

19

Sequences

数列

用几何可视化手段展示数列和极限



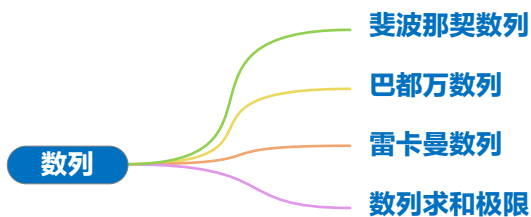
我把心和灵魂投入到绘画中，并在这个过程中失去了理智。

I put my heart and my soul into my work, and have lost my mind in the process.

—— 文森特·梵高 (Vincent van Gogh) | 荷兰后印象派画家 | 1853 ~ 1890



- ◀ `matplotlib.patches.Polygon` 多边形对象
- ◀ `matplotlib.pyplot.Rectangle()` 矩形对象
- ◀ `matplotlib.transforms.Affine2D` 完成对象的二维仿射变换，比如平移、缩放、旋转等几何操作
- ◀ `numpy.cumsum()` 计算累计求和
- ◀ `numpy.insert()` 在数组特定位置插入数值



19.1 什么是数列？

数列是按照一定规律排列的数字集合。常见类型包括**等差数列** (arithmetic progression, arithmetic sequence)、**等比数列** (geometric progression, geometric sequence)、**斐波那契数列** (Fibonacci sequence) 等等。

本质上，数列也是一种特殊函数。在机器学习中，数列常用于表示数据序列，如时间序列预测等等。

一般情况，我们可以通过折线图、火柴梗图、热图展示数列趋势和模式。本章将用几何视角展示数列和数列求和极限。

19.2 斐波那契数列

《编程不难》介绍过斐波那契数列。我们可以在自然界中找到很多斐波那契数列的例子，如植物的分枝结构、蜂窝的排列方式等。图 1 所示为斐波那契数列的一种可视化方案。

Bk2_Ch19_01.ipynb 绘制图 1，下面聊聊代码 1 中关键语句。

a 用 `matplotlib.pyplot.cm`，简作 `plt.cm`，生成一组渐变色。`plt.cm.Blues_r` 根据 `np.linspace(0, 1, num + 1)` 数值生成一组蓝色渐变色。

b 为正方形的旋转角度。

c 为正方形的边长。

d 为正方形“锚点”的坐标。不考虑旋转的话，锚点为矩形左下角顶点的位置坐标。

e 用 `plt.Rectangle()` 矩形对象创建“正方形”实例。

`rec_loc_idx` 是矩形左下角的位置坐标。

`width=side_idx` 和 `height=side_idx` 分别设置了矩形的宽度和高度。

`facecolor=colors[i + 1]` 设置了矩形的填充颜色。

`edgecolor='k'` 设置了矩形的边框颜色为黑色。

`transform` 参数用于指定矩形的变换方式，这里使用了仿射变换 (affine transformation)。

`Affine2D().rotate_deg_around(x_array[i], y_array[i], rotate_i)` 是一个仿射变换，表示绕 `(x_array[i], y_array[i])` 点旋转 `rotate_i` 度。

`ax.transData` 表示将变换应用于数据坐标系。

f 在轴对象 `ax` 添加图形对象。

```

fig, ax = plt.subplots(figsize = (8,8))

a colors = plt.cm.Blues_r(np.linspace(0,1,num + 1))

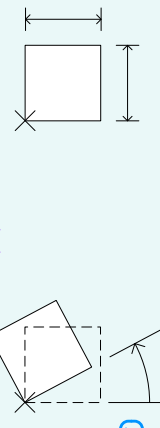
b for i in idx_array:
c     rotate_i = rotation[i]
d     side_idx = fibonacci_array_from_1[i]
    rec_loc_idx = np.array([x_array[i], y_array[i]])

    rec = plt.Rectangle(rec_loc_idx,
                        width=side_idx,
                        height=side_idx,
                        facecolor=colors[i + 1],
                        edgecolor = 'k',
                        transform=Affine2D().rotate_deg_around(
                            x_array[i], y_array[i], rotate_i)
                        + ax.transData)

e
f ax.add_patch(rec)

ax.set_aspect('equal')
ax.plot(0, 0, color='r', marker='x', markersize=10)
ax.axis('off')
plt.show()

```



代码 1. 可视化斐波那契数列 | Bk2_Ch19_01.ipynb

Bk2_Ch19_02.ipynb 绘制图 2。图 2 在图 1 基础上增加了螺旋线，请大家自行分析。

19.3 巴都万数列

图 3 可视化**巴都万数列** (Padovan Sequence)。巴都万数列前 15 项为 1, 1, 1, 2, 2, 3, 4, 5, 7, 9, 12, 16, 21, 28, 37, 49。

大家可以发现，这个数列的前 3 项均为 1，之后的数值递推关系定义为 $P(n) = P(n - 2) + P(n - 3)$ 。

$n > 6$ 时，巴都万数列也可以写成 $P(n) = P(n - 1) + P(n - 5)$ 。这是本例可视化用的公式。

Bk2_Ch19_03.ipynb 绘制图 3。

这个代码中，作图技巧是采用 `matplotlib.patches.Polygon()` 绘制正三角形。代码用到的重要数学工具是平面旋转。请大家自行分析这段代码。

19.4 雷卡曼数列

在数学和计算机科学中，**雷卡曼数列** (Recamán sequence) 通常使用递归的方式来定义。Recamán 数列前 20 项为 0, 1, 3, 6, 2, 7, 13, 20, 12, 21, 11, 22, 10, 23, 9, 24, 8, 25, 43, 62。

图 4 用圆弧方式展示雷卡曼数列的前 60 列。Bk2_Ch19_04.ipynb 绘制图 4 并给出通项公式。

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

19.5 数列求和极限

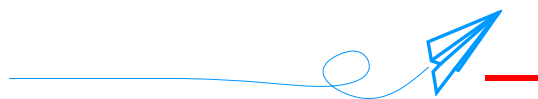
图 5 和图 6 则用几何方式展现数列求和极限。当项数逐渐增加，且数列求和无限接近某个值时，这个值被称为数列求和极限。

如图 5 所示，数列求和 $1/4 + 1/16 + 1/64 + 1/256 + \dots$ 趋向于 $1/3$ 。

Bk2_Ch19_05.ipynb 绘制图 5，请大家自行分析代码。

如图 6 所示，数列求和 $1/2 + 1/4 + 1/8 + 1/16 + \dots$ 趋向于 1 。

Bk2_Ch19_06.ipynb 绘制图 6，请大家自行分析代码。



本章讲解如何可视化斐波那契数列、巴都万数列、雷卡曼数列，以及数列求和极限。

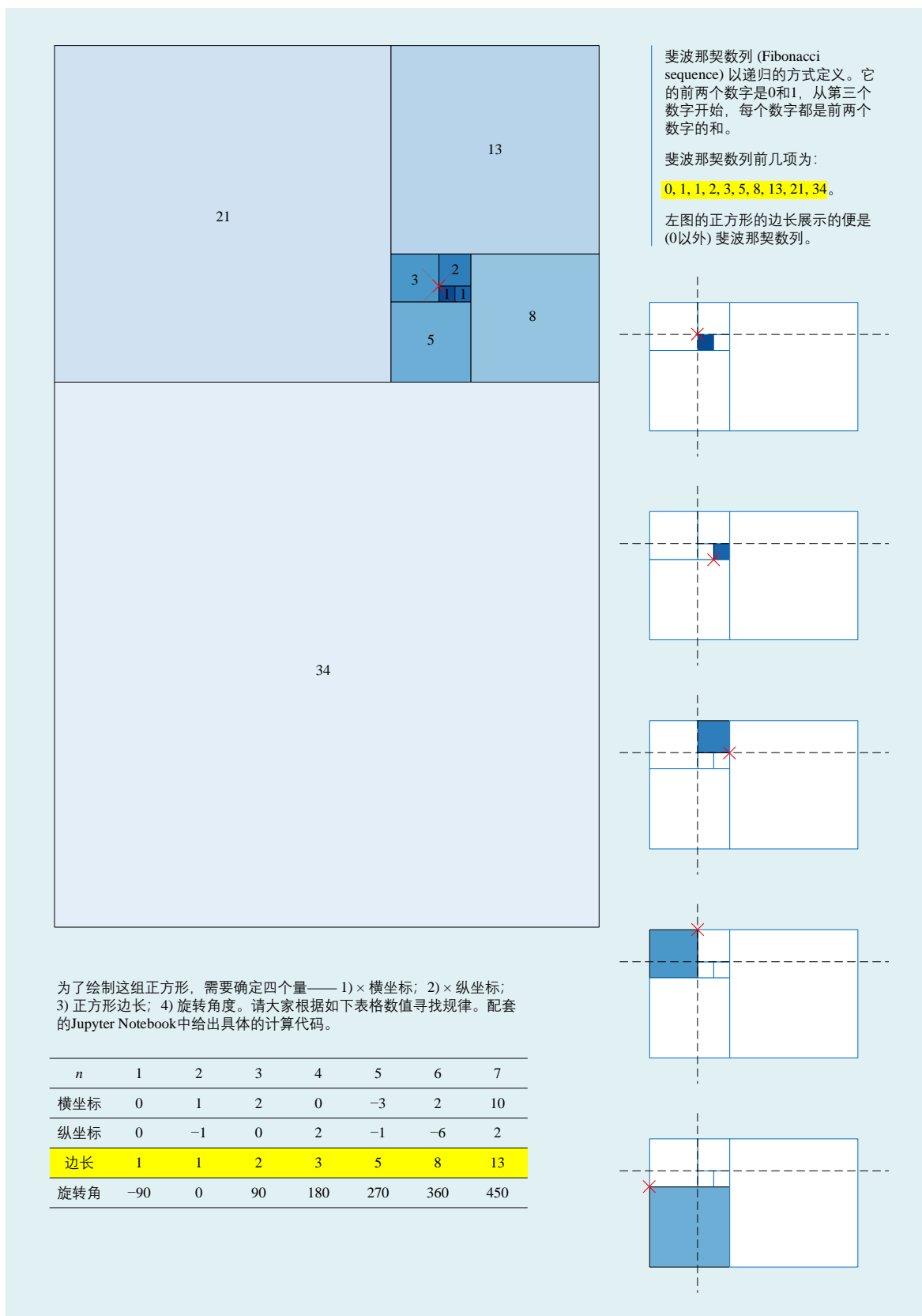


图 1. 可视化斐波那契数列 | Bk2_Ch19_01.ipynb

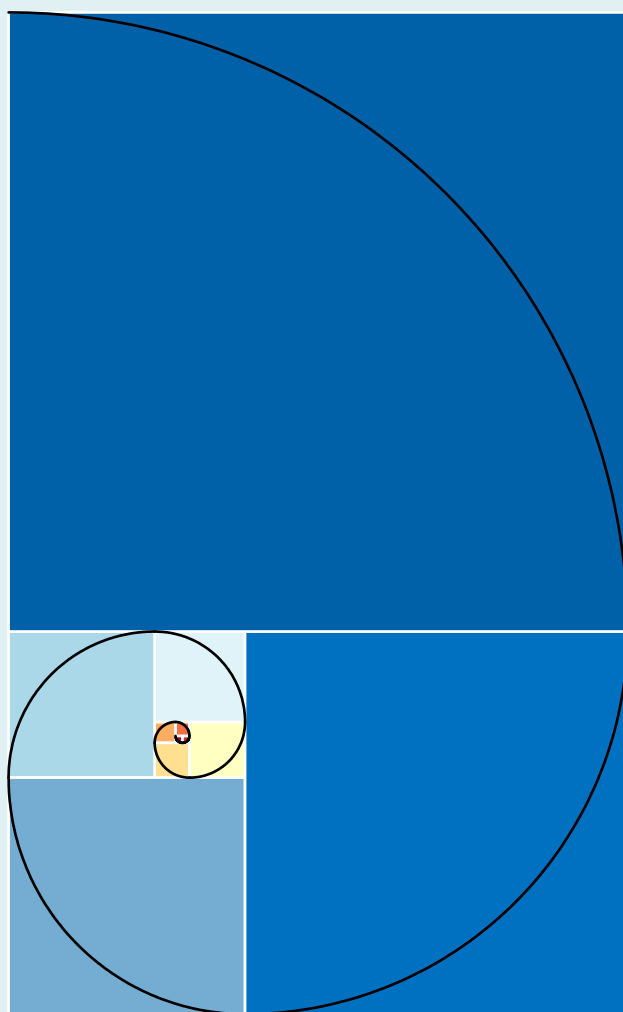
本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

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>

本书配套微课视频均发布在 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



为了绘制斐波那契螺旋线，需要确定四个量——1) 圆心横坐标；2) 圆心纵坐标；3) 半径；4) 1/4圆弧起始角。请大家根据如下表格数值寻找规律。配套的Jupyter Notebook中给出具体的计算代码。

n	1	2	3	4	5	6	7
横坐标	1	1	0	0	2	2	-3
纵坐标	0	0	0	-1	-1	2	2
半径	1	1	2	3	5	8	13
起始角	2×90	3×90	3×90	4×90	5×90	6×90	7×90

代码中还用到了如下三角波。



斐波那契螺旋线基于斐波那契数列。它的构造方式基于上一页的正方形，用1/4圆弧(90度)连接×，最终形成一个螺旋线。

斐波那契螺旋线可以用来近似黄金螺旋 (Golden spiral)。

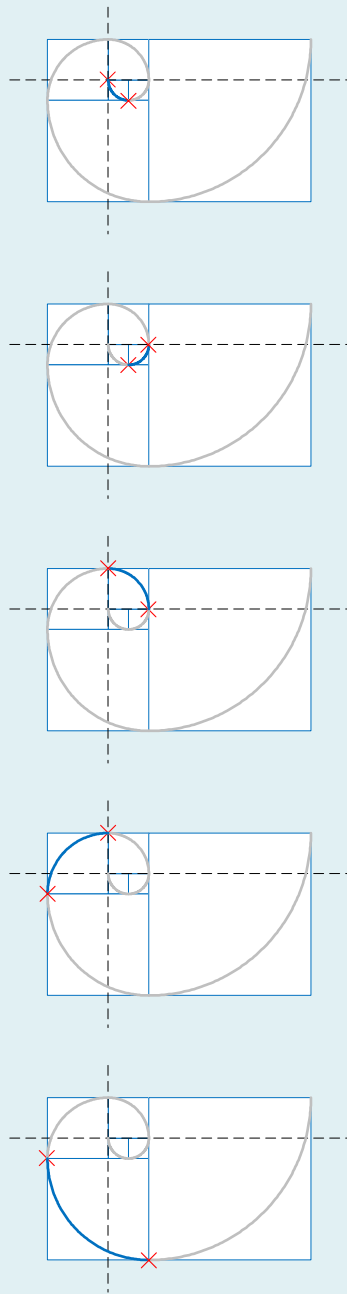


图 2. 可视化斐波那契螺旋线 | Bk2_Ch19_02.ipynb

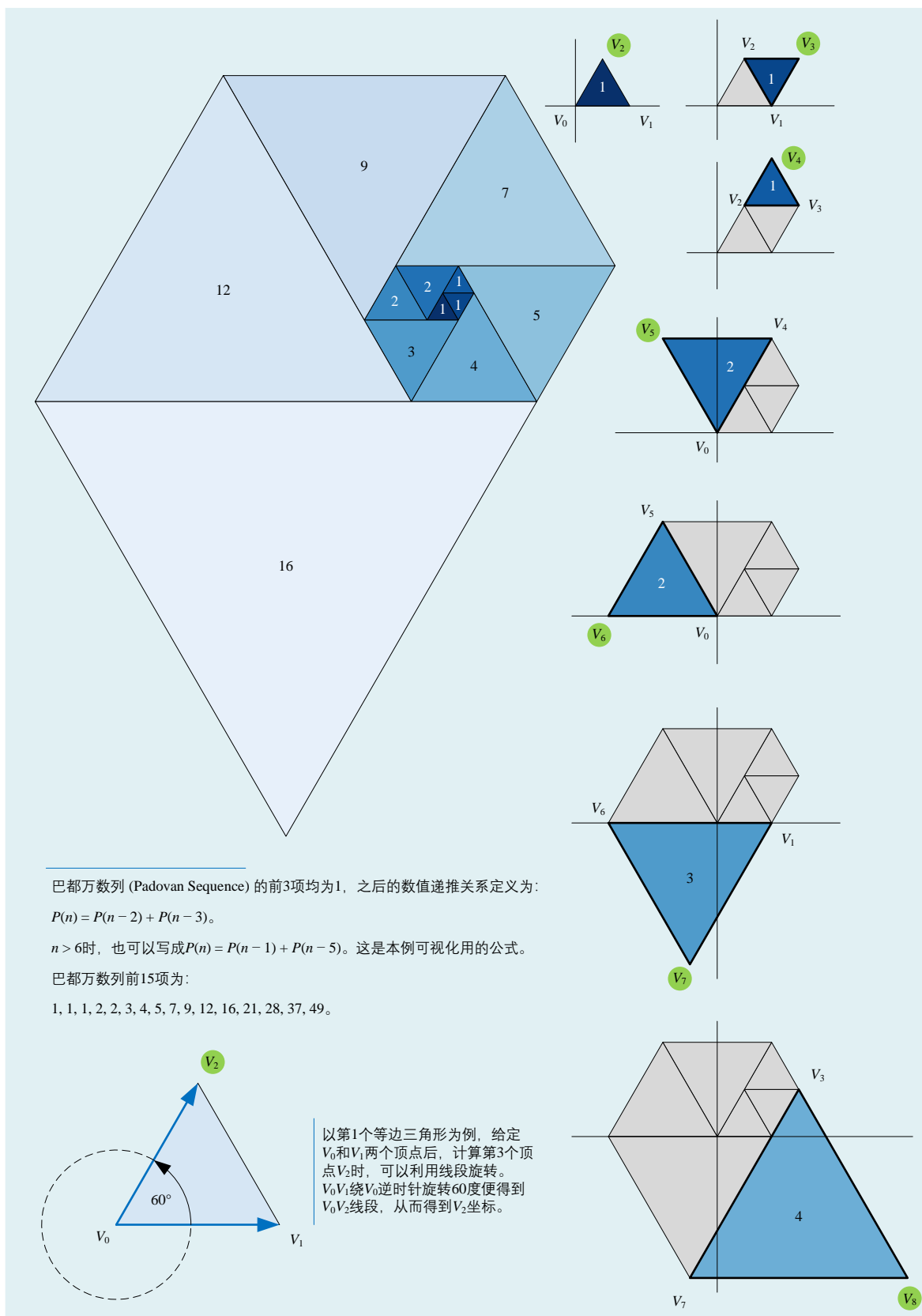


图 3. 可视化巴都万数列 | Bk2_Ch19_03.ipynb

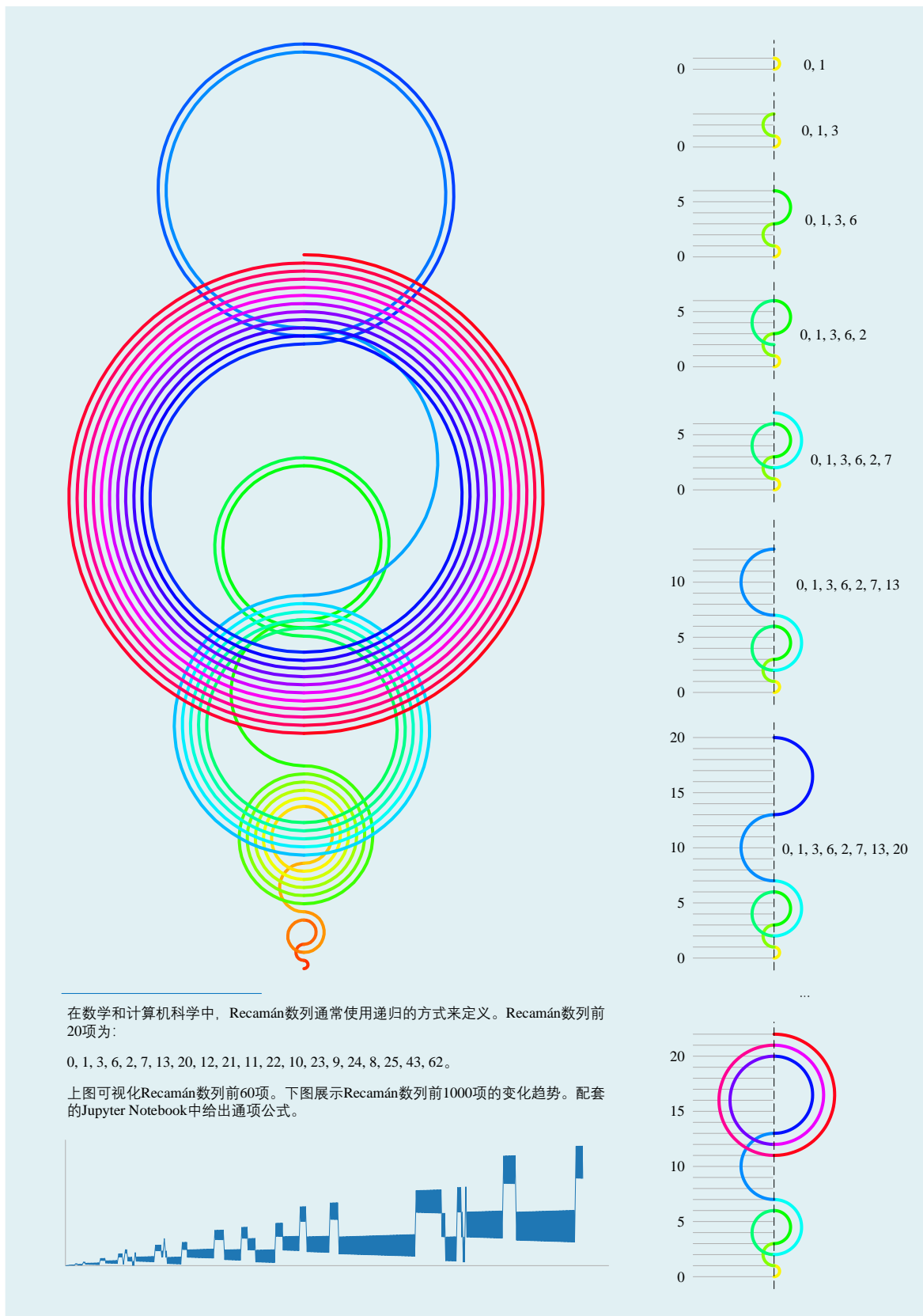
本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

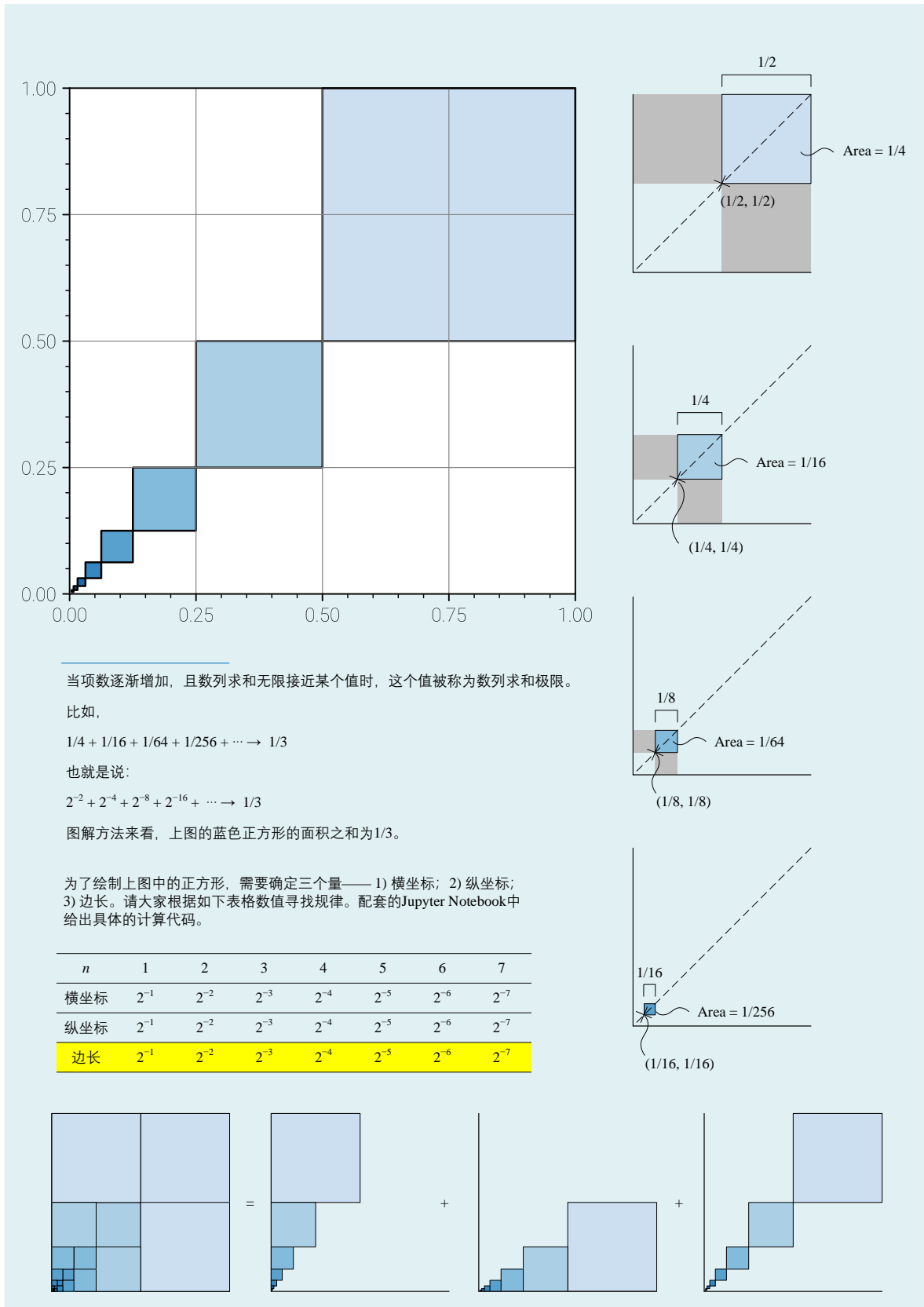
版权归清华大学出版社所有，请勿商用，引用请注明出处。

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

本书配套微课视频均发布在 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

图 4. 可视化雷卡曼数列 | [Bk2_Ch19_04.ipynb](#)



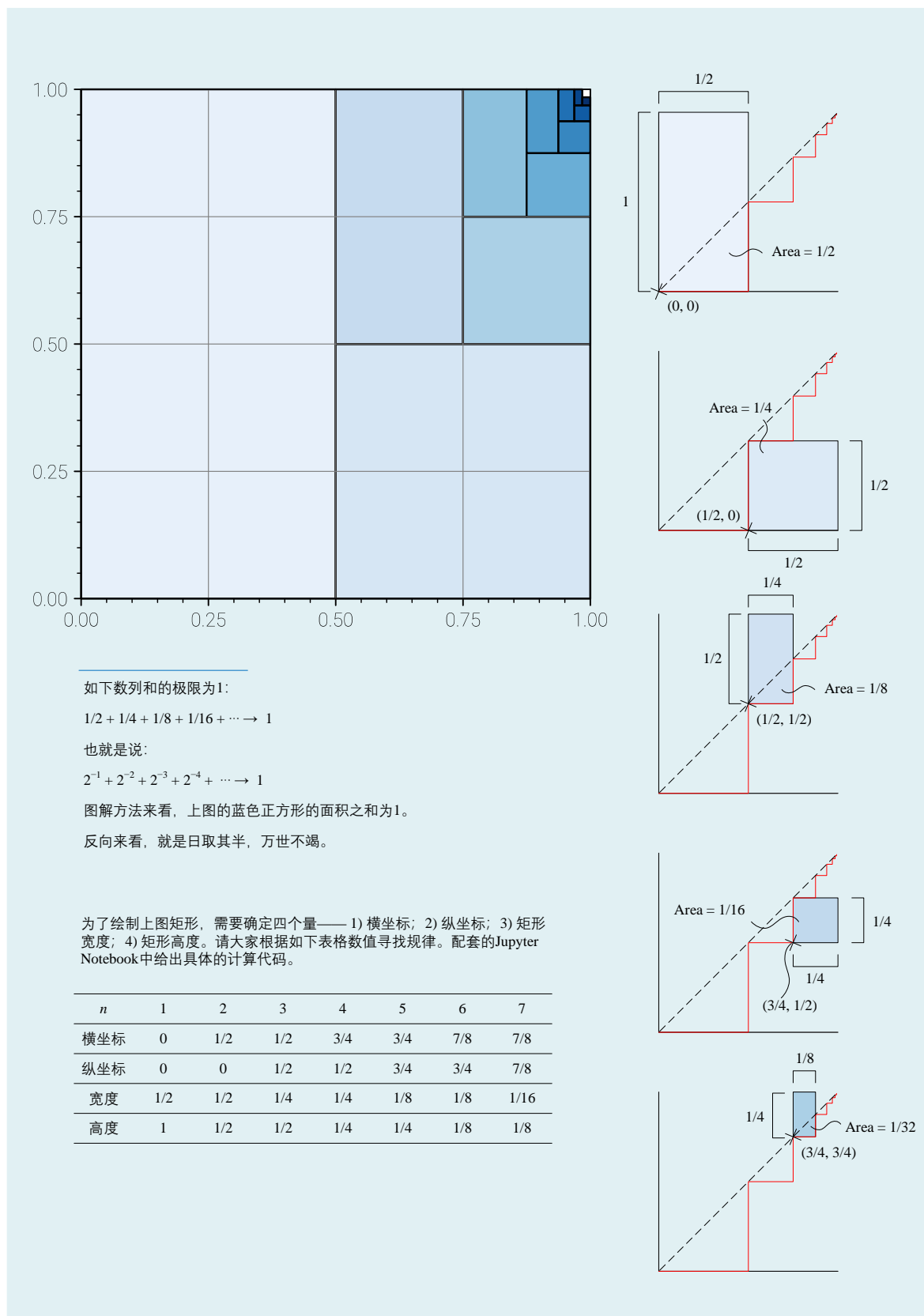


图 6. 可视化数列之和极限, 第 2 组 | Bk2_Ch19_06.ipynb

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>

本书配套微课视频均发布在 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