitely large numbers, infinitely small steps, 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."
- $\lim_{n\to\infty} \text{proc}(n)$: limit expression that describes the value of $\text{proc}(n)$ as the integer n tends to infinity. The integer n usually describes the number of steps in a given procedure, and $\text{proc}(n)$ describes the output of this procedure when n steps are used.
- $\lim_{\delta\to 0} h(\delta)$: 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.

The SymPy function `limit` allows us to compute limit expressions. For example, if we want to see if the exponential function e^x or the polynomial function x^{100} grows faster in the limit as x goes to infinity, The code for computing the limit of the ratio between these two expressions is

```
>>> from sympy import limit, exp, oo
>>> limit(exp(x)/x**100, x, oo)
oo
```

The answer ∞, written as `oo` (two lowercase letters "o"), tells us exponential functions grow faster than polynomial functions.

Limits are important in calculus because they are used in the formal definitions of the derivative and integral operations. The derivative is defined as a rise-over-run calculation for an infinitely short run. The integral is defined as a Riemann sum with infinitely narrow rectangles. We'll explain both of these in the next sections.

Example: Let's begin with a simple example. Say you have a string of length ℓ and you want to divide it into infinitely many, infinitely short segments. There are infinitely many segments, and they are infinitely short, so together the segments add to the string's total length ℓ.

It's easy enough to describe this process in words. Now let's describe the same process using the notion of a limit. If we divide the length of the string ℓ into N equal pieces then each piece will have a length of

$$\delta = \frac{\ell}{N}.$$

Let's make sure that N pieces of length δ added together equal the string's total length:

$$N\delta = N\frac{\ell}{N} = \ell.$$

Now imagine what happens when the variable N becomes larger and larger. The larger N becomes, the shorter the pieces of string will become. In fact, if N goes to infinity (written $N \to \infty$), then the pieces of string will have zero length:

$$\lim_{N\to\infty} \delta = \lim_{N\to\infty} \frac{\ell}{N} = 0.$$

In the limit as $N \to \infty$, the pieces of string are *infinitely small*.

Note we can still add the pieces of string together to obtain the whole length:

$$\lim_{N\to\infty} (N\delta) = \lim_{N\to\infty} \left(N\frac{\ell}{N} \right) = \ell.$$

Even if the pieces of string are *infinitely small*, because there are *infinitely many* of them, they still add to ℓ.

The take-home message is that as long as you clearly define your limits, you can use infinitely small numbers in your calculations. The notion of a limit is one of the central ideas in this course.

A. *Limits at infinity*

In math, we're often interested in describing what happens to a certain function when its input variable tends to infinity. This information helps us draw the function's graph. Does $f(x)$ approach a finite number, or does it keep on growing to ∞?

As an example of this type of calculation, consider the limit of the function $f(x) = \frac{1}{x}$ as x goes to infinity:

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

This statement is true, even though the function $\frac{1}{x}$ never *actually* reaches zero. The function gets closer and closer to the x-axis but never touches it. This is why the concept of a limit is useful: it allows us to write $\lim\limits_{x\to\infty} f(x) = 0$ even though $f(x) \neq 0$ for any $x \in \mathbb{R}$.

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 7 for an illustration. The limit expression is a concise way of saying the following precise mathematical statement: for *any* precision $\epsilon > 0$, there exists a starting point S, after which $f(x)$ equals L within a precision ϵ.

The precise mathematical meaning of $\lim\limits_{x\to\infty} f(x) = L$ is

$$\forall \epsilon > 0 \; \exists S \in \mathbb{R} \text{ such that } \; \forall x \geq S \quad |f(x) - L| < \epsilon.$$

I know what you are thinking. Whoa! What just happened here? Chill. I know we saw that upside-down-A and backward-E

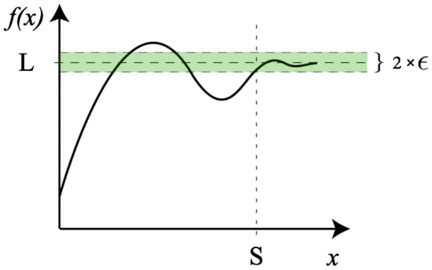


Fig. 7. A function $f(x)$ whose oscillations around the value L are smaller than ϵ for all $x \geq S$. The starting point S depends on the choice of precision ϵ.

business all the way back in Chapter **??** (see page **??**), so let me rewrite the expression for you in plain English:

For all $\epsilon > 0$,
  there exists a number S such that
    $|f(x) - L| < \epsilon$ for all x greater than or equal to S.

The limit equation $\lim\limits_{x\to\infty} f(x) = L$ states that the "limit at infinity" of the function $f(x)$ is equal to the number L. This statement is true if and only if there exists a winning strategy for an ϵ,S-game, similar to the ϵ,N-game played by the computer scientist and the mathematician. In the new ϵ,S-game, the mathematician specifies the precision ϵ, and the computer scientists must find a starting point S after which $f(x)$ becomes (and stays) ϵ-close to the limit L. If the computer scientist can succeed for all levels of precision ϵ, then the mathematician will be convinced the equation $\lim\limits_{x\to\infty} f(x) = L$ is true.

Example 2: Calculate $\lim\limits_{x\to\infty} \frac{2x+1}{x}$.

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. We can rewrite the function as $\frac{2x+1}{x} = 2 + \frac{1}{x}$ to more easily see what is going on:

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

since $\frac{1}{x}$ tends toward zero for large values of x.

In an introductory calculus course, you will not be required to give formal proofs for statements like $\lim_{x\to\infty} \frac{1}{x} = 0$; instead, you can assume the result is obvious and needs no proof. As the denominator x becomes larger and larger, the fraction $\frac{1}{x}$ becomes smaller and smaller.

B. 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. Figure 8 shows the graph of a function $f(x)$ near the point $(a, f(a))$. To prove the statement

$$\lim_{x\to a^+} f(x) = L,$$

you must show that

$$\begin{aligned}&\forall \epsilon > 0,\\ &\quad \exists \delta > 0 \text{ such that}\\ &\qquad \forall x \in (a, a+\delta) \quad |f(x) - L| < \epsilon.\end{aligned}$$

In other words, the limit from the right corresponds to an ϵ,δ-game in which the mathematician specifies the precision $\epsilon > 0$, and the computer scientist must find a distance $\delta > 0$, such that $|f(x) - L| < \epsilon$, for all x in the range $(a, a + \delta)$.

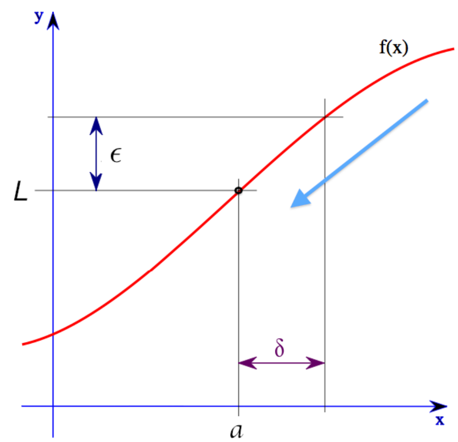


Fig. 8. A function $f(x)$ whose variation around the value L is smaller than ϵ for all x in the interval $(a, a + \delta)$. The value δ depends the choice of ϵ.

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 $\lim\limits_{x\to a} f(x) = L$, we say the function $f(x)$ is continuous at $x = a$.

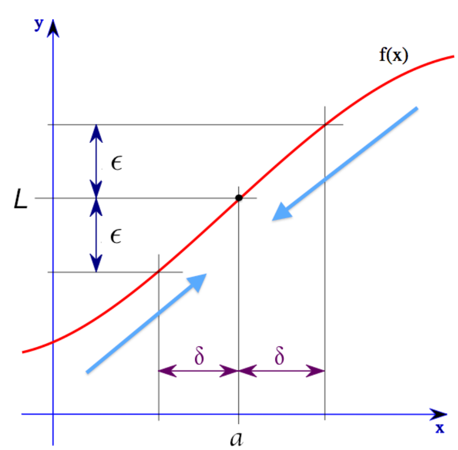


Fig. 9. The two-sided limit $\lim_{x\to a} f(x) = L$ exists if both the limit from the left and the limit from the right exist and are equal to L.

C. 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. Functions that are not defined for some value, as well as functions that make sudden jumps, are not continuous.

For example, consider the function $f : \mathbb{R} \setminus \{0\} \to \mathbb{R}$ defined by

$$f(x) = \frac{|x-3|}{x-3} = \begin{cases} 1 & \text{if } x > 3, \\ -1 & \text{if } x < 3. \end{cases}$$

The function f is continuous everywhere on the real line except at $x = 3$. Since this function f is "missing" only at a single point, we can try to "patch it" by filling in the missing value. Consider the function $g : \mathbb{R} \to \mathbb{R}$ defined as

$$g(x) = \begin{cases} 1 & \text{if } x > 3, \\ 1 & \text{if } x = 3, \\ -1 & \text{if } x < 3. \end{cases}$$

The function g is *continuous from the right* at the point $x = 3$, since $\lim_{x\to 3^+} g(x) = 1 = g(3)$. However, taking the limit from the left, we find $\lim_{x\to 3^-} g(x) = -1 \neq g(3)$, which tells us g is not continuous from the left. We say the function g has a *jump discontinuity* at $x = 3$.

Example 3: We can calculate the limit $\lim\limits_{x\to 5} \dfrac{2x+1}{x}$ as follows:

$$\lim_{x\to 5} \frac{2x+1}{x} = \frac{2(5)+1}{5} = \frac{11}{5}.$$

There is nothing tricky going on here—we plug the number 5 into the equation, and voila. The function $f(x) = \frac{2x+1}{x}$ is continuous at the value $x = 5$, so the limit of the function as $x \to 5$ is equal to the value of the function $\lim\limits_{x\to 5} f(x) = f(5)$.

D. Applications of limits

Limits are fundamentally important for calculus. Indeed, the three main calculus topics we'll discuss in the remainder of this chapter are derivatives, integrals, and series—all of which are defined using limits.

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 **??**.

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 $\epsilon = \frac{b-a}{N}$ and height $f(x)$, and then calculate the sum of the areas of the rectangles:

$$A(a,b) \approx \underbrace{\epsilon f(a) + \epsilon f(a+\epsilon) + \epsilon f(a+2\epsilon) + \cdots + \epsilon f(b-\epsilon)}_{N \text{ terms}}.$$

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:

$$\int_a^b f(x)dx = A(a,b) = \lim_{N\to\infty} \left[\epsilon f(a) + \epsilon f(a+\epsilon) + \epsilon f(a+2\epsilon) + \cdots + \epsilon f(b-\epsilon)\right].$$

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 **??**.

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 $a_n = r^n$ corresponds to the following expression:

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

The *series* a_n is defined as the limit $N \to \infty$ of the above expression. For values of r that obey $|r| < 1$, the series converges:

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

To convince yourself the above formula is correct, observe how the infinite sum S_∞ is similar to a shifted version of itself: $S_\infty = 1 + rS_\infty$. Now solve for S_∞ in the equation $S_\infty = 1 + rS_\infty$.

You'll find more about series in Section **??**.

VI. 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. In general, the slope of a function is different at different values of x, so mathematicians invented a new notation for describing "the slope (rate of change) of the function $f(x)$" and obtained formulas for finding the derivative of any function.

The derivative function $f'(x)$ is defined as the rate of change of the function f at x:

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

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 10 shows the slope calculation for the function $f(x) = \frac{1}{2}x^2$ for two different values of x: $x = -0.5$ and $x = 1$.

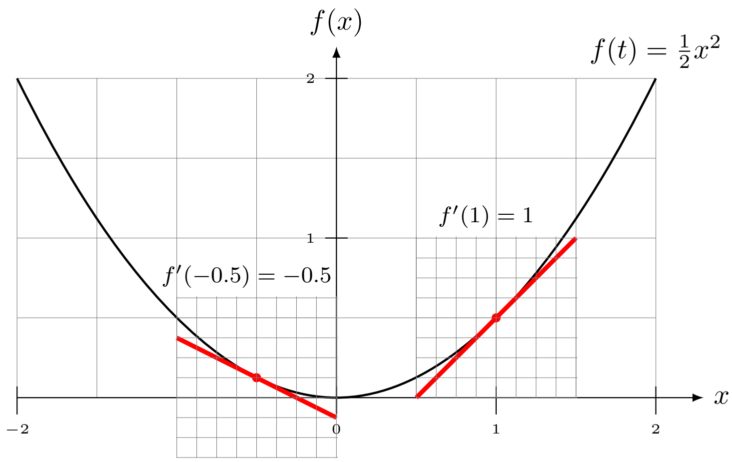


Fig. 10. The derivative of the function at $x = a$ is denoted $f'(a)$ and describes the slope function at that point.

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

LETS SEE SOME CODE

Here is a simple computer program for computing the numerical approximations to the derivative of any function at any point:

```
def differentiate(func, x, delta):
    """
    Compute the slope of the function `func` at `x` using
    the rise-over-run calculation with run of length `delta
    """
    rise = func(x+delta) - func(x)
    run = delta
    return rise/run
```

You can then define the function $f = \frac{1}{2}x^2$ using the code

```
def f(x):
    return 0.5*x**2
```

and compute the derivative at any point x by calling the function `differentiate(f,x,delta)`, where `delta` is some small number.

Here are some of the outputs of the differentiation procedure at $x = 1$ using different values of horizontal "run" parameter `delta`:

- The output of `differentiate(f,x=1,delta=0.01)` is 1.005, which is within 0.5% of the exact answer $f'(1) = 1$.
- The output of `differentiate(f,x=1,delta=0.001)` is 1.0005, which is within within 0.05% accuracy of the exact answer.
- The output for `differentiate(f,1,0.0001)` is 1.00005, which is an even more accurate approximation of the exact value.
- The output of `differentiate(f,1,0.00001)` is 1.000005.

Note the approximations get more and more accurate as the parameter `delta` decreases. For most practical applications, we can always choose a sufficient small `delta` so the fact that numeric approximations computed are a little bit "off" does not become a problem.

B. Derivative formulas

You don't need to apply the complicated derivative formula $f'(x) \equiv \lim_{\delta\to 0} \frac{f(x+\delta)-f(x)}{\delta}$ 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 it comes up again in a calculation. Indeed, that's what most scientists and engineers do—whenever they need to know the derivative of some function $f(x)$, the lookup the answer in a table of derivative formulas.

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

$$\begin{array}{rcl} f(x) & -\text{ derivative} \to & f'(x) \\ a & -\frac{d}{dx}\to & 0 \\ \alpha f(x) + \beta g(x) & -\frac{d}{dx}\to & \alpha f'(x) + \beta g'(x) \\ x & -\frac{d}{dx}\to & 1 \end{array}$$

$$\begin{array}{rcl}
x^n & \xrightarrow{\frac{d}{dx}} & nx^{n-1} \\
\frac{1}{x} \equiv x^{-1} & \xrightarrow{\frac{d}{dx}} & \frac{-1}{x^2} \equiv -x^{-2} \\
\sqrt{x} \equiv x^{\frac{1}{2}} & \xrightarrow{\frac{d}{dx}} & \frac{1}{2\sqrt{x}} \equiv \frac{1}{2}x^{-\frac{1}{2}} \\
e^x & \xrightarrow{\frac{d}{dx}} & e^x \\
a^x & \xrightarrow{\frac{d}{dx}} & a^x \ln(a) \\
\ln(x) & \xrightarrow{\frac{d}{dx}} & \frac{1}{x} \\
\log_a(x) & \xrightarrow{\frac{d}{dx}} & (x\ln(a))^{-1} \\
\sin(x) & \xrightarrow{\frac{d}{dx}} & \cos(x) \\
\cos(x) & \xrightarrow{\frac{d}{dx}} & -\sin(x) \\
\tan(x) & \xrightarrow{\frac{d}{dx}} & \sec^2(x) \equiv \cos^{-2}(x)
\end{array}$$

We'll be using these derivative formulas a lot in Section **??**, so it's a good idea 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 you should know about. These rules will allow you to find derivatives of *composite* functions.

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

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

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:

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

The product rule also applies to the product of three functions $f(x)g(x)h(x)$, for which the derivative is

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

For each term, take the derivative of one of the functions and multiply this derivative by the other two functions:

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 $g'(x)$.

The chain rule so applies to functions of functions of functions $f(g(h(x)))$. To take the derivative, start from the outermost function and work your way toward x.

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

D. *Examples*

The above rules define *all* you need to know to take 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: To compute the derivative of $f(x) = \sin(x)e^{x^2}\ln(x)$, we apply the product rule for three terms: $f'(x) = \cos(x)e^{x^2}\ln(x) + \sin(x)2xe^{x^2}\ln(x) + \sin(x)e^{x^2}\frac{1}{x}$.

Example 4: The derivative of $\sin(x^2)$ requires using the chain rule: $\left[\sin(x^2)\right]' = \cos(x^2)\left[x^2\right]' = \cos(x^2)2x$.

Example 5: The derivative of $\sin(\ln(x^3))$ requires the triple chain rule:

$$\begin{aligned}
\left[\sin(\ln(x^3))\right]' &= \cos(\ln(x^3))\left[\ln(x^3)\right]' \\
&= \cos(\ln(x^3))\tfrac{1}{x^3}\left[x^3\right]' \\
&= \cos(\ln(x^3))\tfrac{3}{x}\,.
\end{aligned}$$

Simple, right?

E. *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. Derivatives are also useful for solving optimization problems, which consist of finding the maximum or minimum value of some function $f(x)$.

Optimization techniques form a key building block for many machine learning algorithms, so it's good to know a derivative of two if you're going to learn about machine learning topics. In this book, we won't go too far so a "superficial" familiarity with the concept will be sufficient, but if you want to pursue machine learning topics in more depth you'll have to read up more on derivatives.

The derivative operation is also important because it is the "inverse operation" of the integration operation, which is the subject we'll discuss in the next section. Skip ahead to page **??** if you want spoilers about the inverse relationship between differentiation and integration, or read on to watch the calculus movie in order.

F. Optimization algorithms

One of the most prominent applications of derivatives is *optimization*: the process of finding a function's maximum and minimum values.

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_0 near the minimum 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.

In this book, we won't discuss the details behind optimization algorithms, and instead rely on the computational tools available in `numpy`, `scipy`, and `sympy` to do optimization-type calculations for us. We'll encounter optimization ideas (maximization and minimization) in several concepts in statistics: *maximum likelihood* and *least squares*, and rely on "visual proofs" for these optimization procedures. If you're interested in attaining a deeper understanding of optimization algorithms, you can follow the links provided at the end of this section, but note such "under the hood" understanding is not required to continue with the rest of the book.

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
>>> def f(x):
        return (x-5)**2
>>> 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.

VII. Integrals

A. Integrals as area calculations

An integral corresponds to the computation of the *area* enclosed between the curve $f(x)$ and the x-axis over some interval of x values:

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

We refer to the numbers a and b as the *limits of integration*, and the notation $\int_a^b f(x)\,dx$ is shorthand for $\int_{x=a}^{x=b} f(x)\,dx$.

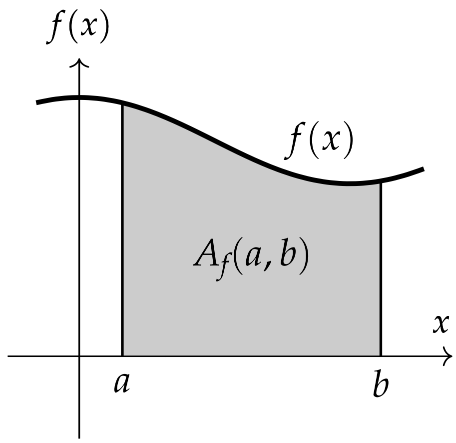


Fig. 11. The integral of the function $f(x)$ between $x = a$ and $x = b$ corresponds to the shaded area.

The notion of an integral is foundational for understanding continuous random variables. Every time we compute the probability of some outcome of a continuous random variable, there is an integral calculation going on under the hood, so integrals is not a topic you can skip.

If this is the first time you're learning about integrals, it's understandable if you feel intimidated by the complicated math notation, but you have to trust me on this one: except for the notation, there is nothing to worry about! In the next few pages, I'll do my best to introduce you to the topic of integrals, and you'll learn three different ways to do compute integrals.

Let's start with 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 the function $f(x)$ between any two points, since the region under the graph has a rectangular shape. See Figure 12 for an illustration.

The area under $f(x)$ between $x = 0$ and $x = 5$ corresponds to the following 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 $3 \cdot 5 = 15$.

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 13. Since the region under the curve is triangular, we can compute its area using the formula for the area of a triangle, which is "base times height divided by 2."

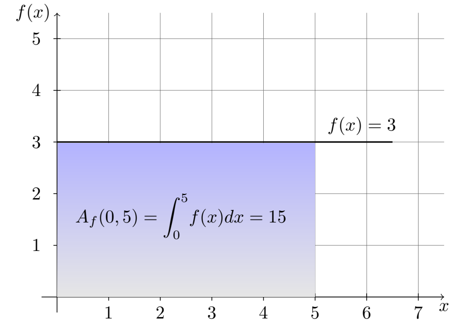


Fig. 12. The area of a rectangle of height 3 and width 5 is equal to 15.

The integral of $g(x)$ from $x = 0$ until $x = 5$ is described by the following calculation:

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

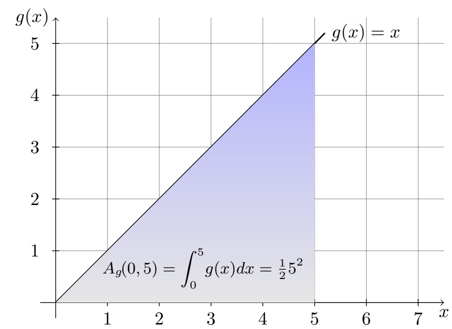


Fig. 13. The area of a triangle with base 5 and height 5 is equal to $\frac{1}{2}5^2 = \frac{25}{2} = 12.5$.

I hope these examples helped you see 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.

B. 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:
$$\int_a^b f(x)\,dx + \int_b^c f(x)\,dx = \int_a^c f(x)\,dx.$$
- **Constant multiple of a function.** The integral of the function $cf(x)$ is equal to c times the integral of $f(x)$, for any constant c:
$$\int cf(x)\,dx = c\int f(x)\,dx.$$
- **Sum of two functions.** The integral of a sum of two functions is equal to the sum of the integrals of the individual functions:
$$\int [f(x) + g(x)]\,dx = \int f(x)\,dx + \int g(x)\,dx.$$

- **Linearity.** The combination of the above two properties tells us that integration is a *linear* operation, meaning 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.

LETS SEE SOME CODE

Relax, we won't be doing the calculation by hand. We can write a computer program and make a computer performs the integration procedure for us. Here is a sample code that takes an arbitrary function `func` and performs the n-rectangle area approximation calculation:

```
def integrate(func, a, b, n):
    """
    Compute the area under `func` between x=`a` and x=`b`
    using an approximation with `n` rectangles.
    """
    dx = (b-a)/n
# width of each rectangle
    total = 0.0
# accumulator variable for S_n
    k = 1                              # counter variable
    x = a + dx
# start at first right endpoint
    while k <= n:                      # repeat n times:
        total += func(x)*dx            #
s_k = height * width
        x += dx                        #
move one step to the right
        k += 1                         #   increment counter
    return total
```

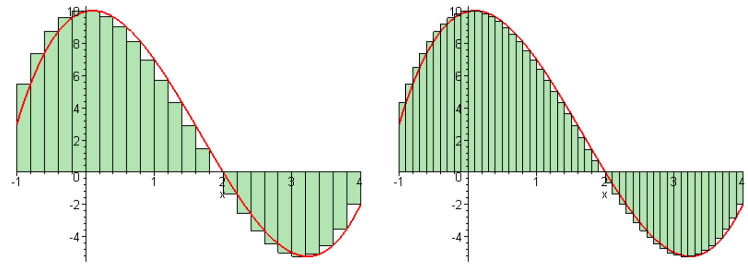
The logic of the of the sample code follows closely follows procedure we defined in the equations above. We variable `dx` holds the information about the width of the rectangles used in the approximation Δx, and we use the counter variable `k` to step through the interval $[a, b]$ using n steps of width Δx.

We can then use this code to compute the integral of any function. To do this, we must first define the function we want to integrate:

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

Then you can compute S_{25} by calling `integrate(f, -1, 4, 25)`, which returns $S_{25} = 12.4$. Calling `integrate(f, -1, 4, 50)` you'll obtain $S_{50} = 12.6625$. The approximations S_n get better and better as the number of rectangles used in the approximations grows. 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.

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



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

Fig. 14. 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.

of rectangles grows to infinity is known as the *Riemann sum* and is the basis for the definitions of the integral:

The definite integral between $x = a$ and $x = b$ is *defined* as the limit of a Riemann sum as n goes to infinity:
$$\int_a^b f(x)\, dx \equiv \lim_{n\to\infty} \sum_{k=1}^{n} f(a + k\Delta x)\Delta x \equiv A(a, b).$$

C. 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:
$$F_0(b) \;\overset{\text{def}}{=}\; A_f(0, b) = \int_{x=0}^{x=b} f(x)\, dx\,.$$
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 15.

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

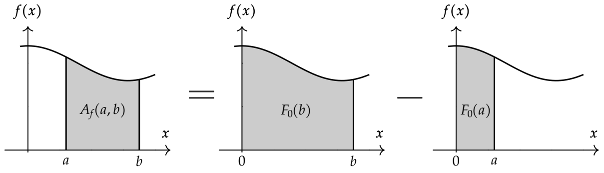


Fig. 15. The area under $f(x)$ between $x = a$ and $x = b$ is computed using the formula $A_f(a, b) = F_0(b) - F_0(a)$, which describes the change in the output of $F_0(x)$ between $x = a$ and $x = b$.

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 16.

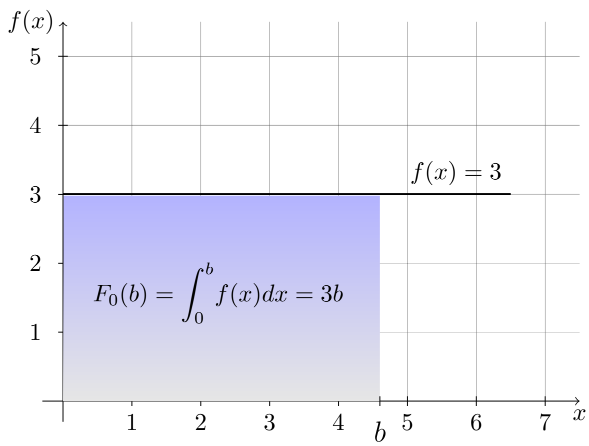


Fig. 16. The area of a rectangle of height 3 and width b is equal to $3b$.

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:

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

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 $A_g(a, b) = G_0(b) - G_0(a) = \frac{1}{2}b^2 - \frac{1}{2}a^2$.

We were able to compute the above integrals thanks to the simple geometries of the areas under the graphs. Computing integrals of more complicated functions requires more advanced techniques. There is an entire course called integral calculus which is dedicated to the task of finding integrals using various tricks and techniques.

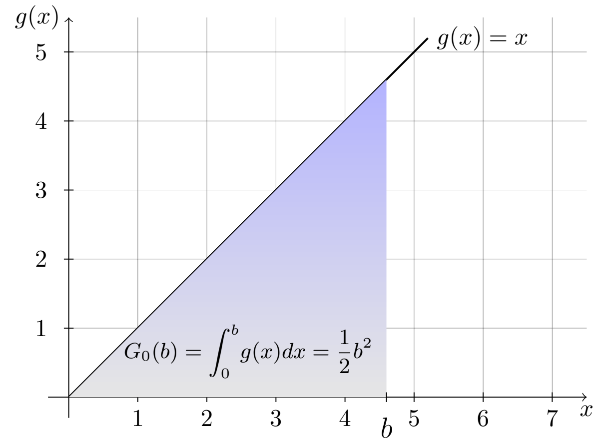


Fig. 17. The area of a triangle with base b and height b is equal to $\frac{1}{2}b^2$.

Taking a calculus course would be useful if you plan to study physics or engineering, but for the purpose of learning probability and statistics, you're not required to learn all these integration techniques. Instead, you can rely on computers to do integration for you. Specifically, you can use the Python modules SciPy and SymPy to compute all the integrals you need, as we'll show in the next two sections.

D. *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.

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 1 revisited again: 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 12) 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 16 for an illustration of the integral function $F_0(b)$.

Example 2 revisited again: 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 17.

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`.

E. 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 $\frac{d}{dx}[\cdot]$.

The integral function $F_a(x)$ is obtained from the original function $f(x)$ using integration, $F_a(x) = \int_a^x f(u)du$. Another way to describe this is to say we *applied* the integration operator $\int \cdot dx$ on the function $f(x)$ to obtain the integral function $F_a(x)$. The derivative function $f'(x)$ is defined by the formula $f'(x) = \lim_{\delta\to 0} \frac{f(x+\delta)\ -\ f(x)}{\delta}$. We can also say we *applied* the derivative operator $\frac{d}{dx}[\cdot]$ to the function $f(x)$ to obtain the derivative function $f'(x)$. I use the word "operator" here to refer to an operation that acts on functions.

There is no reason *a priori* to think that integration and differentiation might be related: the former is a calculation about areas, while the latter is a calculation about slopes. The fundamental theorem of calculus reveals that they are in fact inverse operations: we can obtain the original function $f(x)$ from the integral function $F_a(x)$ by computing it's derivative:

$$\frac{d}{dx}\big[F_a(x)\big] = \frac{d}{dx}\left[\int_a^x f(u)\ du\right] = f(x).$$

Note we used a temporary variable u as the integration variable, since x is already used to denote the upper limit 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 6 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 obtain the same function:

$$\texttt{diff(}\ \texttt{integrate(}f(x)\texttt{)}\ \texttt{)} = f(x),$$

where we've used the SymPy functions `integrate` for computing the integrals, and `diff` (short for "differentiate") for computing derivatives.

The code example below shows how we can construct a complicated-looking function `f` and compute its integral function `F` using SymPy.

```
>>> 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 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).

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

That's kind of cool, no?

In probability theory, the FTC tells us that the probability density can be obtained from the cumulative distribution using differentiation

$$f_X(x) = \frac{d}{dx}[F_X(x)] = \frac{dF_X}{dx}(x) = F_X'(x).$$

The fact that we can obtain f_X from F_X and vice versa, means we only need to define one of the two functions, and obtain the other function using differentiation or integration. In this book, we define the random variable X through its probability distribution function f_X, then define F_X as the integral of f_X. In other books, you might see the random variable X being defined through its cumulative distribution function F_X, with its probability density function f_X defined as the derivative of F_X.

F. Fundamental theorem of calculus 2

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 $x \in (a, b)$.

The fundamental theorem of calculus establishes an equivalence between the set of integral functions and the set of antiderivative functions:

$$A_\alpha(x) = F(x) + C.$$

All integral functions $A_\alpha(x)$ are antiderivatives of $f(x)$.

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.

Integration and differentiation as inverse operations: You previously studied the inverse relationship for functions. Recall that for any *bijective* function f (a one-to-one relationship) there exists an *inverse function* f^{-1} that *undoes* the effects of f:

$$(f^{-1} \circ f)(x) \equiv f^{-1}(f(x)) = 1x$$

and also

$$(f \circ f^{-1})(y) \equiv f(f^{-1}(y)) = 1y.$$

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:

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

Note we need a new variable u inside the integral since x is already used to denote the upper limit of integration.

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

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

TODO: power sentence here to summarize FTC

Computing integrals using the fundamental theorem of calculus: 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 10) 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 $F(x) = \frac{1}{3}x^3 + C$.

G. Techniques of integration

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

$$\begin{aligned}
\int_{x=a}^{x=b}\frac{1}{x-\sqrt{x}}\,dx &= \int_{x=a}^{x=b}\frac{1}{x-\sqrt{x}}\;\frac{1}{2\sqrt{x}}du\\
&= \int_{x=a}^{x=b}\frac{1}{u^2-u}\;2u\,du\\
&= \int_{u(a)}^{u(b)}\frac{1}{u^2-u}\;2u\,du = \int_{u(a)}^{u(b)}\frac{2u}{u^2-u}\;du\\
&= \int_{u(a)}^{u(b)}\frac{2}{u-1}\;du = 2\ln(u-1)\Big|_{u(a)}^{u(b)}\\
&= 2\ln(\sqrt{x}-1)\Big|_{x=a}^{x=b} = 2\ln(\sqrt{b}-1) - 2\ln(\sqrt{a}-1).
\end{aligned}$$

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 $f(x)g'(x)$.

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

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

$$\begin{aligned}
\int_0^5 x\; e^x dx &= (xe^x)\Big|_0^5 \; - \; \int_0^5 e^x\; dx\\
&= (xe^x)\Big|_0^5 \; - \; e^x\Big|_0^5\\
&= \left[5e^5 - 0e^0\right] \; - \; \left[e^5 - e^0\right]\\
&= 5e^5 - e^5 + 1\\
&= 4e^5 + 1.
\end{aligned}$$

H. Computing integrals numerically

There are numerous ways to compute integrals using Python. Computing integrals "numerically" means we're splitting the region of integration into thousands or millions of subregions, computing the areas of these subregions, then adding up the areas of the subregions to obtain the total area.

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.

```
>>> from scipy.integrate import quad
```

Now let's define a Python function `f` that corresponds to the constant function $f(x)=3$.

```
>>> def f(x):
        return 3
>>> f(333)
3
```

No matter what input x we choose, the output will always be the same $f(x)=3$.

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.

```
>>> 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 $b = 5$.

We'll use the function `quad` hundreds of times in the remainder of the book to compute various integrals as part of probability and statistics calculations, so make sure you understand what is going on in the above code examples. 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.

I. Applications of integration

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.

VIII. Sequences and series

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 k^{th} term in the sequence or by listing all the values of the sequence:

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

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 $a(k)$.

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

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

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_n 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:

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

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

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

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

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 18.

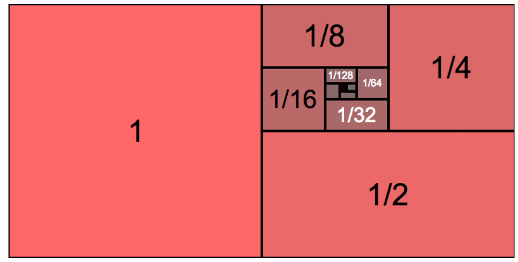


Fig. 18. A graphical representation of the infinite sum of the geometric series with $r = \frac{1}{2}$. The area of each region corresponds to one of the terms in the series. The total area is equal to $\sum_{k=0}^{\infty}(\frac{1}{2})^k = \frac{1}{1-\frac{1}{2}} = 2$.

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:

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

A. 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 k^{th} derivative of $f(x)$ and $k!$ is the factorial function:

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

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 k^{th} coefficient in the power series of $f(x) = e^x$ at the point $x = 0$ is equal to the value of the k^{th} 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 k^{th} 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.

IX. 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. 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

$$\frac{\partial}{\partial x} f(x, y) \equiv \frac{\partial f}{\partial x} \equiv \lim_{\delta \to 0} \frac{f(x + \delta, y) - f(x, y)}{\delta}.$$

Similarly the partial derivative of with respect to y is

$$\frac{\partial}{\partial y} f(x, y) \equiv \frac{\partial f}{\partial y} \equiv \lim_{\delta \to 0} \frac{f(x, y + \delta) - f(x, y)}{\delta}.$$

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

B. Gradient

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

$$\nabla f(x, y) \equiv \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right).$$

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).

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

We identify the z coordinate with the hight of the mountain $z = h(x, y)$ and graph the function $h(x, y)$ is as a surface in 3D as illustrated in Figure 19.

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

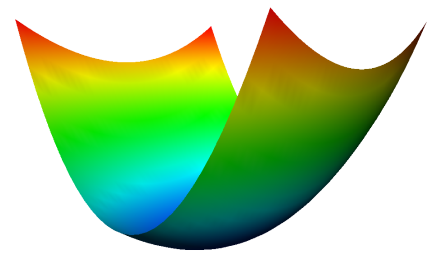


Fig. 19. The 3D surface plot of the the function $h(x, y)$.

Another approach for representing function of the form $h(x, y)$ is to use a two-dimensional plot that shows the "view from above" of the surface $h(x, y)$. We can use colour to represent the height of the function through different sharing: darker-shading to represent large values of $h(x, y)$ and lighter-shading to represent small values of $h(x, y)$. 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.

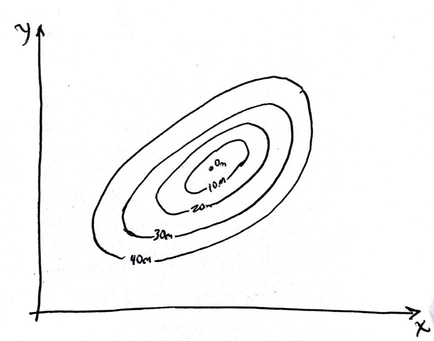


Fig. 20. The topographic map that shows the height of the function $h(x, y)$ using sharing to represent height. The level curves at each 10m intervals are also shown.

The curve labeled 30m line you see in Figure 20 represents the solution to the equation $30 = h(x, y)$, where $h(x, y)$ is the height of this hill for all coordinates (x, y) on the map.

Recall that the gradient vector $\nabla f(x, y)$ at any point (x, y) tells you which way is "uphill" on the surface, and by extension, the negative of the gradient vector points "downhill." The gradient vector is always perpendicular to the *level curve* at that point.

The notion of an uphill or downhill direction for the surface $h(x, y)$ turns out to be very useful for optimization. If you want to find the local maximum of a function, you can start at some point and keep moving uphill (in the direction of $\nabla f(x, y)$ and you'll arrive at a local peak of the mountain. Similarly, to find the lowest point on the surface (minimum value of $h(x, y)$), you can start at some point and keep moving in the opposite direction to the gradient $-\nabla f(x, y)$.

Figure 21 illustrates this process. Consider the path of a water stream whose source in some arbitrary point on the surface of the mountain. The water stream will naturally move downhill and descend 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.

We know we've reached the bottom of the valley, since the the

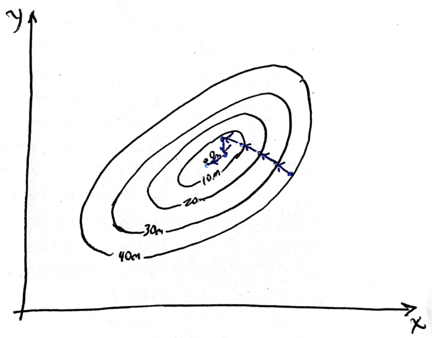


Fig. 21. This graph shows the path taken by a hypothetical water as it flows to the bottom of the valley $h(x, y)$.

gradient vector will be zero at the minimum of the function $h(x, y)$, since surface is locally flat there.

C. *Multivariable 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:

$$\int\int_{(x,y)\in R} f(x, y)\, dxdy,$$

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

X. Calculus using SymPy

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.

A. Infinity

```
from sympy import oo
```

The infinity symbol is denoted `oo` (two lowercase `o`s) in SymPy. Infinity is not a number but a process: the process of counting forever. Thus, $\infty + 1 = \infty$, ∞ is greater than any finite number, and $1/\infty$ is an infinitely small number. Sympy knows how to correctly treat infinity in expressions:

```
>>> oo+1
oo
>>> 5000 < oo
True
>>> 1/oo
0
```

B. Limits

```
from sympy import limit
```

We use limits to describe, with mathematical precision, infinitely large quantities, infinitely small quantities, and procedures with infinitely many steps.

The number e is defined as the limit $e \stackrel{\text{def}}{=} \lim_{n\to\infty}\left(1+\frac{1}{n}\right)^n$:

```
>>> limit( (1+1/n)**n, n, oo)
E          # = 2.71828182845905
```

This limit expression describes the annual growth rate of a loan with a nominal interest rate of 100% and infinitely frequent compounding. Borrow \$1000 in such a scheme, and you'll owe \$2718.28 after one year.

Limits are also useful to describe the behaviour of functions. Consider the function $f(x) = \frac{1}{x}$. The `limit` command shows us what happens to $f(x)$ near $x = 0$ and as x goes to infinity:

```
>>> limit( 1/x, x, 0, dir="+")
oo
>>> limit( 1/x, x, 0, dir="-")
-oo
>>> limit( 1/x, x, oo)
0
```

As x becomes larger and larger, the fraction $\frac{1}{x}$ becomes smaller and smaller. In the limit where x goes to infinity, $\frac{1}{x}$ approaches zero: $\lim_{x\to\infty}\frac{1}{x} = 0$. On the other hand, when x takes on smaller and smaller positive values, the expression $\frac{1}{x}$ becomes infinite: $\lim_{x\to 0^+}\frac{1}{x} = \infty$. When x approaches 0 from the left, we have $\lim_{x\to 0^-}\frac{1}{x} = -\infty$. If these calculations are not clear to you, study the graph of $f(x) = \frac{1}{x}$.

Here are some other examples of limits:

```
>>> limit(sin(x)/x, x, 0)
1
>>> limit(sin(x)**2/x, x, 0)
0
>>> limit(exp(x)/x**100,x,oo) # which is bigger e^x or x^100 ?
oo                           # exp f >> all poly f for big x
```

Limits are used to define the derivative and the integral operations.

C. Derivatives

The derivative function, denoted $f'(x)$, $\frac{d}{dx}f(x)$, $\frac{df}{dx}$, or $\frac{dy}{dx}$, describes the *rate of change* of the function $f(x)$. The SymPy function `diff` computes the derivative of any expression:

```
>>> diff(x**3, x)
3*x**2
```

The differentiation operation knows the product rule $[f(x)g(x)]' = f'(x)g(x) + f(x)g'(x)$, the chain rule $f(g(x))' = f'(g(x))g'(x)$, and the quotient rule $\left[\frac{f(x)}{g(x)}\right]' = \frac{f'(x)g(x)-f(x)g'(x)}{g(x)^2}$:

```
>>> diff( x**2*sin(x), x )
2*x*sin(x) + x**2*cos(x)
>>> diff( sin(x**2), x )
cos(x**2)*2*x
>>> diff( x**2/sin(x), x )
(2*x*sin(x) - x**2*cos(x))/sin(x)**2
```

The second derivative of a function `f` is `diff(f,x,2)`:

```
>>> diff(x**3, x, 2)      # same as diff(diff(x**3, x), x)
6*x
```

D. 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

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

What is the equation of the tangent line to $f(x) = \frac{1}{2}x^2$ at $x_0 = 1$?

```
>>> f = S('1/2')*x**2
>>> f
x**2/2
>>> df = diff(f, x)
>>> df
x
>>> T_1 = f.subs({x:1}) + df.subs({x:1})*(x - 1)
>>> T_1
x - 1/2          #  y = x - 1/2
```

The tangent line $T_1(x)$ has the same value and slope as the function $f(x)$ at $x = 1$:

```
>>> T_1.subs({x:1}) == f.subs({x:1})
True
>>> diff(T_1, x).subs({x:1}) == diff(f, x).subs({x:1})
True
```

E. Optimization

Optimization is about choosing an input for a function $f(x)$ that results in the best value for $f(x)$. The best value usually means the *maximum* value (if the function represents something desirable like profits) or the *minimum* value (if the function represents something undesirable like costs).

The derivative $f'(x)$ encodes the information about the *slope* of $f(x)$. Positive slope $f'(x) > 0$ means $f(x)$ is increasing, negative slope $f'(x) < 0$ means $f(x)$ is decreasing, and zero slope $f'(x) = 0$ means the graph of the function is horizontal. The *critical points* of a function $f(x)$ are the solutions to the equation $f'(x) = 0$. Each critical point is a candidate to be either a maximum or a minimum of the function.

The second derivative $f''(x)$ encodes the information about the *curvature* of $f(x)$. Positive curvature means the function looks like x^2, negative curvature means the function looks like $-x^2$.

Let's find the critical points of the function $f(x) = x^3 - 2x^2 + x$ and use the information from its second derivative to find the maximum of the function on the interval $x \in [0, 1]$.

```
>>> x = Symbol('x')
>>> f = x**3-2*x**2+x
>>> diff(f, x)
3*x**2 - 4*x + 1
>>> sols = solve( diff(f,x),  x)
>>> sols
[1/3, 1]
>>> diff(diff(f,x), x).subs( {x:sols[0]} )
-2
>>> diff(diff(f,x), x).subs( {x:sols[1]} )
2
```

It will help to look at the graph of this function. The point $x = \frac{1}{3}$ is a local maximum because it is a critical point of $f(x)$ where the curvature is negative, meaning $f(x)$ looks like the peak of a mountain at $x = \frac{1}{3}$. The maximum value of $f(x)$ on the interval $x \in [0, 1]$ is $f\left(\frac{1}{3}\right) = \frac{4}{27}$. The point $x = 1$ is a local minimum because it is a critical point with positive curvature, meaning $f(x)$ looks like the bottom of a valley at $x = 1$.

F. Integrals

The *integral* of $f(x)$ corresponds to the computation of the area under the graph of $f(x)$. The area under $f(x)$ between the points $x = a$ and $x = b$ is denoted as follows:

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

The *integral function* F corresponds to the area calculation as a function of the upper limit of integration:

$$F(c) \stackrel{\text{def}}{=} \int_0^c f(x)\, dx\,.$$

The area under $f(x)$ between $x = a$ and $x = b$ is obtained by calculating the *change* in the integral function:

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

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
```

Integrals correspond to *signed* area calculations:

```
>>> integrate(sin(x), (x,0,pi))
2
>>> integrate(sin(x), (x,pi,2*pi))
-2
>>> integrate(sin(x), (x,0,2*pi))
0
```

During the first half of its 2π-cycle, the graph of $\sin(x)$ is above the x-axis, so it has a positive contribution to the area under the curve. During the second half of its cycle (from $x = \pi$ to $x = 2\pi$), $\sin(x)$ is below the x-axis, so it contributes negative area. Draw a graph of $\sin(x)$ to see what is going on.

G. Fundamental theorem of calculus

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 fundamental theorem of calculus is important because it tells us how 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 $f(x) = \int g(x)\, dx + C$.

H. Sequences

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

We define a sequence by specifying an expression for its n^{th} term:

```
>>> a_n = 1/n
>>> b_n = 1/factorial(n)
```

Substitute the desired value of n to see the value of the n^{th} term:

```
>>> a_n.subs({n: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_n.subs({n:i}) for i in range(0,8) ]
[oo, 1, 1/2, 1/3, 1/4,  1/5,   1/6,   1/7]
>>> [ b_n.subs({n:i}) for i in range(0,8) ]
[1,  1, 1/2, 1/6, 1/24, 1/120, 1/720, 1/5040]
```

Observe that a_n is not properly defined for $n = 0$ since $\frac{1}{0}$ is a division-by-zero error. To be precise, we should say a_n's domain is the positive naturals $a_n : \mathbb{N}^+ \to \mathbb{R}$. Observe how quickly the `factorial` function $n! = 1 \cdot 2 \cdot 3 \cdots (n-1) \cdot n$ grows: $7! = 5040$, $10! = 3628800$, $20! > 10^{18}$.

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

```
>>> limit(a_n, n, oo)
0
>>> limit(b_n, n, oo)
0
```

Both $a_n = \frac{1}{n}$ and $b_n = \frac{1}{n!}$ *converge* to 0 as $n \to \infty$.

Many important math quantities are defined as limit expressions. An interesting example to consider is the number π, which is defined as the area of a circle of radius 1. We can approximate the area of the unit circle by drawing a many-sided regular polygon around the circle. Splitting the n-sided regular polygon into identical triangular splices, we can obtain a formula for its area A_n. In the limit as $n \to \infty$, the n-sided-polygon approximation to the area of the unit-circle becomes exact:

```
>>> A_n = n*tan(2*pi/(2*n))
>>> limit(A_n, n, oo)
pi
```

I. Series

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

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

```
>>> a_n = 1/n
>>> b_n = 1/factorial(n)
>>> summation(a_n, [n, 1, oo])
oo
>>> summation(b_n, [n, 0, oo])
E
```

We say the series $\sum a_n$ *diverges* to infinity (or *is divergent*) while the series $\sum b_n$ converges (or *is convergent*). As we sum together more and more terms of the sequence b_n, the total becomes closer and closer to some finite number. In this case, the infinite sum $\sum_{n=0}^\infty \frac{1}{n!}$ 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 that is accurate to six decimals by summing 10 terms in the series:

```
>>> import math
>>> def b_nf(n):
        return 1.0/math.factorial(n)
>>> sum( [b_nf(n) for n in range(0,10)] )
2.718281 52557319
>>> E.evalf()
2.718281 82845905       # true value
```

J. Taylor series

Wait, there's more! Not only can we use series to approximate numbers, we can also use them to approximate functions.

A *power series* is a series whose terms contain different powers of the variable x. The n^{th} term in a power series is a function of both the sequence index n and the input variable x.

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

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

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_xn = x**n/factorial(n)
>>> summation( exp_xn.subs({x:5}), [n, 0, oo] ).evalf()
148.413159102577
>>> exp(5).evalf()
148.413159102577        # the true value
```

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

```
>>> summation( exp_xn.subs({x:5}), [n, 0, oo])
exp(5)
```

Taking as few as 35 terms in the series is sufficient to obtain an approximation to e that is accurate to 16 decimals:

```
>>> import math                       # redo using only python
>>> def exp_xnf(x,n):
        return x**n/math.factorial(n)
>>> sum( [exp_xnf(5.0,i) for i in range(0,35)] )
148.413159102577
```

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 n^{th} term in the Taylor series of $f(x)$ expanded at $x = c$ is $a_n(x) = \frac{f^{(n)}(c)}{n!}(x-c)^n$, where $f^{(n)}(c)$ is the value of the n^{th} derivative of $f(x)$ evaluated at $x = c$. The term *Maclaurin series* refers to Taylor series expansions at $x = 0$.

The SymPy function `series` is a convenient way to obtain the 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`:

```
>>> series( sin(x), x, 0, 8)
x - x**3/6 + x**5/120 - x**7/5040 + O(x**8)
>>> series( cos(x), x, 0, 8)
1 - x**2/2 + x**4/24 - x**6/720 + O(x**8)
>>> series( sinh(x), x, 0, 8)
x + x**3/6 + x**5/120 + x**7/5040 + O(x**8)
>>> series( cosh(x), x, 0, 8)
1 + x**2/2 + x**4/24 + x**6/720 + O(x**8)
```

Some functions are not defined at $x = 0$, so we expand them at a different value of x. For example, the power series of $\ln(x)$ expanded at $x = 1$ is

```
>>> series(ln(x), x, 1, 6)     # Taylor series of ln(x) at x=1
x - x**2/2 + x**3/3 - x**4/4 + x**5/5  + O(x**6)
```

Here, the result SymPy returns is misleading. The Taylor series of $\ln(x)$ expanded at $x = 1$ has terms of the form $(x-1)^n$:

$$\ln(x) = (x-1) - \frac{(x-1)^2}{2} + \frac{(x-1)^3}{3} - \frac{(x-1)^4}{4} + \frac{(x-1)^5}{5} + \cdots .$$

Verify this is the correct formula by substituting $x = 1$. SymPy returns an answer in terms of coordinates *relative* to $x = 1$.

Instead of expanding $\ln(x)$ around $x = 1$, we can obtain an equivalent expression if we expand $\ln(x+1)$ around $x = 0$:

```
>>> series(ln(x+1), x, 0, 6)   # Maclaurin series of ln(x+1)
x - x**2/2 + x**3/3 - x**4/4 + x**5/5 + O(x**6)
```

Exercises

TODO: 10x exercises on sets, functions, and integrals (geometric, numeric, and symbolic)

TODO: 3 more

Links

K. Recommended calculus learning resources

Above all, my advice is not to think of calculus as "advanced math theory" that might be difficult to understand, but instead as practical, useful math that allows you to do calculations—just look at the name of the thing! 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. The problems and exercises in the books *Calculus made simple* and *No bullshit guide to math and physics* are therefore your best route for learning calculus, if you choose to pursue this subject.

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

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