Fundamentals of Integral

18 积分

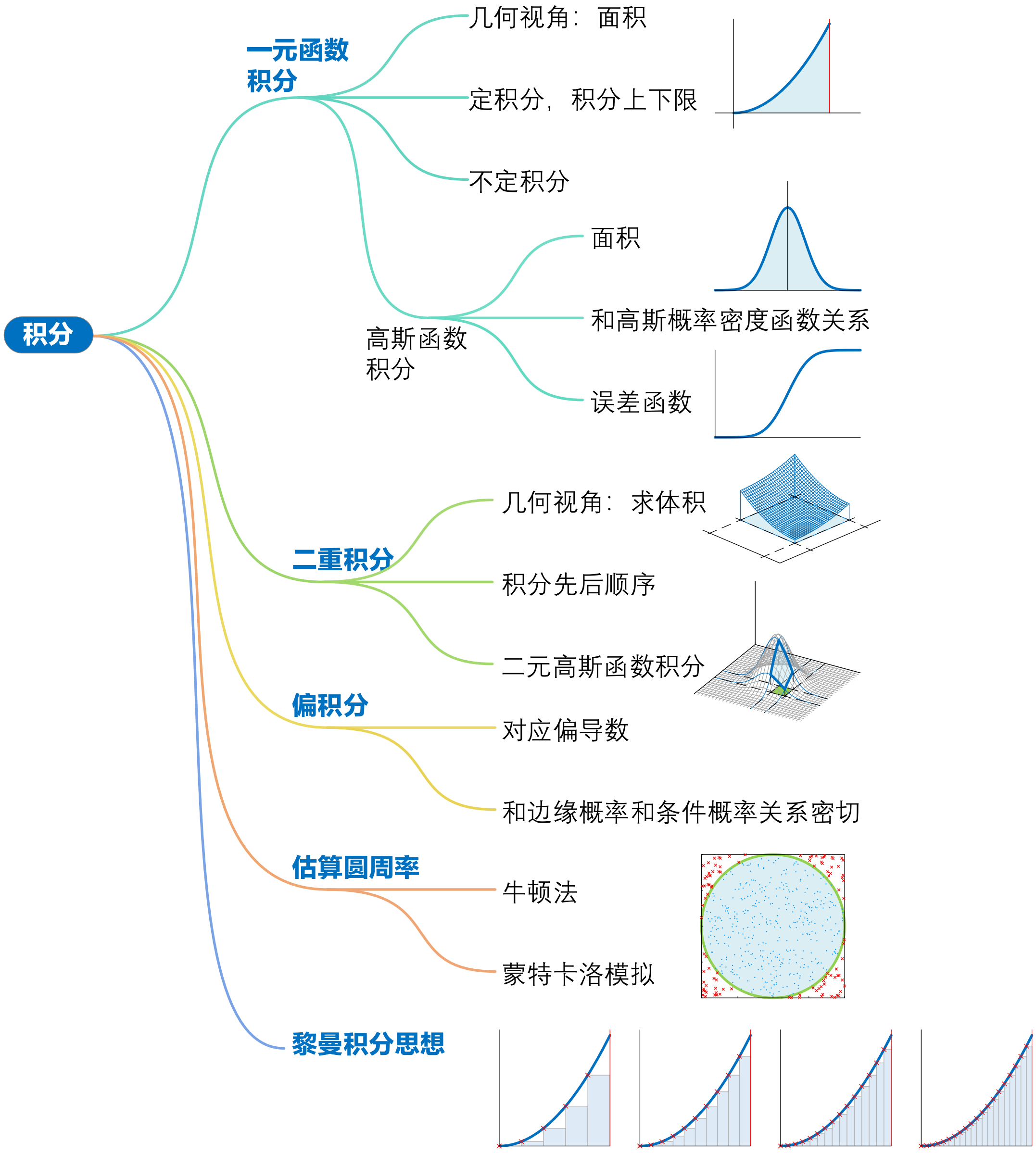
源自于求面积、体积等数学问题

有苦才有甜。

He who hasn't tasted bitter things hasn't earned sweet things.

——戈特弗里德·莱布尼茨 (Gottfried Wilhelm Leibniz) | 德意志数学家、哲学家 | 1646 ~ 1716

- ◀ `numpy.vectorize()` 将自定义函数向量化
- ◀ `sympy.abc import x` 定义符号变量 x
- ◀ `sympy.diff()` 求解符号导数和偏导解析式
- ◀ `sympy.Eq()` 定义符号等式
- ◀ `sympy.evalf()` 将符号解析式中未知量替换为具体数值
- ◀ `sympy.integrate()` 符号积分
- ◀ `sympy.symbols()` 定义符号变量


积分
一元函数积分
几何视角：面积
定积分，积分上下限
不定积分
高斯函数积分
面积
和高斯概率密度函数关系
误差函数
二重积分
几何视角：求体积
积分先后顺序
二元高斯函数积分
偏积分
对应偏导数
和边缘概率和条件概率关系密切
估算圆周率
牛顿法
蒙特卡洛模拟
黎曼积分思想

18.1 莱布尼茨：既生瑜，何生亮

实际上，人类对积分的探索要远早于微分。古时候，各个文明都在探索不同方法计算不规则形状的长度、面积、体积，人类几何知识则在这个过程中不断进步并且体系化。前文介绍过早期数学家估算圆周率时用内接或外切正多边形近似正圆，其中蕴含的数学思想也是积分的基础。

积分的本来含义就是求和，拉丁语 summa 首字母 s 纵向拉伸，便得到积分符号 $\int$。积分符号 $\int$ 的发明者便是莱布尼茨 (Gottfried Wilhelm Leibniz)。

戈特弗里德·威廉·莱布尼茨 (Gottfried Wilhelm Leibniz)
德国哲学家、数学家 | 1646年 ~ 1716年
和牛顿先后独立发明了微积分，创造的微积分符号至今被广泛使用

莱布尼兹是十七世纪少有的通才，这个德国人是律师、哲学家、工程师，更是优秀的数学家。

牛顿和莱布尼兹各自独立发明微积分，两者就微积分发明权争执了很长时间。牛顿在十七世纪的学术界呼风唤雨，是学术天空中最耀眼的一颗星辰，莱布尼兹和其他学者的光芒则显得暗淡很多。很可能是因为这个原因，英国皇家学会公开判定“牛顿是微积分的第一发明人”。

但是，莱布尼兹显得大度很多，他公开表示“在从世界开始到牛顿生活的时代的全部数学中，牛顿的工作超过了一半。”

不管谁发明了微积分，莱布尼兹的微积分数学符号被后世广泛采用，这也算是一种胜利。

18.2 从小车匀加速直线运动说起

回顾本书第 15 章讲解导数时给出的匀加速直线运动的例子。

如图 1 所示，匀加速直线运动中，加速度 $a(t)$ 是常数函数，图像是水平线 $a(t) = 1$ (忽略单位)。时间 $0 \sim t$，水平线和横轴围成的面积是个矩形。容易求解矩形面积，这个面积对应速度函数 $v(t) = t$。

显然，$v(t)$ 是个一次函数，图像为一条通过原点的斜线。时间范围为 $0 \sim t$，$v(t)$ 斜线和横轴围成的面积是个三角形。三角形的面积对应距离函数 $s(t) = t^2/2$。而 $s(t)$ 是个二次函数，图像为抛物线。

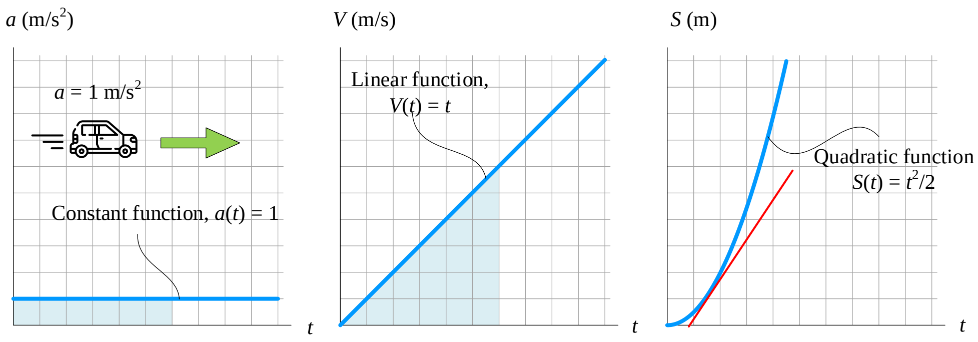


图 1. 匀加速直线运动：加速度、速度、距离

求解矩形面积和三角形面积显然难不倒我们。但是，当我们把问题的难度稍微提高。比如，将变量从 t 换成 x，把距离函数写成 $f(x) = x^2/2$。

如图 2 所示，x 在一定区间内，二元函数 $f(x)$ 曲线和横轴构成这块形状不规则图形，要精确计算它的面积，怎么办呢？

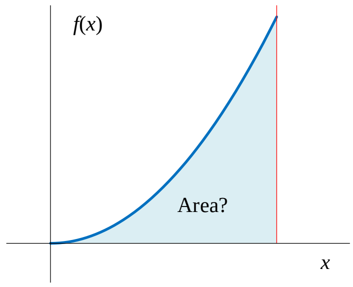


图 2. $f(x)$ 在固定区间积分求面积

还有，如何计算图 3 中 $f(x, y)$ 曲面在 D 区域内和水平面围成的几何形体体积？

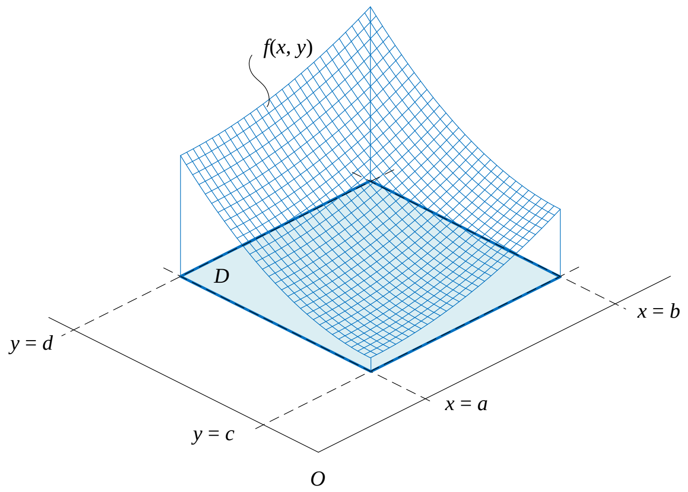


图 3. $f(x, y)$ 在区域 D 进行二重积分求体积

解决这些问题需要借助本章要讲解的重要数学工具——积分。

18.3 一元函数积分

导数、偏导、积分、二重积分

本书第 15 章聊过，导数关注变化率。对于一元函数，图 4 所示，几何角度来看，导数相当于是曲线切线斜率。而对于二元函数来说，偏导数是二元函数曲面某点在特定方向的切线斜率。微分，则是线性近似。

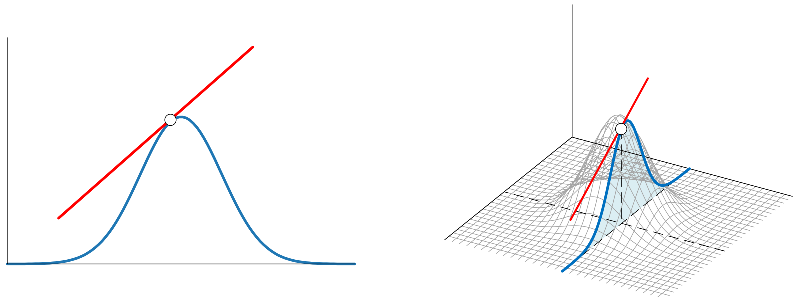


图 4. 几何视角看导数、偏导数、微分等数学工具

积分是微分的逆运算，积分关注变化累积，比如曲线面积、曲面体积，如图 5 所示。导数、微分、积分这个数学工具合称微积分，微积分是定量研究变化过程的重要数学工具。

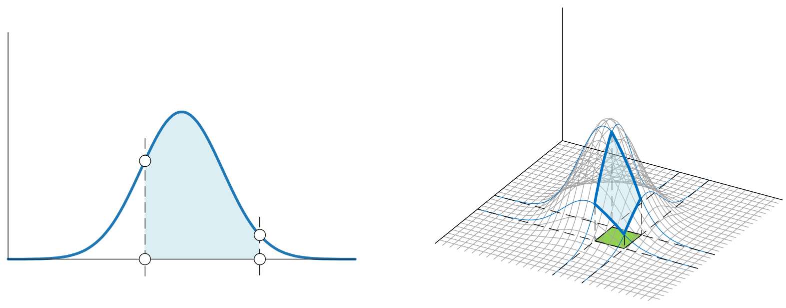


图 5. 几何视角看积分、多重积分等数学工具

一元函数积分

一元函数 $f(x)$ 自变量 x 在区间 $[a, b]$ 上定积分运算记做：

$$\int_a^b f(x)\,\mathrm{d}x \tag{1}$$

其中，a 叫积分下限 (lower bound)，b 叫积分上限 (upper bound)。

⚠ 注意，积分有正负之分，也就是说面积值有正负。

如图 6 所示，对于一元函数，曲线在横轴之上包围的面积为正，曲线在横轴之下包围的面积为负。

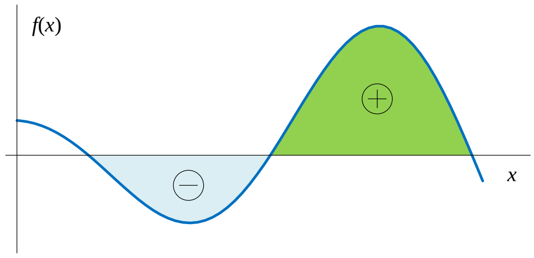


图 6. 积分有正负之分

类比的话，一次函数积分类似如下累加：

$$\sum_{i=1}^{n} a_i = a_1 + a_2 + \cdots + a_{n-1} + a_n \tag{2}$$

举个例子

图 7 (a) 所示为如下一次函数的定积分：

$$\int_0^1 \left(x^2 + \frac{1}{2}\right) \mathrm{d}\,x = \left(\frac{1}{3}x^3 + \frac{1}{2}x\right)\Bigg|_0^1 = \frac{5}{6} \approx 0.8333 \tag{3}$$

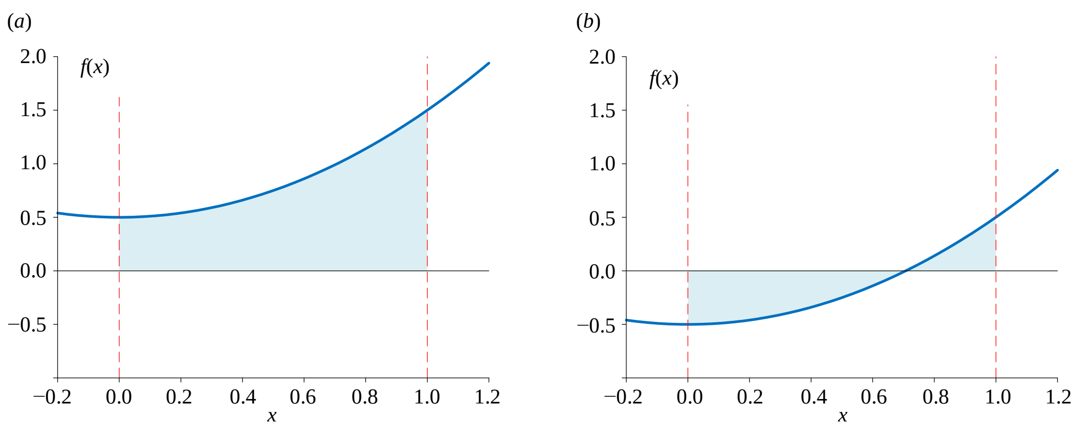


图 7. 两个函数的定积分

图 7 (b) 所示为如下函数的定积分：

$$\int_0^1 \left(x^2 - \frac{1}{2} \right) \mathrm{d}\, x = \left(\frac{1}{3} x^3 - \frac{1}{2} x \right) \Bigg|_0^1 = -\frac{1}{6} \approx -0.1667 \tag{4}$$

图 7 (a) 函数图像向下移动一个单位便得到图 7 (b)。因此，(3) 和 (4) 两个定积分存在以下关系：

$$\int_0^1 \left(x^2 + \frac{1}{2} \right) \mathrm{d}\, x - \int_0^1 \left(x^2 - \frac{1}{2} \right) \mathrm{d}\, x = 1 \tag{5}$$

若定积分存在，定积分则是一个具体的数值；而不定积分结果一般是一个函数表达式。

表 1. 积分的英文表达

数学表达	英文表达
$\int_1^3 x^3 \mathrm{d}\, x$	The integral from one to three of *x* cubed d *x*
$\int f(x) \mathrm{d}\, x$	The integral *f* of *x* d *x* The indefinite integral of *f* with respect to *x*
$\int_a^b f(x) \mathrm{d}\, x$	The integral from *a* to *b* of *f* of *x* d *x*

Bk3_Ch18_01.py 计算定积分并绘制图 7 两幅子图。

18.4 高斯函数积分

高斯函数积分有自己的名字——高斯积分 (Gaussian integral)。

前文提过，高斯函数和高斯分布 (Gaussian distribution) 联系紧密；因此，高斯积分在概率统计中也有扮演重要角色。

坐标变换方法可以求解高斯积分；但是，本书不会介绍如何推导高斯积分，这部分内容留给感兴趣的读者自己探索。

本章想从几何视角和大家聊聊有关高斯积分的一些重要性质，这部分内容和本系列丛书后续内容高斯分布有着密切联系。

对于一元高斯函数积分，请大家首先留意如下积分结果：

$$\int_{-\infty}^{\infty} \exp\left(-x^2\right) \mathrm{d}\, x = \sqrt{\pi} \tag{6}$$

如图 8 所示，高斯函数 $f(x) = \exp(-x^2)$ 和整个 x 轴围成的面积为 $\sqrt{\pi}$ 。再次强调，图 8 中高斯函数趋向正、负无穷时，函数值无限接近 0，但是达不到 0。

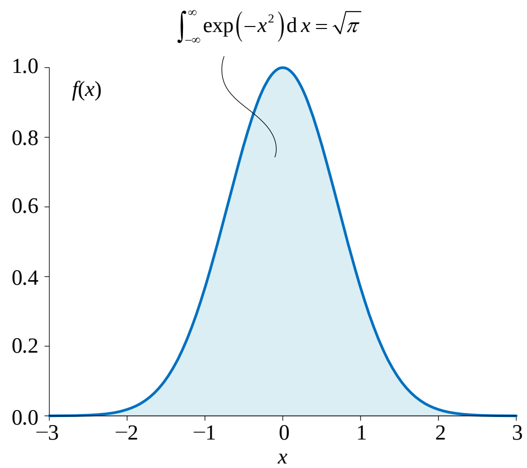


图 8. 高斯函数正负无穷积分面积

定积分

再举个定积分的例子，给定如下积分上下限，计算高斯函数定积分：

$$\int_{-0.5}^{1} \exp\left(-x^2\right) \mathrm{d}\, x \approx 1.208 \tag{7}$$

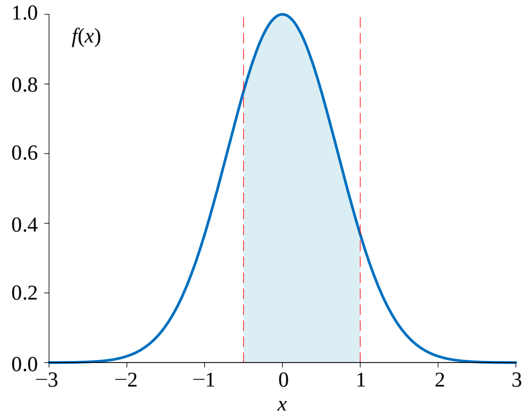


图 9. 高斯函数定积分

前文提过不定积分存在的话，函数的不定积分结果是函数。比如，对高斯函数从−∞积分到 x 得到：

$$F(x)=\int_{-\infty}^{x}\exp\left(-t^2\right)\mathrm{d}t \tag{8}$$

图 10 中蓝色曲线所示为上述高斯积分 $F(x)$ 随 x 变化。

⚠ 注意，高斯函数积分没有解析解。

这样 (7) 可以用 $F(x)$ 计算如下定积分：

$$\int_{-0.5}^{1}\exp\left(-x^2\right)\mathrm{d}x=F(1)-F(-0.5)\approx 1.633-0.425=1.208 \tag{9}$$

图 10 所示为利用 $F(x)$ 计算 (7) 定积分原理。

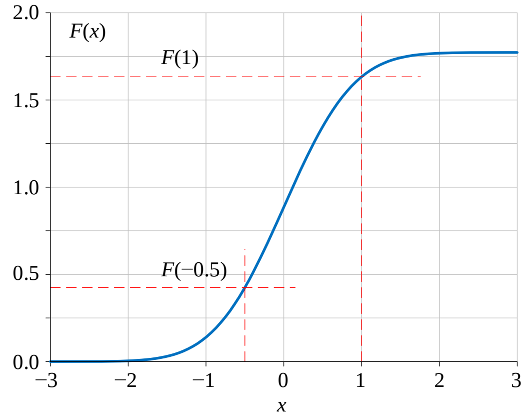


图 10. 用 $F(x)$ 计算高斯定积分

另外注意下面这个积分公式：

$$\int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{x^2}{2}\right)\mathrm{d}x=1 \tag{10}$$

看到这个公式，大家是否联想到一元标准正态分布概率密度函数 PDF。分母上为 $\sqrt{2\pi}$ 作用是归一化，也就是让函数和整个横轴围成的面积为 1。这解释了为什么高斯分布概率密度函数的分母上有 $\sqrt{2\pi}$ 这个缩放系数。

Bk3_Ch18_02.py 计算高斯函数积分并且绘制图 9 和图 10。

18.5 误差函数：S 型函数的一种

通过调取代码结果，大家可能已经发现高斯函数积分结果是用 erf() 函数来表达的。

erf(x) 函数就是鼎鼎有名的误差函数 (error function)：

$$\operatorname{erf}(x)=\frac{1}{\sqrt{\pi}}\int_{-x}^{x}\exp\left(-t^2\right)\mathrm{d}t=\frac{2}{\sqrt{\pi}}\int_{0}^{x}\exp\left(-t^2\right)\mathrm{d}t \tag{11}$$

误差函数是利用高斯积分定义的，它没有一般意义上的解析式。

前文提过，误差函数是 S 型函数的一种。误差函数在概率统计、数据科学、机器学习中应用广泛。

一般情况，误差函数自变量 x 的取值为正值，但是为了计算方便，erf() 的输入也可以是负值。x 为负值时，下式成立：

$$\operatorname{erf}(x)=-\operatorname{erf}(-x) \tag{12}$$

图 11 所示为误差函数图像。

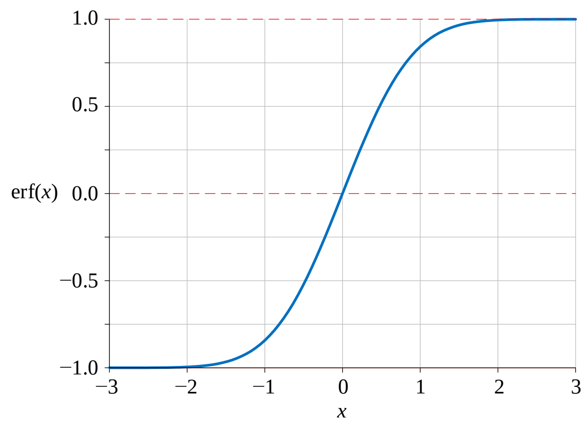


图 11. 误差函数

高斯函数从−∞积分到 x 对应的高斯积分和误差函数的关系为：

$$F(x)=\int_{-\infty}^{x}\exp\left(-t^2\right)\mathrm{d}t=\underbrace{\frac{\sqrt{\pi}}{2}}_{\text{Scale}}\operatorname{erf}(x)+\underbrace{\frac{\sqrt{\pi}}{2}}_{\text{Shift}} \tag{13}$$

通过上述公式可以看出误差函数先是通过纵轴缩放，再沿纵轴平移得到高斯积分。

Bk3_Ch18_03.py 绘制图 11。注意，sympy.erf() 可以接受负值。

18.6 二重积分：类似二重求和

先对 x 积分

给定积分区域 $D = \{(x, y) \mid a < x < b, c < y < d\}$，$f(x,y)$ 二重积分记做：

$$\int_c^d \int_a^b f(x,y)\,\mathrm{d}x\,\mathrm{d}y \tag{14}$$

请注意上式二重积分的先后次序，先对 x 积分，再对 y 积分。也就是说，内部这一层 $\int_{x=a}^{x=b} f(x,y)\,\mathrm{d}x$ 先消去 x，变成有关 y 的一元函数；然后在对 y 积分：

$$\int_c^d \int_a^b f(x,y)\,\mathrm{d}x\,\mathrm{d}y = \int_{y=c}^{y=d} \underbrace{\overbrace{\int_{x=a}^{x=b} f(x,y)\,\mathrm{d}x}^{\text{Eliminate } x}}_{\text{A function of } y}\,\mathrm{d}y \tag{15}$$

$\int_{x=a}^{x=b} f(x,y)\,\mathrm{d}x$ 相当于降维，也就是“压缩”。

从几何角度，如图 12 所示，当 $y = c$ 时，$\int_{x=a}^{x=b} f(x,y=c)\,\mathrm{d}x$ 结果为图中暖色阴影区域面积，也就是压缩为一个值。

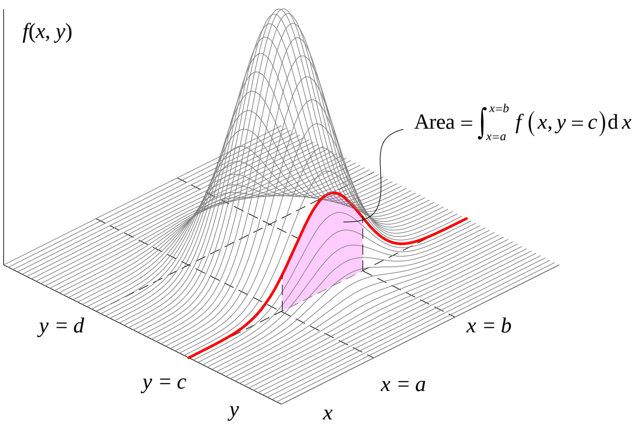


图 12. $f(x,y)$ 先对 x 积分，相当于沿 x 轴压缩

先对 y 积分

如果调换积分顺序，先对 y 积分，$\int_{y=c}^{y=d} f(x,y)\,\mathrm{d}y$ 相当于消去 y，得到有关 x 的一元函数；然后再对 x 积分：

$$\int_a^b\int_c^d f(x,y)\,\mathrm{d}y\,\mathrm{d}x = \int_{x=a}^{x=b}\overbrace{\underbrace{\int_{y=c}^{y=d} f(x,y)\,\mathrm{d}y}_{\text{A function of } x}}^{\text{Eliminate } y}\,\mathrm{d}x \tag{16}$$

如图 13 所示，当 $x = a$ 时，$\int_{y=c}^{y=d} f(x=a,y)\,\mathrm{d}y$ 结果为图中冷色阴影区域面积，即压缩为一个值。

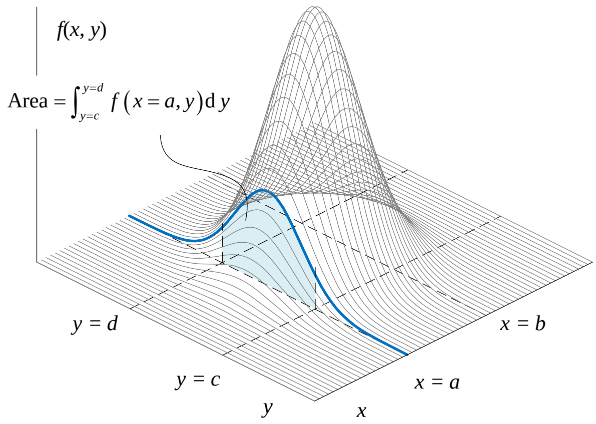


图 13. $f(x,y)$ 先对 y 积分，相当于沿 y 轴压缩

调换积分顺序

特别地，如果 $f(x, y)$ 在矩形局域 $D = \{(x, y) \mid a < x < b, c < y < d\}$ 连续，如下二重积分先后顺序可以调换：

$$\int_c^d\int_a^b f(x,y)\,\mathrm{d}x\,\mathrm{d}y = \int_a^b\int_c^d f(x,y)\,\mathrm{d}y\,\mathrm{d}x \tag{17}$$

白话说，如果积分的区域相对于坐标系“方方正正”，积分顺序可以调换。

⚠ 千万注意，二重积分、多重积分中，积分先后顺序不能随意调换，上述例子仅仅是个特例而已。有关多重积分顺序调换内容，本书不展开讲解。

类比的话，二重积分类似如下二重累加：

$$\sum_{i=1}^{n}\sum_{j=1}^{m} a_{i,j} \tag{18}$$

二元高斯函数

举个例子，给定二元高斯函数 $f(x, y)$，

$$f(x,y) = \exp\left(-x^2 - y^2\right) \tag{19}$$

$f(x, y)$ 的二重不定积分 $F(x,y)$ 可以用误差函数表达为：

$$F(x,y) = \int_{-\infty}^{y}\int_{-\infty}^{x} \exp\left(-u^2 - v^2\right) \mathrm{d}u\,\mathrm{d}v = \frac{\pi}{4}\operatorname{erf}(x)\operatorname{erf}(y) + \frac{\pi}{4}\operatorname{erf}(x) + \frac{\pi}{4}\operatorname{erf}(y) + \frac{\pi}{4} \tag{20}$$

图 14 所示为 $F(x,y)$ 曲面以及三维等高线。图 15 所示为 $F(x,y)$ 曲面在 xz 平面、yz 平面投影，可以发现投影得到的曲线形状类似误差函数。

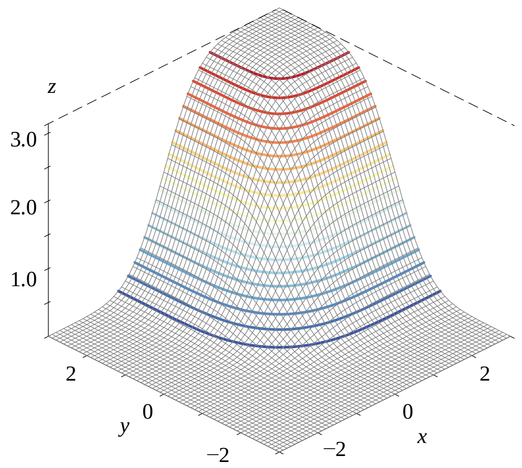


图 14. 二元高斯函数二重不定积分 $F(x,y)$ 曲面

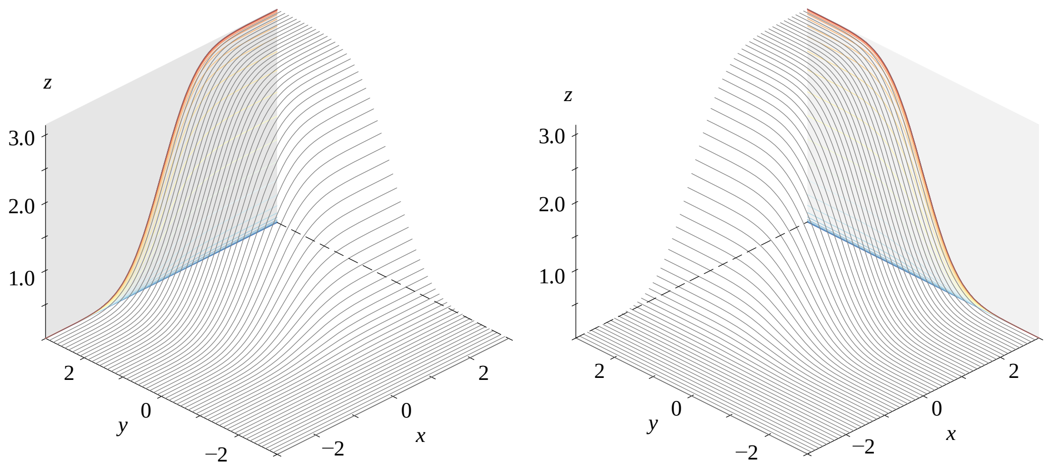


图 15. $F(x,y)$ 曲面在 xz 平面、yz 平面投影

特别地，$f(x, y)$ 曲面和整个水平面围成的体积为 π，即，

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\exp\left(-x^2-y^2\right)\mathrm{d}\,x\,\mathrm{d}\,y=\pi \tag{21}$$

Bk3_Ch18_04.py 完成本节二重积分计算。

18.7 “偏积分”：类似偏求和

本书第 14 章介绍过“偏求和”、偏导数，“偏”字的意思是是考虑一个变量，其他变量视为定值。本节自创一个积分概念——“偏积分”，如下两式就是“偏积分”：

$$\begin{gathered}\int_a^b f\left(x,y\right)\mathrm{d}\,x\\ \int_c^d f\left(x,y\right)\mathrm{d}\,y\end{gathered} \tag{22}$$

类比的话，偏积分类似前文介绍的偏求和：

$$\sum_{i=1}^{n} a_{i,j},\quad \sum_{j=1}^{m} a_{i,j} \tag{23}$$

对 y 偏积分

给定二元高斯函数 $f(x, y)$：

$$f\left(x,y\right)=\exp\left(-x^2-y^2\right) \tag{24}$$

$f(x, y)$ 对于 y 从负无穷到正无穷偏积分，得到结果变成了关于 x 的高斯函数：

$$\int_{-\infty}^{\infty}\exp\left(-x^2-y^2\right)\mathrm{d}\,y=\sqrt{\pi}\exp\left(-x^2\right) \tag{25}$$

从几何角度来看，如图 16 所示，$f(x, y)$ 对 y 从负无穷到正无穷偏积分，相当于 x 取某个值时，比如 $x = c$，对二元高斯函数 $f(x, y)$ 曲线做个剖面，剖面线 (图 16 彩色曲线) 和其水平面投影构成面积 (图 16 彩色阴影区域) 就是偏积分结果。

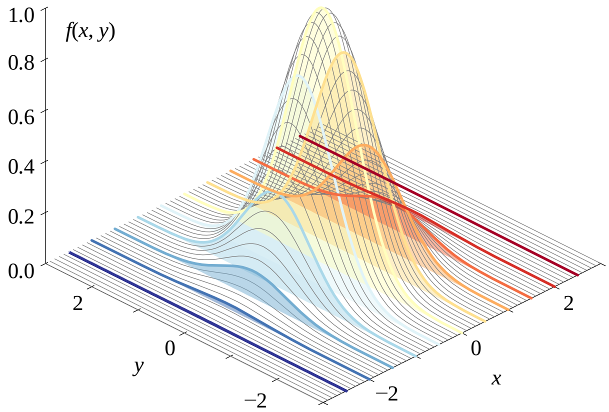


图 16. 二元高斯函数 $f(x, y)$ 对 y 偏积分

正如图 17 所示，(25) 的偏积分结果是有关 x 的一元高斯函数。

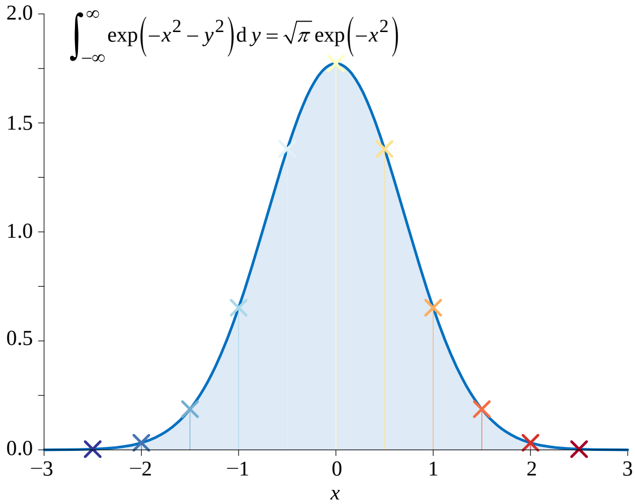


图 17. 对 y 偏积分的结果是关于 x 的高斯函数

对 x 偏积分

类似地，二元高斯函数 $f(x, y)$ 对 x 从负无穷到正无穷偏积分，结果为关于 y 的高斯函数：

$$\int_{-\infty}^{\infty} \exp\left(-x^2 - y^2\right) \mathrm{d}\, x = \sqrt{\pi} \exp\left(-y^2\right) \tag{26}$$

图 18 为上式的几何含义。

再次注意，偏积分是我们创造的一个词，对应偏导数。预告一下，“偏积分”这个概念在概率统计中会帮助我们理解连续随机变量的边缘概率和条件概率。

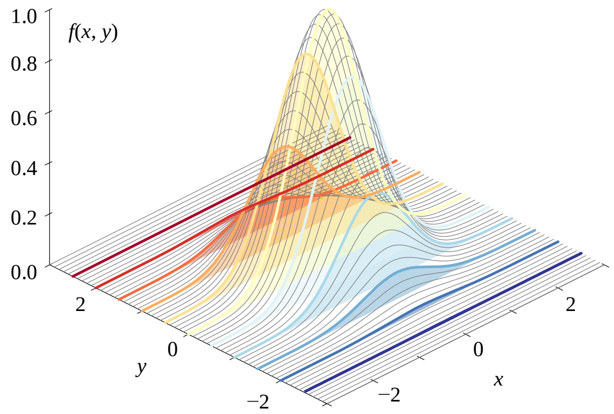


图 18. 二元高斯函数 $f(x, y)$ 对 x 偏积分

Bk3_Ch18_05.py 计算高斯二元函数偏积分。

18.8 估算圆周率：牛顿法

本书前文介绍过估算圆周率的不同方法。随着数学工具的不断升级，有了微积分这个强有力的工具，我们可以介绍牛顿估算圆周率的方法。

图 19 中给出的函数 $f(x)$ 是某个圆形的上半圆。这个圆的中心位于 (0.5, 0)，半径为 0.5。上半圆函数 $f(x)$ 的解析式为：

$$f(x) = \sqrt{x - x^2} \tag{27}$$

在这个半圆中，划定图 19 左图所示的阴影区域，它对应的圆心角度为 60°。

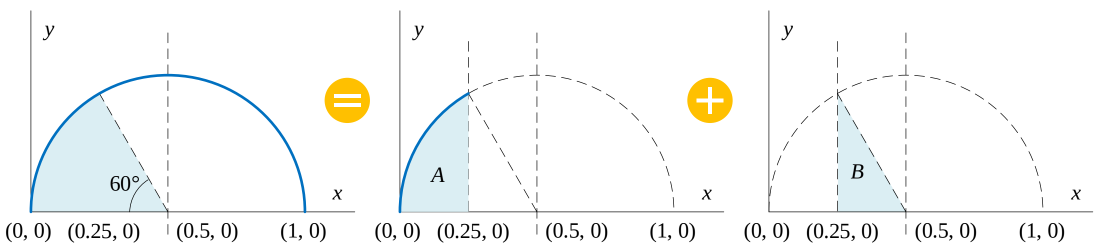


图 19. 面积关系

整个阴影区域的面积为 $\pi/24$。而这个区域的面积可以分成 A 和 B 两部分。

B 部分为直角三角形，面积很容易求得，具体值为：

$$B = \frac{\sqrt{3}}{32} \tag{28}$$

因此，A 的面积为扇形面积减去 B 的面积：

$$A = \frac{\pi}{24} - \frac{\sqrt{3}}{32} \tag{29}$$

整理 (29)，得到圆周率和 A 的关系：

$$\pi = 24 \times \left(\frac{\sqrt{3}}{32} + A \right) \tag{30}$$

而面积 A 可以通过如下定积分得到：

$$A = \int_0^{\frac{1}{4}} \sqrt{x - x^2} \,\mathrm{d}x = \int_0^{\frac{1}{4}} \sqrt{x}\sqrt{1 - x} \,\mathrm{d}x \tag{31}$$

其中 $\sqrt{1-x}$ 可以用泰勒展开写成：

$$\sqrt{1-x} = 1 - \frac{1}{2}x - \frac{1}{8}x^2 - \frac{1}{16}x^3 - \frac{5}{128}x^4 \ldots \tag{32}$$

(32) 代入积分式 (31)，得到：

$$\begin{aligned}
A &= \int_0^{\frac{1}{4}} \sqrt{x - x^2} \,\mathrm{d}x \\
&= \int_0^{\frac{1}{4}} x^{\frac{1}{2}} \left(1 - \frac{1}{2}x - \frac{1}{8}x^2 - \frac{1}{16}x^3 - \frac{5}{128}x^4 .. \right) \mathrm{d}x \\
&= \int_0^{\frac{1}{4}} \left(x^{\frac{1}{2}} - \frac{1}{2}x^{\frac{3}{2}} - \frac{1}{8}x^{\frac{5}{2}} - \frac{1}{16}x^{\frac{7}{2}} - \frac{5}{128}x^{\frac{9}{2}} .. \right) \mathrm{d}x \\
&= \left(\frac{2}{3}x^{\frac{3}{2}} - \frac{1}{5}x^{\frac{5}{2}} - \frac{1}{28}x^{\frac{7}{2}} - \frac{1}{72}x^{\frac{9}{2}} - \frac{5}{704}x^{\frac{11}{2}} \ldots \right) \Bigg|_{x=0}^{x=\frac{1}{4}} \\
&= \frac{2}{3 \times 2^3} - \frac{1}{5 \times 2^5} - \frac{1}{28 \times 2^7} - \frac{1}{72 \times 2^9} - \frac{5}{704 \times 2^{11}} \ldots
\end{aligned} \tag{33}$$

这样 A 可以写成级数求和：

$$A = -\sum_{n=0}^{\infty} \frac{(2n)!}{2^{4n+2} (n!)^2 (2n-1)(2n+3)} \tag{34}$$

于是圆周率可以通过下式近似得到：

$$\boxed{\pi = 24 \times \left(\frac{\sqrt{3}}{32} - \sum_{n=0}^{\infty} \frac{(2n)!}{2^{4n+2} (n!)^2 (2n-1)(2n+3)} \right)} \tag{35}$$

图 20 所示为圆周率估算结果随 n 增加而不断收敛。观察曲线，可以发现这个估算过程收敛的速度很快。以上就是牛顿估算圆周率的方法。

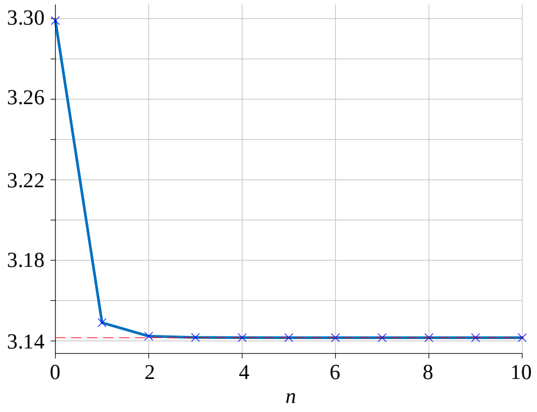


图 20. 牛顿方法估算圆周率

Bk3_Ch18_06.py 绘制图 22。

本章前文黎曼积分的思想——通过无限逼近来确定积分值。利用这一思路，我们也可以估算圆周率。如图 21 所示，我们可以用不断细分的正方形估算单位圆的面积，从而估算圆周率。而这一思路实际上就是蒙特卡洛模拟估算圆周率的内核。

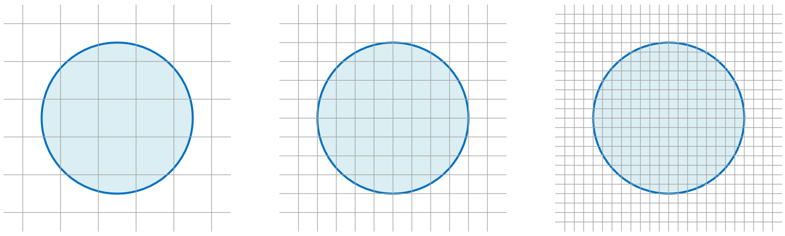


图 21. 用不断细分的正方形估算单位圆面积

蒙特卡洛模拟 (Monte Carlo simulation) 在大数据分析和机器学习中占据重要的位置。蒙特卡洛模拟以摩纳哥的赌城蒙特卡罗命名，是一种使用随机数并以概率理论为指导的数值计算方法。

下面，简单介绍利用如何用蒙特卡洛模拟估算圆周率 π。

在如图 21 所示单位圆 ($r = 1$) 的周围，构造一个以圆心为中心、以圆直径为边长的外切正方形。圆形面积 A_{circle} 和正方形面积 A_{square} 容易求得：

$$\begin{cases} A_{\text{circle}} = \pi r^2 = \pi \\ A_{\text{square}} = (2r)^2 = 4 \end{cases} \tag{36}$$

进而求得圆周率 π 和两个面积的比例关系：

$$\pi = 4 \times \frac{A_{\text{circle}}}{A_{\text{square}}} \tag{37}$$

然后，在这个正方形区域内产生满足均匀随机分布的 n 个数据点。生活中均匀随机分布无处不在。大家可以想象一下，一段时间没有人打理的房间内，落满灰尘。不考虑房间内特殊位置 (窗口、暖气口等) 的气流影响，灰尘的分布就类似“均匀随机分布”。

统计落入圆内的数据点个数 m 与总数据点总数 n 的比值，这个比值即为圆面积和正方形面积之比近似值。带入 (37) 可得：

$$\pi \approx 4 \times \frac{m}{n} \tag{38}$$

图 22 所示为四个蒙特卡洛模拟实验，随机点总数 n 分别为 100、500、1000 和 5000。可以发现随着 n 增大，估算得到的圆周率 π 不断接近真实值。

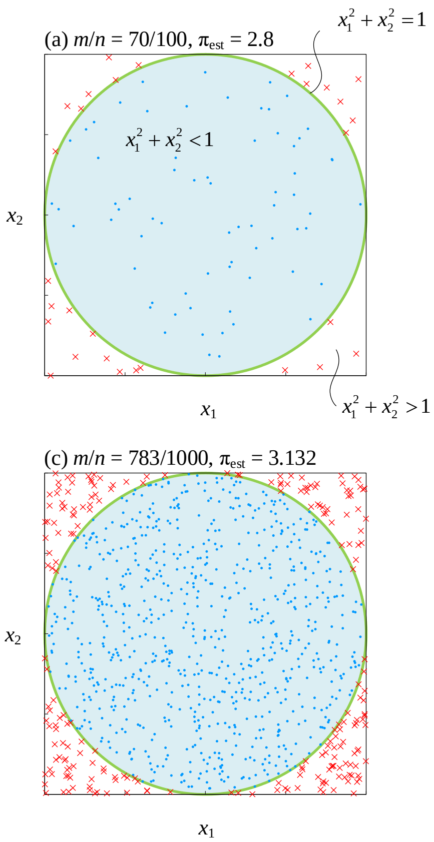


图 22. 蒙特卡洛模拟方法估算圆周率

这种估算圆周率的方法思想来源于在 18 世纪提出的布丰投针问题 (Buffon's needle problem)。实际上，布丰投针实验要比这里介绍的蒙特卡洛模拟方法更为复杂。本系列丛书将在《概率统计》一本书中和大家探讨布丰投针这一经典实验以及如何用 Python 编写代码实现模拟。

18.9 数值积分：黎曼求积

有些函数看着不复杂，竟然也没有积分解析解，比如高斯函数 $f(x) = \exp(-x^2)$。因为高斯函数积分很常用，人们还创造出误差函数。对于没有解析解的积分，我们通常使用数值积分方法。本节讨论如何用数值方法估算积分。

将平面图形切成细长条

德国数学家黎曼 (Bernhard Riemann, 1826 ~ 1866) 提出了一个求积解决方案——将不规则图形切成细长条。然后这些细长条近似看成一个个矩形，计算出它们的面积。这些矩形面积求和，可以用来近似不规则形状的面积。

狭长长方形越细，也就是图 23 中 Δx 越小，长方形越贴合区域形状，就越能精确估算面积。特别地，当细长条的宽度 Δx 趋近于 0 时，得到的面积的极限值就是不规则形状的面积。

看到图 23 这幅图，大家有没有想到本书第 3 章介绍圆周率估算时，刘徽说的“割之弥细，所失弥少，割之又割，以至于不可割，则与圆周合体而无所失矣。”两者思想如出一辙。

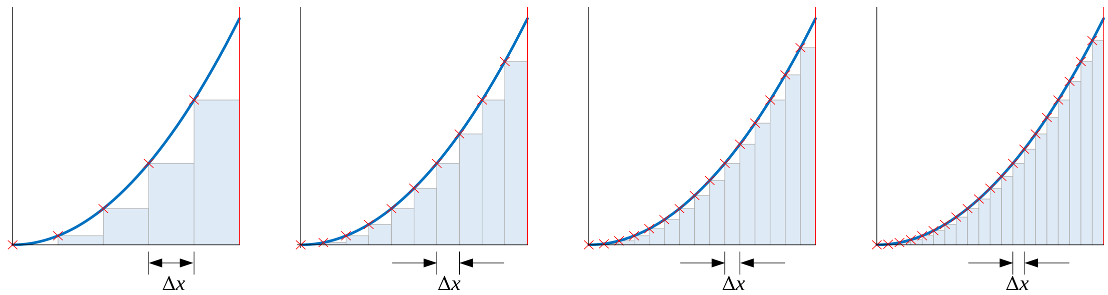


图 23. 细长条切得越细，面积估算越精确

将立体图形切成细高立方体

也用黎曼求积思路计算体积。如图 24 所示，我们可以用一个个细高立方体体积之和来近似估算几何体的体积。

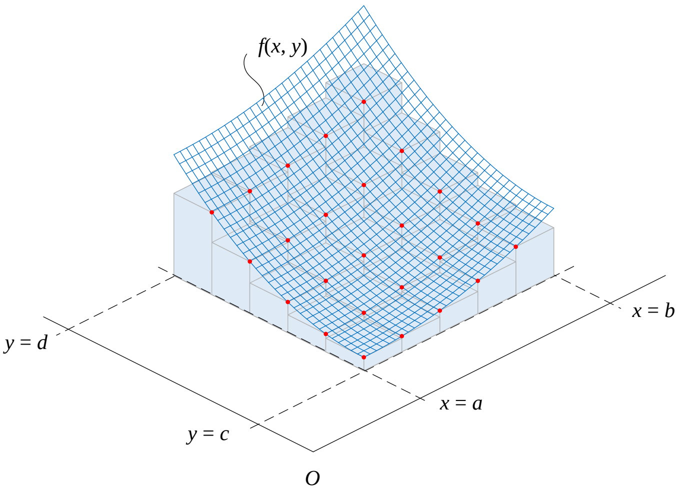


图 24. 将不规则几何体分割成细高立方体

如图 25 所示，随着细长立方体不断变小，这些立方体的体积之和不断接近不规则几何体的体积。

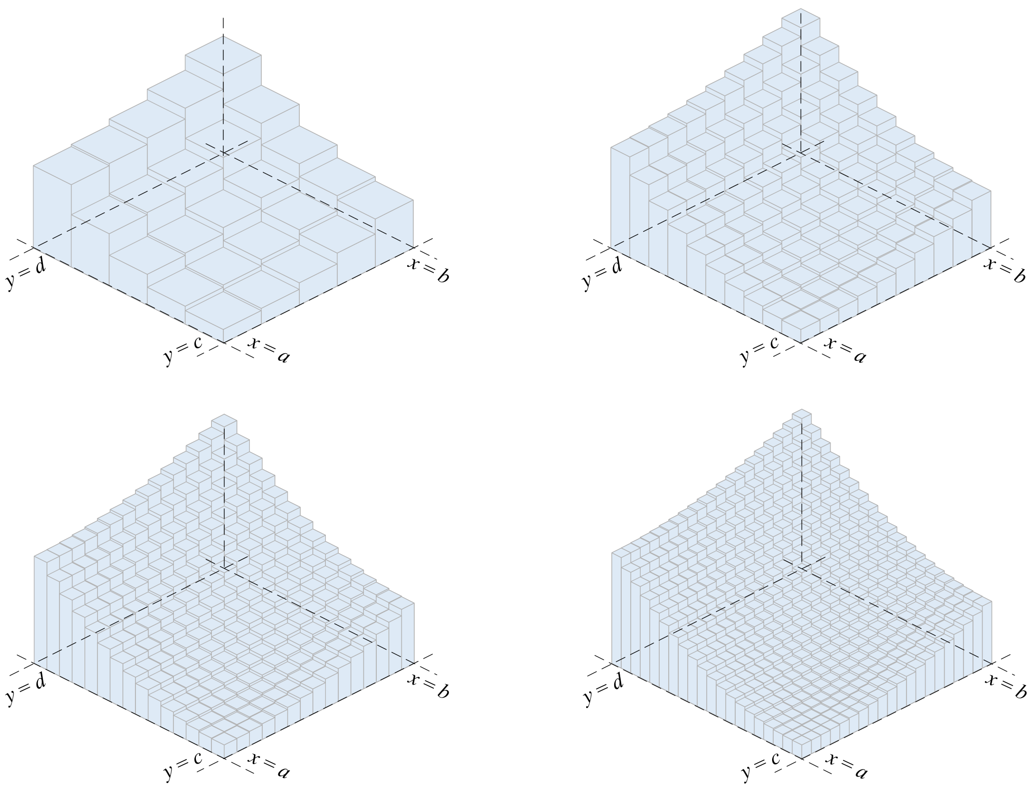


图 25. 随着细长立方体不断变小，立方体体积之和不断接近不规则形体的真实体积

基本数值积分方法

如图 26 (a) 所示，在 $[a, b]$ 区间内，为估算函数在区间内和 x 轴形成的面积，用左侧 a 点的函数值 $f(a)$ 进行积分估值运算：

$$\int_a^b f(x)\mathrm{d}x \approx (b-a)f(a) \tag{39}$$

这种方法叫做向前差分，也叫 left Riemann sum。这实际上就是用矩阵面积估算函数积分。

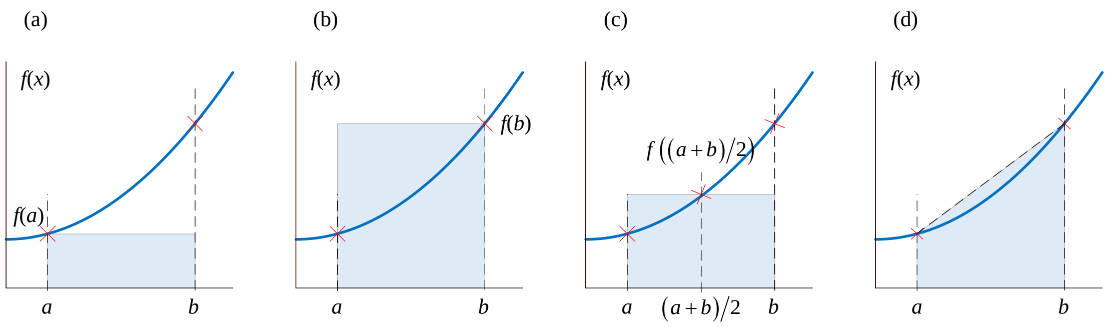


图 26. 四种不同方法

如图 26 (b) 所示，用 $[a, b]$ 区间右侧 b 点函数值 $f(b)$ 进行积分估值运算叫做向后差分，也叫 right Riemann sum：

$$\int_a^b f(x)\mathrm{d}x \approx (b-a)f(b) \tag{40}$$

如图 26 (c) 所示，用 $[a, b]$ 区间中间点 $(a+b)/2$ 函数值 $f((a+b)/2)$ 作为矩形高度来估值叫做中值差分，也叫 middle Riemann sum：

$$\int_a^b f(x)\mathrm{d}x \approx (b-a)f\left(\frac{a+b}{2}\right) \tag{41}$$

图 26 中前三种都是用矩形面积估算积分。

图 26 (d) 给出的是所谓梯形法，这种方法用 $f(a)$ 和 $f(b)$ 的平均值：

$$\int_a^b f(x)\mathrm{d}x \approx (b-a)\left(\frac{f(a)+f(b)}{2}\right) \tag{42}$$

如果数值积分采用固定步长 Δx。把 $[a, b]$ 区间分成 n 段，Δx 为：

$$\Delta x = \frac{b-a}{n} \tag{43}$$

当然，我们也可以采用可变步长，这不是本节要介绍的内容。

实践中，我们还会用到其他的数值积分方法。它们的差别一般在于两点之间的插值方法，也就是用什么样的简单函数尽可能逼近原函数。比如，图 26 前三种方法采用的是水平线 (常数函数)，只不过水平线的高度不同而已。图 26 中第四种方法采用两点之间斜线，即一次函数。再比如，辛普森法 (Simpson method) 用抛物线插值，牛顿-柯蒂斯法 (Newton-Cotes method) 采用的是 Lagrange 插值。

代码实现

本节仅介绍如何用代码实现图 26 中前三种数值积分方法。

图 27、图 28、图 29 三幅图对比步长 Δx 分别取 0.2、0.1、0.05 时三种数值积分法结果。图 30 所示为随分段数 n 增大三种数值积分结果不断收敛过程。容易发现，middle Riemann sum 更快地逼近真实值，它的精度显然更高。感兴趣的读者，可以自行了解数值积分中代数精度和误差等概念。

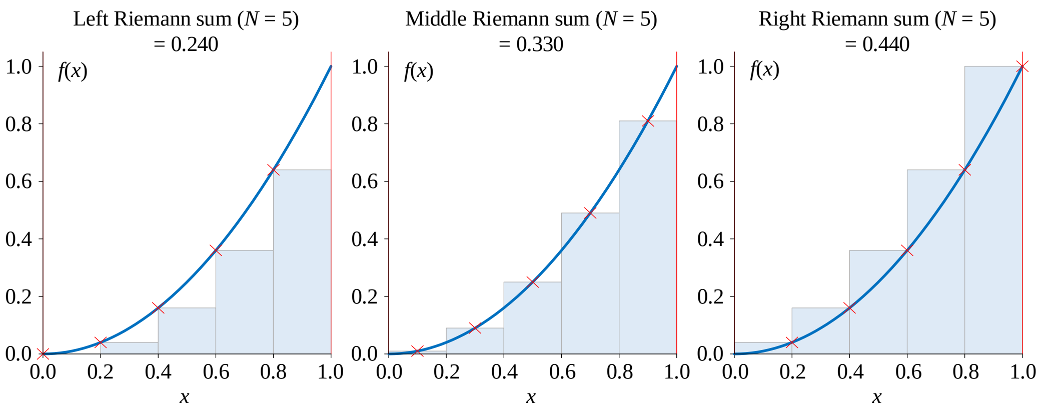


图 27. 步长 $\Delta x = 0.2$

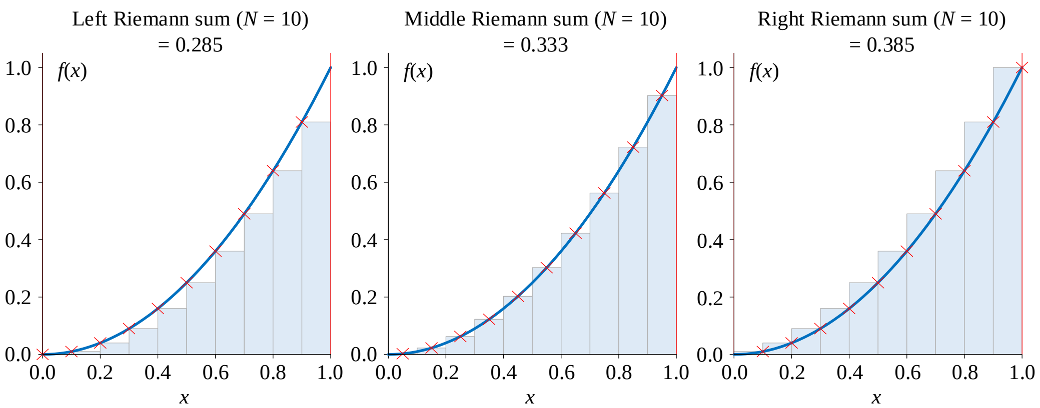


图 28. 步长 $\Delta x = 0.1$

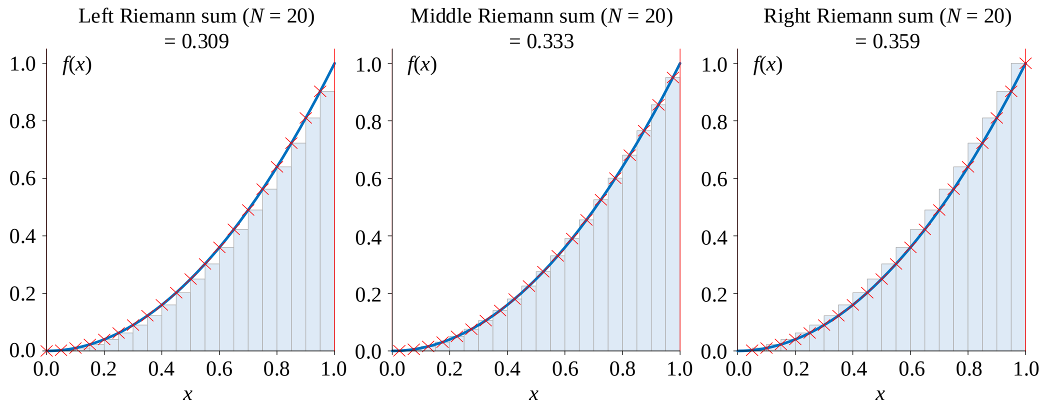


图 29. 步长 $\Delta x = 0.05$

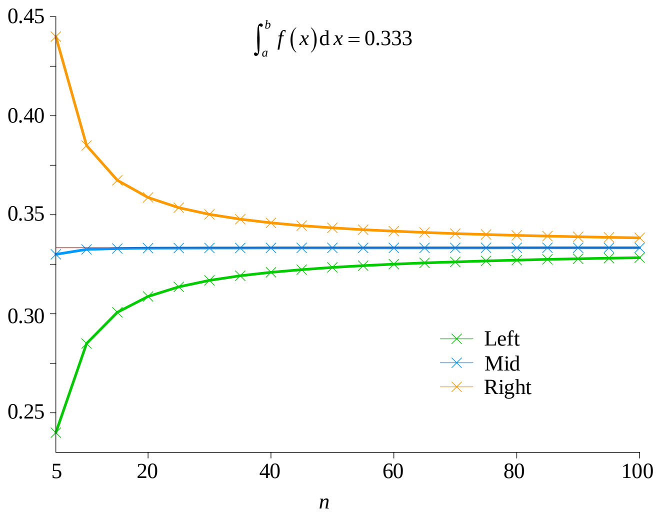


图 30. 三种数值积分随分段数 n 变化

`Bk3_Ch18_07.py` 绘制图 27、图 28、图 29。请大家自行编写代码绘制图 30。

在 `Bk3_Ch18_07.py` 基础上，我们做了一个 App 展示步长 Δx 对数值积分结果影响。请参考 `Streamlit_Bk3_Ch18_07.py`。

二重数值积分

本节最后介绍如何用代码实现二重数值积分。图 31 所示为某个二元函数曲面，我们要计算曲面和水平面在图中给出的区域内包围的体积。

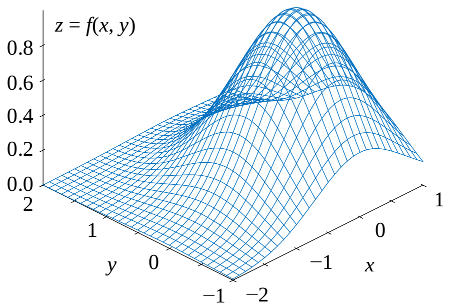


图 31. 二元函数曲面

图 32 所示为用数值积分方法，不断减小在 x 和 y 方向的步长，从而提高二重积分估算精度。

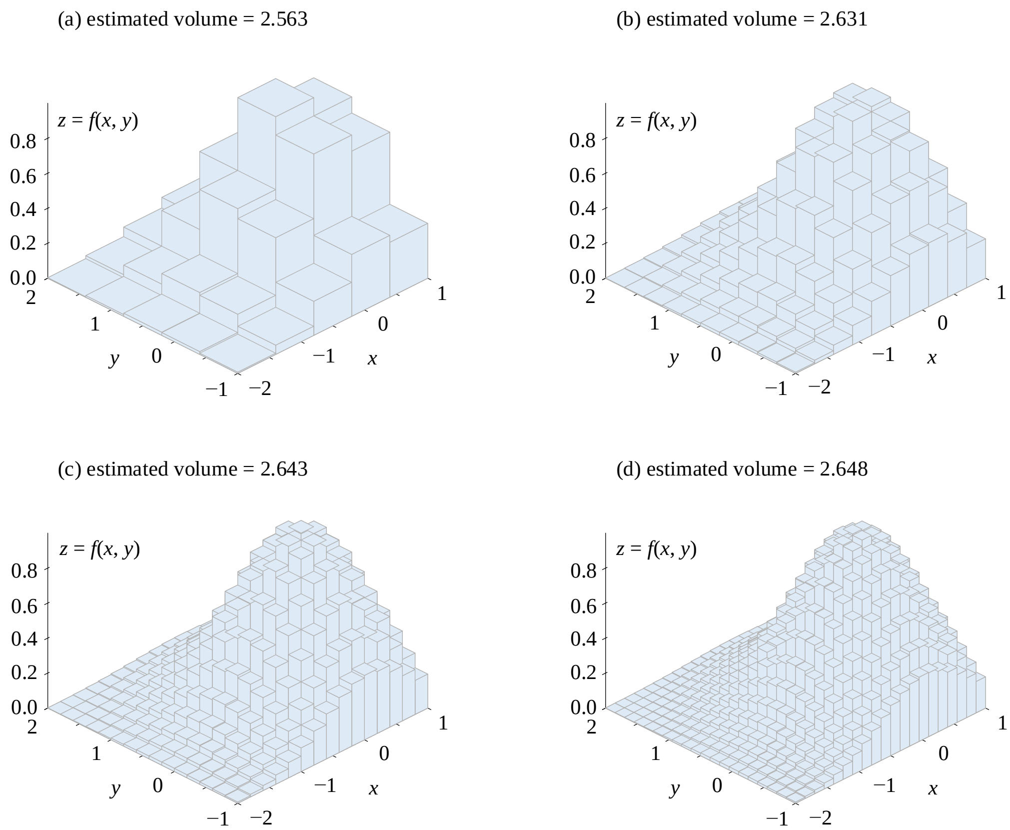


图 32. 不断减小步长提高估算精度

Bk3_Ch18_08.py 绘制图 32。

本书有关微积分的内容到此告一段落。机器学习特别是深度学习中，还有两个重要的微积分话题——自动求导、卷积。很遗憾，限于篇幅，本书只能蜻蜓点水般聊一聊在数据科学、机器学习最常用的微积分内容。本书介绍的微积分内容只是整个微积分体系的冰山一角，希望读者日后能够更全面学习提高。

有了微积分这个数学工具，下一章初步探讨优化问题相关内容。