

作者	生姜 DrGinger
脚本	生姜 DrGinger
视频	崔崔 CuiCui
开源学习资源	<a href="https://github.com/Visualize-ML">https://github.com/Visualize-ML</a>
平台	<a href="https://www.youtube.com/@DrGinger_Jiang">https://www.youtube.com/@DrGinger_Jiang</a> <a href="https://space.bilibili.com/3546865719052873">https://space.bilibili.com/3546865719052873</a> <a href="https://space.bilibili.com/513194466">https://space.bilibili.com/513194466</a>

## 4.4 手解行列式



### 本节你将掌握的核心技能：

- ▶ 余子矩阵：去掉指定元素所在行、列。
- ▶ 余子式：余子矩阵的行列式，标量。
- ▶ 代数余子式：对余子式进行符号调整。
- ▶ 拉普拉斯展开：选定任意一行或列，将每个元素乘以其代数余子式并求和，得到行列式。
- ▶ 优先选择含零最多的行或列展开，可减少计算量。

在本节中，我们将以  $3 \times 3$  行列式为例，介绍如何手解行列式。

不需要了解手解行列式的读者，这一节可以跳过。

虽然本章前面已经给出了  $3 \times 3$  矩阵  $A$  的行列式的公式：

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \quad (1)$$

$$= a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} - a_{12}a_{21}a_{33} - a_{11}a_{23}a_{32}$$

但是，这个公式并不容易记住。因此，我们将介绍一个更具结构化的方法——**拉普拉斯展开** (Laplace Expansion)，并引入**余子式** (minor)、**代数余子式** (cofactor) 这两个概念，以帮助我们更系统地计算行列式。

### 余子式

在计算行列式时，第一个重要的概念是余子式。

给定一个  $n \times n$  方阵  $A$ ：

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} \end{bmatrix} \quad (2)$$

矩阵  $A$  的第  $i$  行、第  $j$  列元素  $a_{ij}$  对应的**余子式**记作  $M_{ij}$ 。

**余子式**  $M_{ij}$  是从矩阵  $A$  中去掉第  $i$  行、第  $j$  列后，剩下的**余子矩阵**的行列式

$$M_{i,j} = \det(A_{i,j}) \quad (3)$$

其中，**余子矩阵**  $A_{i,j}$  形状为  $(n-1) \times (n-1)$ 。

⚠ 注意，余子矩阵  $A_{i,j}$  是个矩阵，余子式  $M_{ij}$  是个标量。

### 举个例子

对于  $3 \times 3$  方阵  $A$  来说，每个元素  $a_{ij}$  都有一个相应的  $2 \times 2$  **余子矩阵**，每个**余子矩阵** (矩阵) 有自己的**余子式** (标量)。

如图 1 (a) 所示，第 1 行、第 1 列元素  $a_{1,1}$  的**余子式**为

$$M_{1,1} = \begin{vmatrix} a_{2,2} & a_{2,3} \\ a_{3,2} & a_{3,3} \end{vmatrix} = a_{2,2}a_{3,3} - a_{2,3}a_{3,2} \quad (4)$$

图 1 (a) 中，蓝色方块代表元素  $a_{1,1}$ ；灰色 (还有蓝色) 色块为去掉的元素；绿色代表剩余的元素，它们构成了余子矩阵。

图 1 (b) 中，第 1 行、第 2 列元素  $a_{1,2}$  的**余子式**

$$M_{1,2} = \begin{vmatrix} a_{2,1} & a_{2,3} \\ a_{3,1} & a_{3,3} \end{vmatrix} = a_{2,1}a_{3,3} - a_{2,3}a_{3,1} \quad (5)$$

图 1 (c) 中，第 1 行、第 3 列元素  $a_{1,3}$  的**余子式**

$$M_{1,3} = \begin{vmatrix} a_{2,1} & a_{2,2} \\ a_{3,1} & a_{3,2} \end{vmatrix} = a_{2,1}a_{3,2} - a_{2,2}a_{3,1} \quad (6)$$

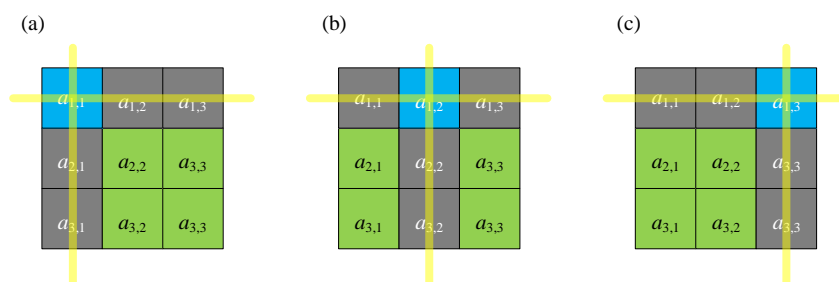


图 1.  $3 \times 3$  方阵第 1 行各个元素的余子矩阵

我们可以选择沿不同的行、列展开。

图 2 所示为按第 2 行展开后各个元素**余子矩阵**。

图 2 (a)，第 2 行、第 1 列元素  $a_{2,1}$  的**余子式**

$$M_{2,1} = \begin{vmatrix} a_{1,2} & a_{1,3} \\ a_{3,2} & a_{3,3} \end{vmatrix} = a_{1,2}a_{3,3} - a_{1,3}a_{3,2} \quad (7)$$

图 2 (b)，第 2 行、第 2 列元素  $a_{2,2}$  的**余子式**

$$M_{2,2} = \begin{vmatrix} a_{1,1} & a_{1,3} \\ a_{3,1} & a_{3,3} \end{vmatrix} = a_{1,1}a_{3,3} - a_{1,3}a_{3,1} \quad (8)$$

图 2 (c)，第 2 行、第 3 列元素  $a_{2,3}$  的**余子式**

$$M_{2,3} = \begin{vmatrix} a_{1,1} & a_{1,2} \\ a_{3,1} & a_{3,2} \end{vmatrix} = a_{1,1}a_{3,2} - a_{1,2}a_{3,1} \quad (9)$$

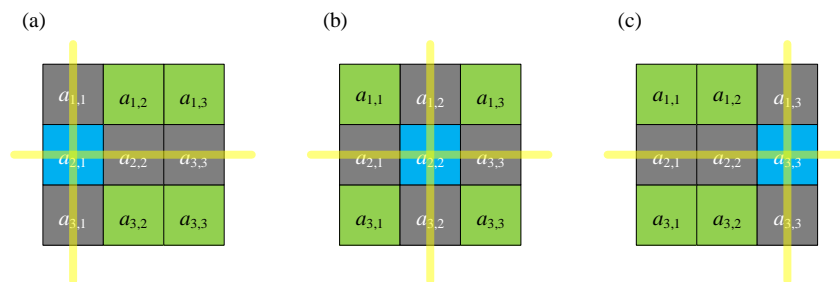


图 2.  $3 \times 3$  方阵第 2 行各个元素的余子矩阵

图 3 所示为按第 2 行展开后个元素**余子矩阵**，请大家计算这三个**余子式**。

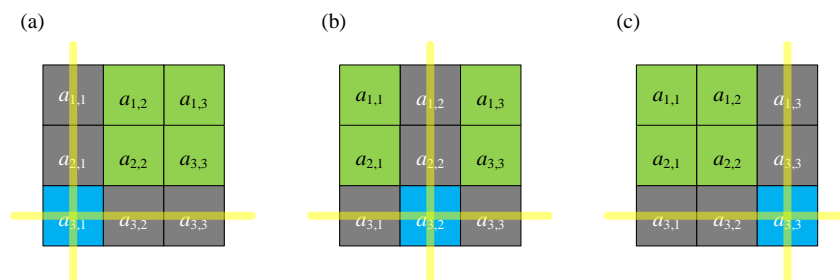


图 3.  $3 \times 3$  方阵第 3 行各个元素的余子矩阵

## 代入具体数值

给定如下  $3 \times 3$  方阵

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad (10)$$

根据 (4)，第 1 行、第 1 列元素  $a_{1,1}$  的**余子式**

$$M_{1,1} = \begin{vmatrix} 5 & 6 \\ 8 & 9 \end{vmatrix} = (5 \times 9) - (6 \times 8) = 45 - 48 = -3 \quad (11)$$

根据 (5)，第 1 行、第 2 列元素  $a_{1,2}$  的**余子式**

$$M_{1,2} = \begin{vmatrix} 4 & 6 \\ 7 & 9 \end{vmatrix} = (4 \times 9) - (6 \times 7) = 36 - 42 = -6 \quad (12)$$

根据 (6)，第 1 行、第 3 列元素  $a_{1,3}$  的**余子式**

$$M_{1,3} = \begin{vmatrix} 4 & 5 \\ 7 & 8 \end{vmatrix} = (4 \times 8) - (5 \times 7) = 32 - 35 = -3 \quad (13)$$

请大家计算 (10) 给出方阵的其他六个**余子式**。

## 代数余子式

**代数余子式**是在余子式的基础上引入符号的调整。

**代数余子式**  $C_{i,j}$  由如下公式定义：

$$C_{i,j} = (-1)^{i+j} M_{i,j} \quad (14)$$

其中， $(-1)^{i+j}$  是根据元素位置调整符号的因子。

以 (10) 给出方阵为例，计算第 1 行、第 1 列元素  $a_{1,1}$  的**代数余子式**

$$C_{1,1} = (-1)^{1+1} M_{1,1} = 1 \times (-3) = -3 \quad (15)$$

计算第 1 行、第 2 列元素  $a_{1,2}$  的**代数余子式**

$$C_{1,2} = (-1)^{1+2} M_{1,2} = (-1) \times (-6) = 6 \quad (16)$$

计算第 1 行、第 3 列元素  $a_{1,3}$  的**代数余子式**

$$C_{1,3} = (-1)^{1+3} M_{1,3} = 1 \times (-3) = -3 \quad (17)$$

由于  $(-1)^{i+j}$  形成交替正负的模式，因此  $3 \times 3$  方阵的代数余子式符号调整矩阵为

$$\begin{bmatrix} + & - & + \\ - & + & - \\ + & - & + \end{bmatrix} \quad (18)$$

## 拉普拉斯展开

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)

拉普拉斯展开提供了一种更具系统性的方法来计算行列式。我们可以沿着任意一行、一列展开求解行列式。

比如，沿着第  $i$  行展开计算  $A$  的行列式，

$$\det(A) = \sum_{j=1}^n a_{i,j} C_{i,j} \quad (19)$$

比如指定第一行展开，行列式为

$$\det(A) = \sum_{j=1}^n a_{1,j} C_{1,j} \quad (20)$$

对于  $3 \times 3$  方阵  $A$ ，我们可以使用第一行展开， $A$  的行列式为

$$\det(A) = a_{1,1} C_{1,1} + a_{1,2} C_{1,2} + a_{1,3} C_{1,3} \quad (21)$$

代入 (11) ~ (17) 计算的结果：

$$\det(A) = (1 \times (-3)) + (2 \times 6) + (3 \times (-3)) = -3 + 12 - 9 = 0 \quad (22)$$

这也告诉我们，求解一个方阵的行列式有很多不同的路径。

## 步骤

总结上述内容，我们可以得到求解行列式的具体步骤：

- 选取计算行列式展开时用到的某一行或列，提取**余子矩阵** (方阵)。
- 根据**余子矩阵** (方阵) 计算**余子式** (标量)；
- 计算每个元素的**代数余子式** (标量)，即调整符号后的**余子式** (标量)；
- 根据**拉普拉斯展开**，将选定行或列的元素与其**代数余子式**相乘，并求和，最终得到**行列式**的值。

通过这些步骤，我们可以系统地计算行列式，而不仅仅依赖记忆公式。这种方法可以扩展到更高维的矩阵。

## 一个完整的例子

给定如下  $3 \times 3$  方阵  $A$ ，让我们根据以上步骤沿行、列方向展开一步步计算行列式。

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 0 & 1 \\ 1 & 2 & 1 \end{bmatrix} \quad (23)$$

### a) 取出余子矩阵

取出每个元素对应的**余子矩阵**  $A_{i,j}$ 。

回顾**余子矩阵**这个概念。对于  $3 \times 3$  方阵  $A$ ,  $A_{i,j}$  是去掉第  $i$  行、第  $j$  列后剩下的  $2 \times 2$  矩阵。

**a.1)** 如果选取第 1 行展开, **余子矩阵**  $A_{1,1}$  (去掉第 1 行、第 1 列) 为

$$A_{1,1} = \begin{bmatrix} 0 & 1 \\ 2 & 1 \end{bmatrix} \quad (24)$$

**余子矩阵**  $A_{1,2}$  (去掉第 1 行、第 2 列):

$$A_{1,2} = \begin{bmatrix} 3 & 1 \\ 1 & 1 \end{bmatrix} \quad (25)$$

**余子矩阵**  $A_{1,3}$  (去掉第 1 行、第 3 列):

$$A_{1,3} = \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix} \quad (26)$$

**a.2)** 如果选取第 2 行展开, **余子矩阵**  $A_{2,1}$  (去掉第 2 行、第 1 列):

$$A_{2,1} = \begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix} \quad (27)$$

**余子矩阵**  $A_{2,2}$  (去掉第 2 行、第 2 列):

$$A_{2,2} = \begin{bmatrix} 1 & 3 \\ 1 & 1 \end{bmatrix} \quad (28)$$

**余子矩阵**  $A_{2,3}$  (去掉第 2 行、第 3 列):

$$A_{2,3} = \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix} \quad (29)$$

**a.3)** 如果选取第 3 行展开, **余子矩阵**  $A_{3,1}$  (去掉第 3 行、第 1 列):

$$A_{3,1} = \begin{bmatrix} 2 & 3 \\ 0 & 1 \end{bmatrix} \quad (30)$$

**余子矩阵**  $A_{3,2}$  (去掉第 3 行、第 2 列):

$$A_{3,2} = \begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix} \quad (31)$$

**余子矩阵**  $A_{3,3}$  (去掉第 3 行、第 3 列):

$$A_{3,3} = \begin{bmatrix} 1 & 2 \\ 3 & 0 \end{bmatrix} \quad (32)$$

**b) 计算余子式**

计算每个余子矩阵  $A_{i,j}$  的行列式，即余子式  $M_{i,j}$ 。

**b.1)** 第 1 行各个元素的余子式。余子式  $M_{1,1}$ ：

$$M_{1,1} = \begin{vmatrix} 0 & 1 \\ 2 & 1 \end{vmatrix} = (0 \times 1) - (1 \times 2) = 0 - 2 = -2 \quad (33)$$

余子式  $M_{1,2}$ ：

$$M_{1,2} = \begin{vmatrix} 3 & 1 \\ 1 & 1 \end{vmatrix} = (3 \times 1) - (1 \times 1) = 3 - 1 = 2 \quad (34)$$

余子式  $M_{1,3}$ ：

$$M_{1,3} = \begin{vmatrix} 3 & 0 \\ 1 & 2 \end{vmatrix} = (3 \times 2) - (0 \times 1) = 6 - 0 = 6 \quad (35)$$

**b.2)** 第 2 行各个元素的余子式。余子式  $M_{2,1}$ ：

$$M_{2,1} = \begin{vmatrix} 2 & 3 \\ 2 & 1 \end{vmatrix} = (2 \times 1) - (3 \times 2) = 2 - 6 = -4 \quad (36)$$

余子式  $M_{2,2}$ ：

$$M_{2,2} = \begin{vmatrix} 1 & 3 \\ 1 & 1 \end{vmatrix} = (1 \times 1) - (3 \times 1) = 1 - 3 = -2 \quad (37)$$

余子式  $M_{2,3}$ ：

$$M_{2,3} = \begin{vmatrix} 1 & 2 \\ 1 & 2 \end{vmatrix} = (1 \times 2) - (2 \times 1) = 2 - 2 = 0 \quad (38)$$

**b.3)** 第 3 行各个元素的余子式。余子式  $M_{3,1}$ ：

$$M_{3,1} = \begin{vmatrix} 2 & 3 \\ 0 & 1 \end{vmatrix} = (2 \times 1) - (3 \times 0) = 2 - 0 = 2 \quad (39)$$

余子式  $M_{3,2}$ ：

$$M_{3,2} = \begin{vmatrix} 1 & 3 \\ 3 & 1 \end{vmatrix} = (1 \times 1) - (3 \times 3) = 1 - 9 = -8 \quad (40)$$

余子式  $M_{3,3}$ ：

$$M_{3,3} = \begin{vmatrix} 1 & 2 \\ 3 & 0 \end{vmatrix} = (1 \times 0) - (2 \times 3) = 0 - 6 = -6 \quad (41)$$

### c) 计算代数余子式

**代数余子式**就是在**余子式**上加了符号，根据的法则为  $C_{i,j} = (-1)^{i+j} M_{i,j}$ 。

**c.1)** 第 1 行各个元素的**代数余子式**。**代数余子式**  $C_{1,1}$ :

$$C_{1,1} = (-1)^{1+1} M_{1,1} = (1) \times (-2) = -2 \quad (42)$$

**代数余子式**  $C_{1,2}$ :

$$C_{1,2} = (-1)^{1+2} M_{1,2} = (-1) \times (2) = -2 \quad (43)$$

**代数余子式**  $C_{1,3}$ :

$$C_{1,3} = (-1)^{1+3} M_{1,3} = (1) \times (6) = 6 \quad (44)$$

**c.2)** 第 2 行各个元素的**代数余子式**。**代数余子式**  $C_{2,1}$ :

$$C_{2,1} = (-1)^{2+1} M_{2,1} = (-1) \times (-4) = 4 \quad (45)$$

**代数余子式**  $C_{2,2}$ :

$$C_{2,2} = (-1)^{2+2} M_{2,2} = (1) \times (-2) = -2 \quad (46)$$

**代数余子式**  $C_{2,3}$ :

$$C_{2,3} = (-1)^{2+3} M_{2,3} = (-1) \times (0) = 0 \quad (47)$$

**c.3)** 第 3 行各个元素的**代数余子式**。**代数余子式**  $C_{3,1}$ :

$$C_{3,1} = (-1)^{3+1} M_{3,1} = (1) \times 2 = 2 \quad (48)$$

**代数余子式**  $C_{3,2}$ :

$$C_{3,2} = (-1)^{3+2} M_{3,2} = (-1) \times (-8) = 8 \quad (49)$$

**代数余子式**  $C_{3,3}$ :

$$C_{3,3} = (-1)^{3+3} M_{3,3} = (1) \times (-6) = -6 \quad (50)$$

### d) (行展开) 计算行列式

相信大家早已发现，我们实际上提供了沿行展开计算  $3 \times 3$  方阵行列式的三条“平行”路径。

**d.1)** 第 1 行展开计算**行列式**

$$\det(A) = a_{1,1}C_{1,1} + a_{1,2}C_{1,2} + a_{1,3}C_{1,3} \quad (51)$$

代入值:

$$\det(A) = 1 \times (-2) + 2 \times (-2) + 3 \times 6 = -2 - 4 + 18 = 12 \quad (52)$$



**d.2) 第 2 行展开计算行列式**

$$\det(A) = a_{2,1}C_{2,1} + a_{2,2}C_{2,2} + a_{2,3}C_{2,3} \quad (53)$$

代入值：

$$\det(A) = 3 \times 4 + 0 \times (-2) + 1 \times 0 = 12 + 0 + 0 = 12 \quad (54)$$

结果显然和 (52) 一致。

**d.3) 再用第 3 行展开计算行列式**

$$\det(A) = a_{3,1}C_{3,1} + a_{3,2}C_{3,2} + a_{3,3}C_{3,3} \quad (55)$$

代入值：

$$\det(A) = 1 \times 2 + 2 \times 8 + 1 \times (-6) = 2 + 16 - 6 = 12 \quad (56)$$

我们发现无论选择哪一行展开，都得到  $\det(A) = 12$ 。

**e) (列展开) 计算行列式**

既然可以行展开计算行列式，也可以列展开。下面让我们逐个看一下。

**e.1) 第 1 列展开计算行列式**

$$\det(A) = a_{1,1}C_{1,1} + a_{2,1}C_{2,1} + a_{3,1}C_{3,1} \quad (57)$$

代入值：

$$\det(A) = 1 \times (-2) + 3 \times 4 + 1 \times 2 = -2 + 12 + 2 = 12 \quad (58)$$

**e.2) 第 2 列展开计算行列式**

$$\det(A) = a_{1,2}C_{1,2} + a_{2,2}C_{2,2} + a_{3,2}C_{3,2} \quad (59)$$

代入值：

$$\det(A) = 2 \times (-2) + 0 \times (-2) + 2 \times 8 = -4 + 0 + 16 = 12 \quad (60)$$

结果显然和 (52) 一致。

**e.3) 再用第 3 列展开计算行列式**

$$\det(A) = a_{1,3}C_{1,3} + a_{2,3}C_{2,3} + a_{3,3}C_{3,3} \quad (61)$$

代入值：

$$\det(A) = 3 \times 6 + 1 \times 0 + 1 \times (-6) = 18 + 0 - 6 = 12 \quad (62)$$

我们发现无论选择哪一列展开，都得到  $\det(A) = 12$ 。

至此，我们找到了 6 条计算  $3 \times 3$  方阵的行列式，结果完全相同。

在进行拉普拉斯展开时，应优先选择包含最多 0 的行或列，因为这些项的计算会变得更简单。比如，对于如下矩阵  $A$ ，沿第 2 列展开计算行列式，计算量最小（只需要计算一个  $2 \times 2$  方阵行列式）。

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix} \quad (63)$$

当然，如果遇到对角方阵、上三角、下三角矩阵，计算行列式就更简单了。

## Python 编程计算行列式

代码 1 展示如何用 Python 编程计算行列式。下面聊聊其中关键语句。

**a** 定义一个叫 `cal_minor` 的函数，`def` 是“定义”的意思。这个函数有三个输入：一个矩阵 `A`，一个行号 `row_idx`，一个列号 `col_idx`。它的作用是：从这个矩阵中，把指定的那一行和那一列删掉，留下来的就是我们要的“余子矩阵”。

**b** 先用 `numpy.delete()` 把矩阵 `A` 的第 `row_idx` 行删掉，`axis=0` 表示删“行”。删完之后结果存在 `A_i` 这个变量里。接下来，在上一步的结果 `A_i` 里，把第 `col_idx` 列删掉。`axis=1` 表示删“列”。删完之后的结果存在 `A_ij` 里，这个就是我们想要的“余子矩阵”。

**c** 定义了一个新的函数，叫 `determinant`。这个函数的目标是：给它一个矩阵 `A`，它会计算出这个矩阵的“行列式”。

**d** 首先判断矩阵是不是只有一行一列。如果是，那它就是一个数字，不用计算了，直接返回这个数字。然后判断，如果矩阵是  $2 \times 2$  的，这里直接按照固定的规则算出它的行列式。就是对角线相乘再相减。

**e** 开始一个循环，用 `col_idx` 从 0 数到 `n-1`。也就是说，我们会处理矩阵的第一行的每一列。因为我们要“沿第一行展开”，所以每列都要参与计算。

然后，调用前面写好的 `cal_minor` 函数，传入矩阵 `A`，第 0 行和当前这一列的编号，得到去掉这一行这一列之后的“余子矩阵”。

再计算“代数余子式”。注意，这里用到了“递归”：我们又在调用 `determinant()` 自己！程序会不断地拆成小矩阵，再小矩阵，直到变成  $2 \times 2$  或  $1 \times 1$  才停止。

`det += A[0, col] * cofactor` 是关键一步，即 Laplace 展开。我们用原矩阵第一行的每个元素，乘上它对应的“代数余子式”，加到 `det` 上。程序就是通过这一层层累加来最终算出大矩阵的行列式。

代码 1. 计算行列式 |  LA\_04\_04\_01.ipynb

```

## 初始化
import numpy as np

## 自定义函数，提取余子矩阵
a
def cal_minor(A, row_idx, col_idx):
    b
    A_i = np.delete(A, row_idx, axis=0)
    A_ij = np.delete(A_i, col_idx, axis=1)
    # 去掉指定的行、列
    return A_ij

## 用Laplace展开递归计算行列式
c
def determinant(A):
    n = A.shape[0]
    d
    if n == 1:
        return A[0, 0]
    if n == 2:
        return A[0, 0] * A[1, 1] - A[0, 1] * A[1, 0]

    det = 0
    e
    for col_idx in range(n):
        minor = cal_minor(A, 0, col_idx) # 沿第一行展开
        cofactor = ((-1) ** col_idx) * determinant(minor)
        # 计算代数余子式
        # 相当于 (-1) ** ((col_idx + 1) + 1) = (-1) ** col_idx
        det += A[0, col_idx] * cofactor # 计算行列式

    return det

## 测试
A = np.array([[1, 2, 3],
              [3, 0, 1],
              [1, 2, 1]])
A_det = determinant(A)

```



请大家用 DeepSeek/ChatGPT 等工具完成本节如下习题。

**Q1.** 对于下列方阵请大家任选一行、一列计算对应元素的余子式 (标量)。也请大家思考，选择哪一行，哪一列展开最方便计算。

► 
$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

► 
$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)

► 
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

**Q2.** 用本节拉普拉斯展开计算如下  $2 \times 2$  矩阵行列式。

► 
$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

**Q3.** 请大家用列方向展开计算 (23) 行列式。

**Q4.** 请自学对角线法则 (Sarrus 法则)。

**Q5.** 请修改代码 1，增加语句判断矩阵是否为方阵，只有方阵才继续行列式运算。

**Q6.** 请修改代码 1，增加语句优先选择零最多的行或列展开计算行列式。