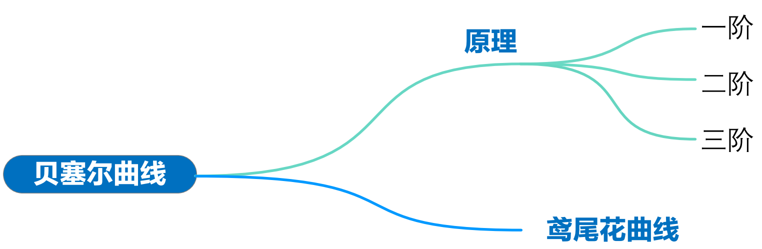
33 Bézier curve 贝塞尔曲线

计算机图形学中特别重要的参数曲线

> 我要扼住命运的喉咙。
>
> ***I shall seize fate by the throat.***
>
> —— 路德维希·范·贝多芬 (Ludwig van Beethoven) | 德意志作曲家、钢琴演奏家 | 1770 ~ 1827

- ◄ `math.factorial()` 计算给定整数的阶乘
- ◄ `numpy.column_stack()` 将两个矩阵按列合并
- ◄ `numpy.interp()` 给定的一维数组上进行线性插值
- ◄ `numpy.random.rand()` 用于生成指定形状的随机数组，随机数服从 0 到 1 之间的均匀分布
- ◄ `random.random()` 生成一个介于 0 到 1 之间的随机浮点数，包括 0 但不包括 1
- ◄ `scipy.interpolate.interp1d()` 一维插值
- ◄ `scipy.interpolate.interp2d()` 二维插值



33.1 贝塞尔曲线

贝塞尔曲线 (Bézier curve) 是一种常用于计算机图形学中的数学曲线。它由法国工程师**皮埃尔·贝塞尔** (Pierre Bézier) 在 19 世纪中叶发明。

贝塞尔曲线最初是为了描述船只的水线曲线。后来，贝塞尔曲线被广泛应用于计算机图形学中，用于绘制平滑曲线，如字体、二维图形和三维模型等。多数矢量图形都离不开贝塞尔曲线。

贝塞尔曲线是由一组控制点和一个阶数确定的曲线。控制点是定义曲线形状的关键点，阶数是定义贝塞尔曲线逼近实际曲线的程度的参数。通常情况下，阶数等于控制点的数量减 1。

贝塞尔曲线的特点是它们具有局部控制性，这意味着通过调整单个控制点的位置，可以轻松地改变曲线的形状。此外，它们也具有平滑的曲线形状和良好的数学性质。Adobe Photoshop、Illustrator 中的钢笔曲线绘图工具实际上使用的便是贝塞尔曲线。

本质上来讲，贝塞尔曲线就是一种插值方法。贝塞尔曲线可以是一阶曲线、二阶曲线、三阶曲线等，其阶数决定了曲线的平滑程度。下面首先介绍一阶贝塞尔曲线原理。

33.2 一阶

一阶曲线由两个控制点组成，形成一条直线。如图 1 所示，简单来说一阶贝塞尔曲线就是两点之间连线。图中 t 代表权重，取值范围为 [0, 1]。t 越大，点 $B(t)$ 距离 P_0 越近，如图中暖色 ×，相当于 P_0 对 $B(t)$ 影响越大。相反，t 越小，点 $B(t)$ 距离 P_1 越近，如图中冷色 ×，相当于 P_1 对 $B(t)$ 影响大。

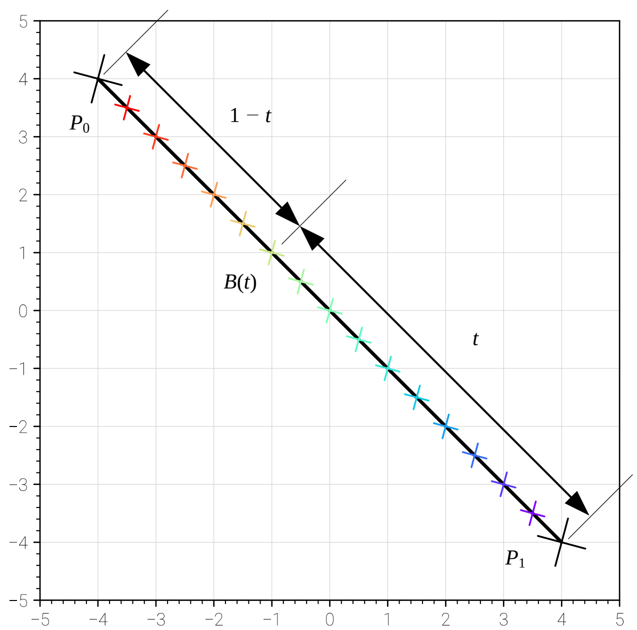


图 1. 一阶贝塞尔曲线原理 | Bk_2_Ch33_01.ipynb

33.3 二阶

二阶贝塞尔曲线由三个控制点组成，形成一条弯曲的曲线。如图 2 所示，P_0 和 P_2 点控制了曲线（黑色线）的两个端点，而 P_1 则决定的曲线的弯曲行为。实际上图 2 中黑色二阶贝塞尔曲线上的每一个点都经历了两组线性插值得到。

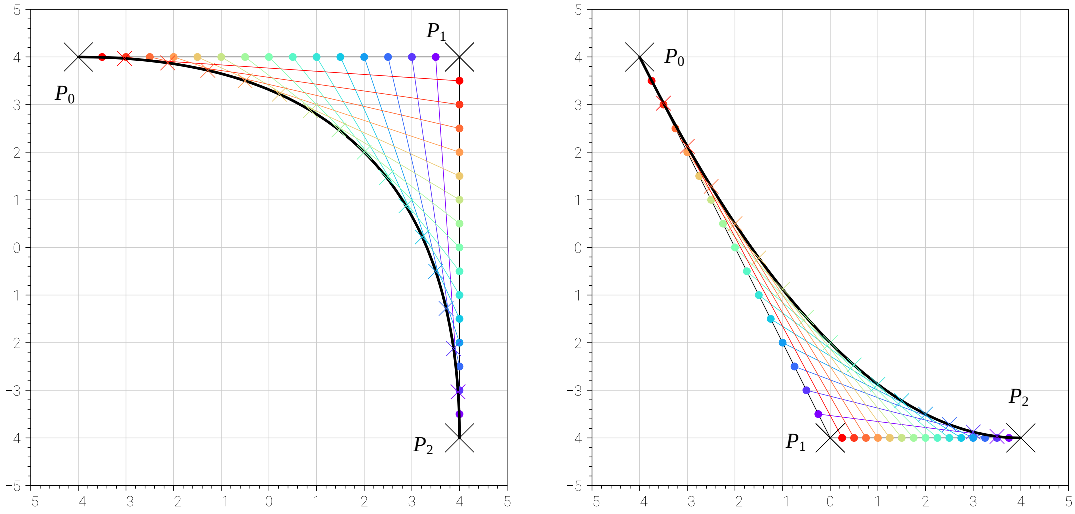


图 2. 二阶贝塞尔曲线原理 | Bk_2_Ch33_02.ipynb

如图 3 所示，设定 t = 13/16，通过第一组线性插值，我们分别得到了 P_0P_1 线段上的 A_0，以及 P_1P_2 线段上的 A_1。然后通过第二组线性插值，我们便得到 A_0A_1 线段上的 B(13/16)。当 t 在 [0, 1] 之间连续取值时，我们便得到了二阶贝塞尔曲线上的一系列点。

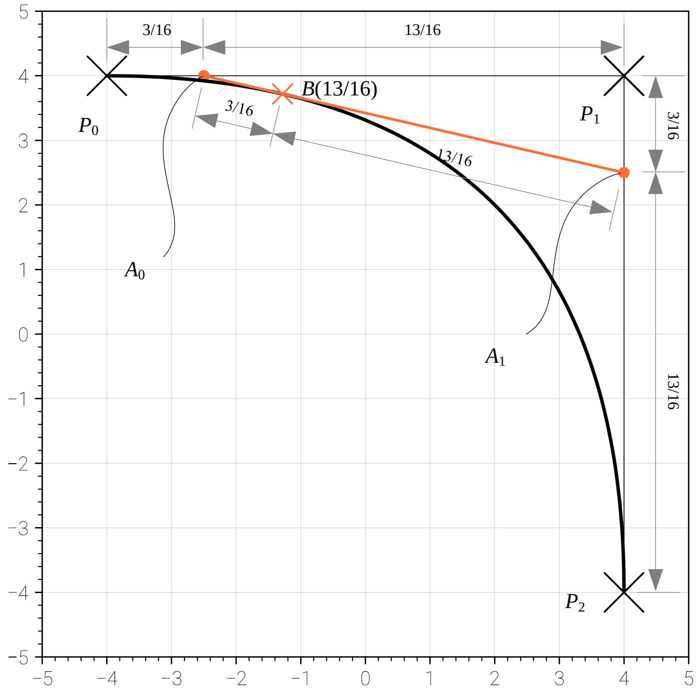


图 3. 二阶贝塞尔曲线原理，以 B(3/16) 为例

图 4 所示为用 Streamlit 搭建的展示二阶贝塞尔曲线原理的 App。

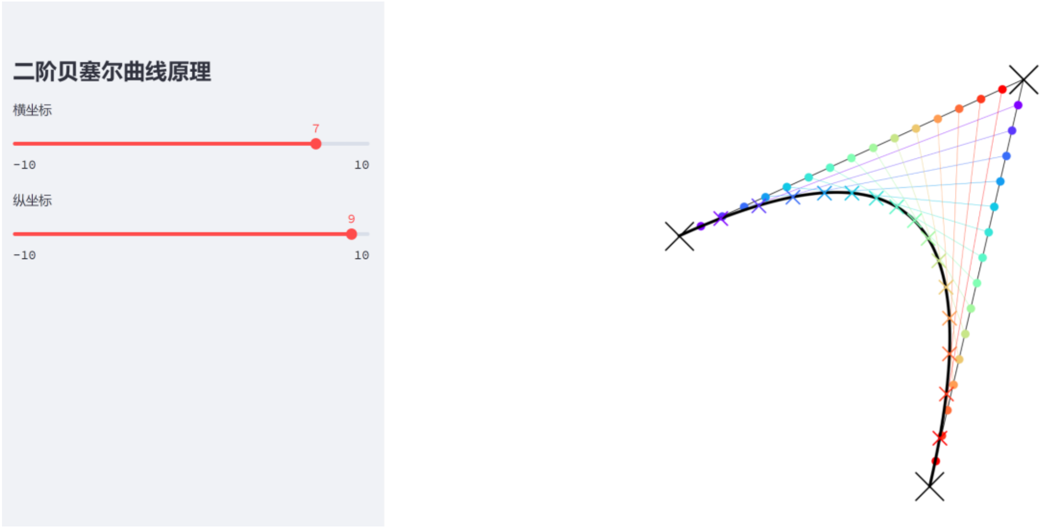


图 4. 展示二阶贝塞尔曲线原理的 App，Streamlit 搭建 | Streamlit_Bezier_2nd_order.py

图 6 给出几个不同的贝塞尔曲线，P_1 点坐标为随机生成。大家可能已经发现，贝塞尔曲线一般不会经过 P_1 点，除非 P_0、P_1、P_2 三点在同一条直线上。

33.4 三阶

图 5 比较一阶、二阶、三阶贝塞尔曲线。三阶贝塞尔曲线由四个控制点组成，形成更加复杂的曲线。如图 5 (c) 所示，P_0 和 P_3 点同样控制了曲线的两个端点，而 P_1 和 P_2 两点决定的曲线的弯曲行为。图 7 所示为一系列三阶贝塞尔曲线，P_1 和 P_2 为随机数。

图 8 所示为一组四阶贝塞尔曲线，曲线的弯曲行为更加复杂。

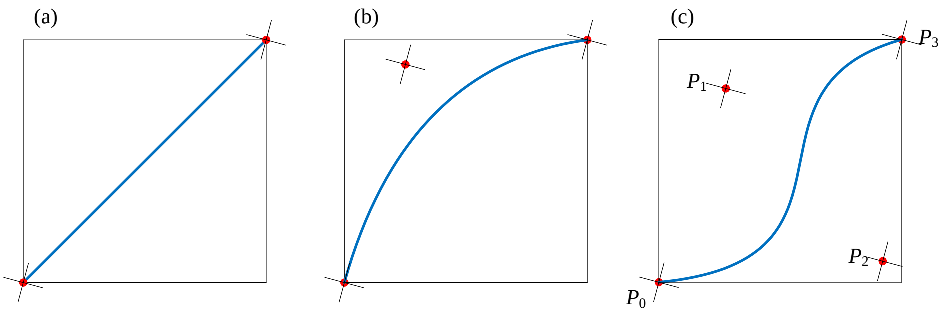


图 5. 贝塞尔曲线原理，比较一阶、二阶、三阶

33.5 三维空间

上述贝塞尔曲线还都仅限于平面，贝塞尔曲线也可以很容易扩展到三维空间。为了可视化贝塞尔曲线，我们把它们放在 RGB 色彩空间中。

图 9 所示为 RGB 色彩空间中 1 ~ 6 阶贝塞尔曲线。图中控制点的表示为 ×，控制点之间的顺序连线为划线。图 9 中这些贝塞尔曲线有一个共同特点，它们的首尾两个控制点分别是黑色 (0, 0, 0)、白色 (1, 1, 1)。其他控制点则均由随机数发生器生成。

图 10 所示 8 阶贝塞尔曲线的 9 个控制点都是随机数发生器生成。

33.5 鸢尾花曲线

图 11 则是采用 Python 复刻的用贝塞尔曲线创作的“鸢尾花曲线”。“鸢尾花曲线”来自于 Oliver Brotherhood 的开源设计创意。请大家尝试用 Streamlit 搭建一个 App 展示不同随机数种子条件下的鸢尾花曲线。

贝塞尔曲线是通过控制点来定义平滑曲线的数学工具。它基于贝塞尔方程，通过插值和逼近生成曲线。在图形设计和计算机图形学中广泛应用，能创建流畅的曲线和复杂的形状。本章利用各种可视化手段展示贝塞尔曲线背后的数学原理，并用贝塞尔曲线创作生成艺术。

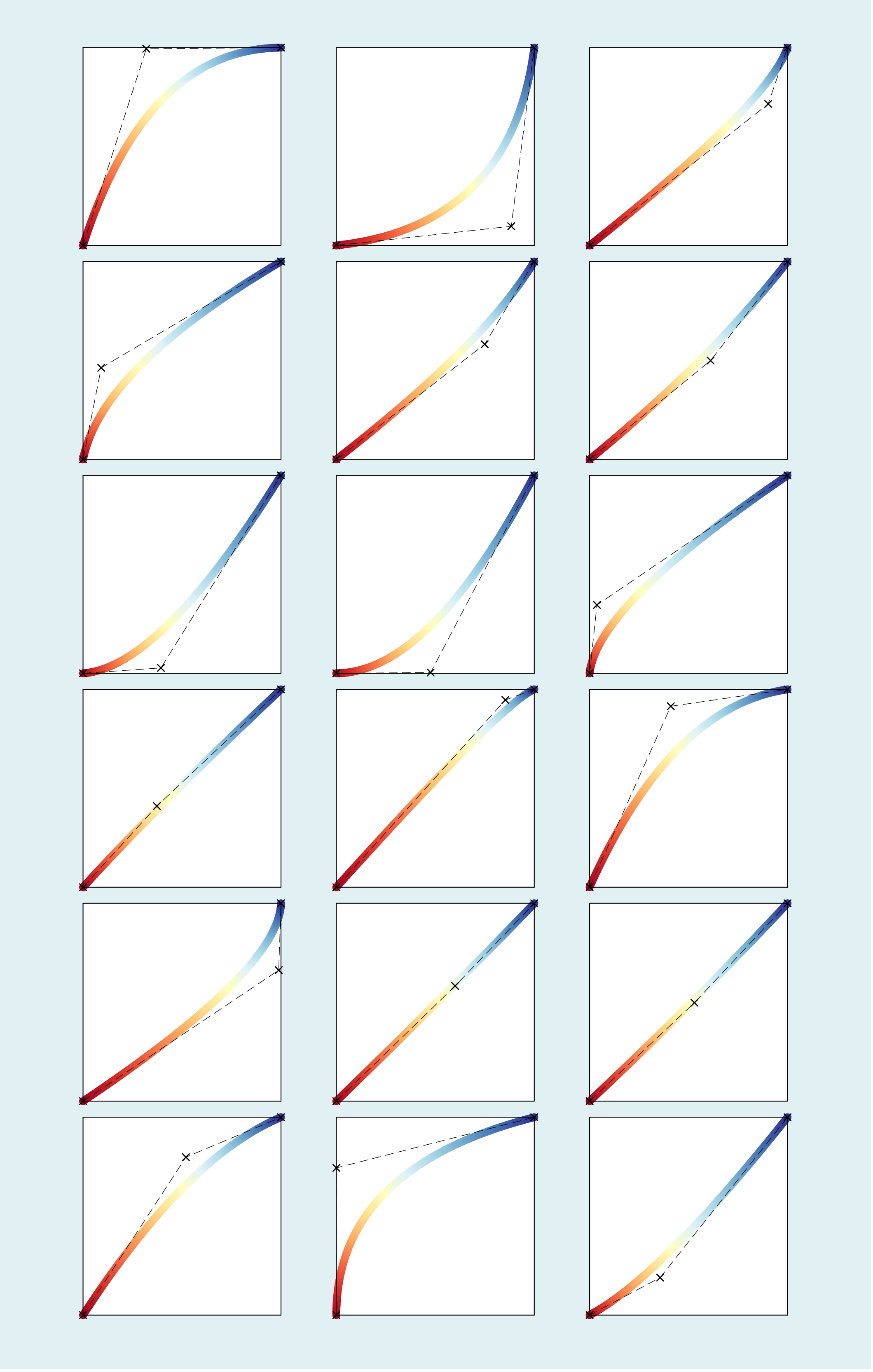


图 6. 二阶贝塞尔曲线 | Bk_2_Ch33_03.ipynb

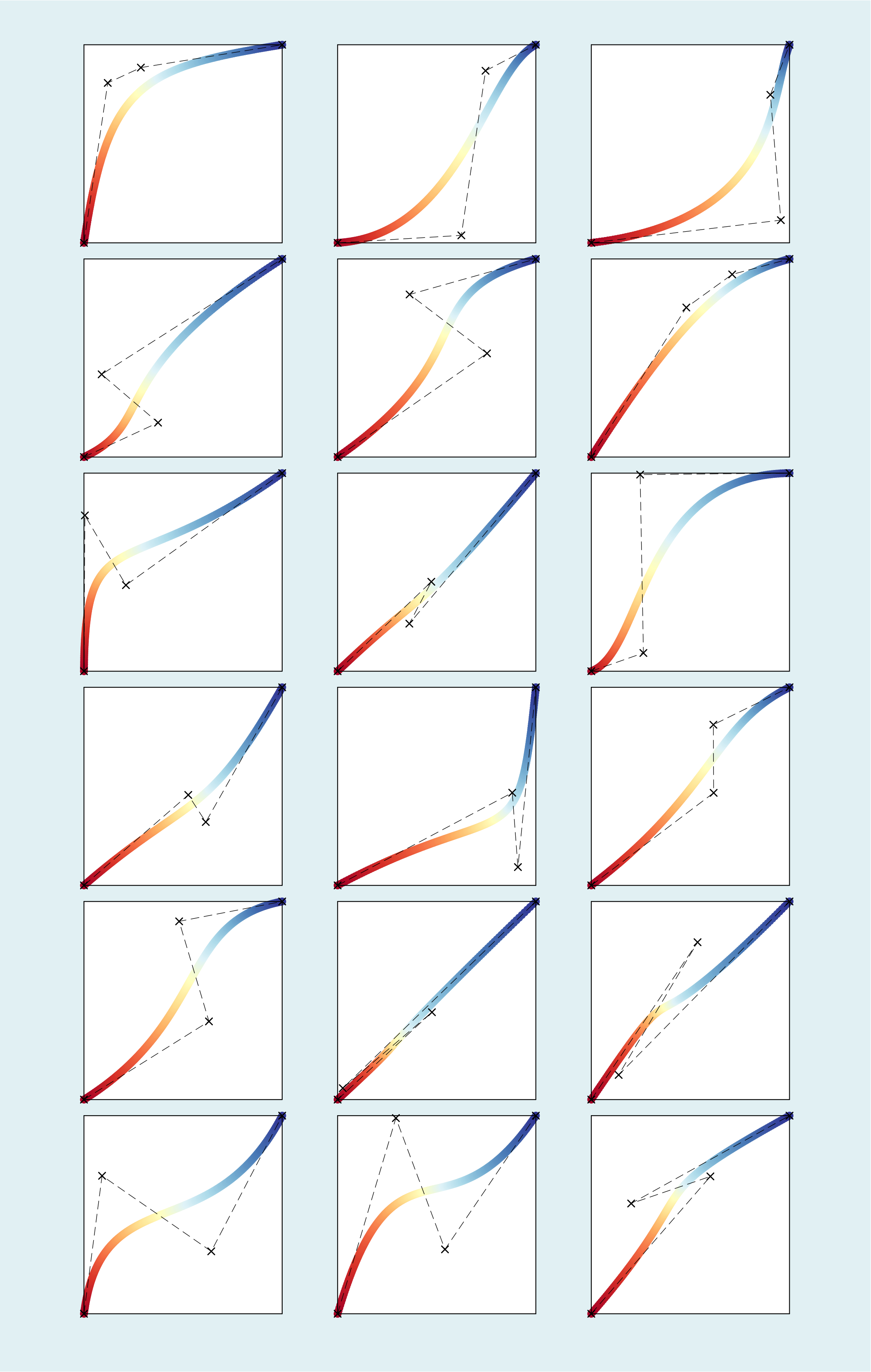


图 7. 三阶贝塞尔曲线 | Bk_2_Ch33_03.ipynb

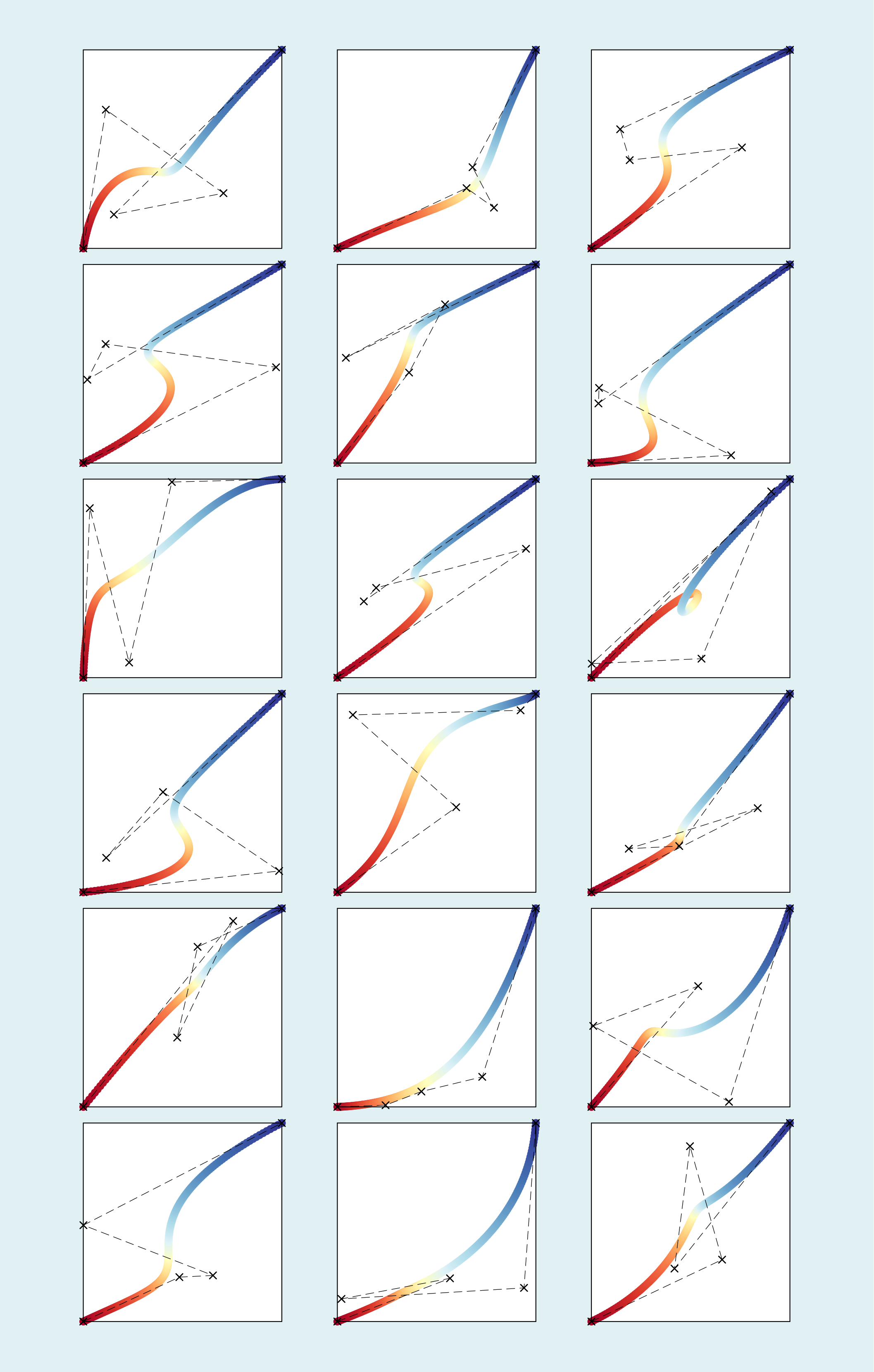


图 8. 四阶贝塞尔曲线 | Bk_2_Ch33_03.ipynb

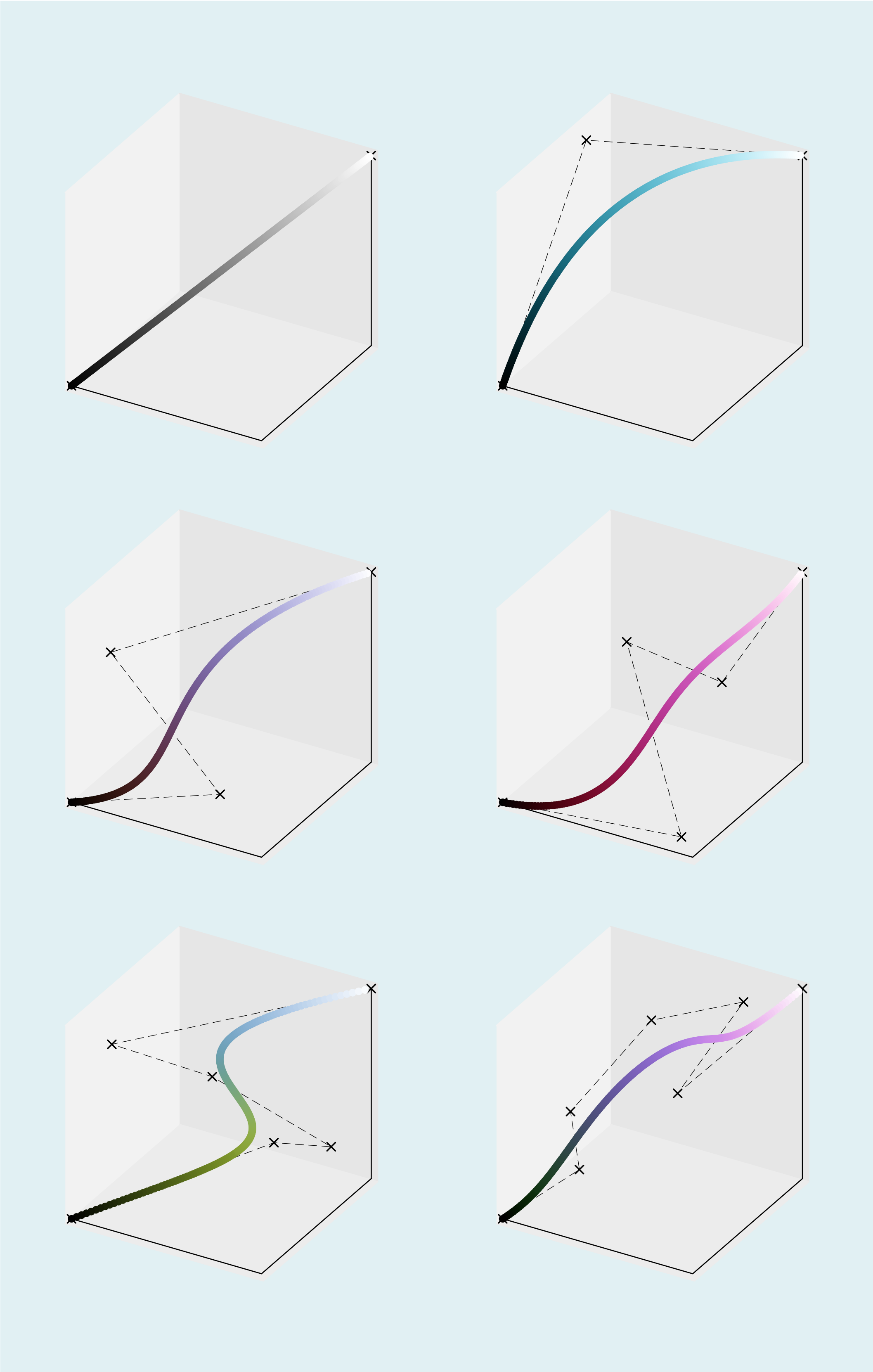


图 9. RGB 色彩空间中的 1 ~ 6 阶贝塞尔曲线 | Bk_2_Ch33_04.ipynb

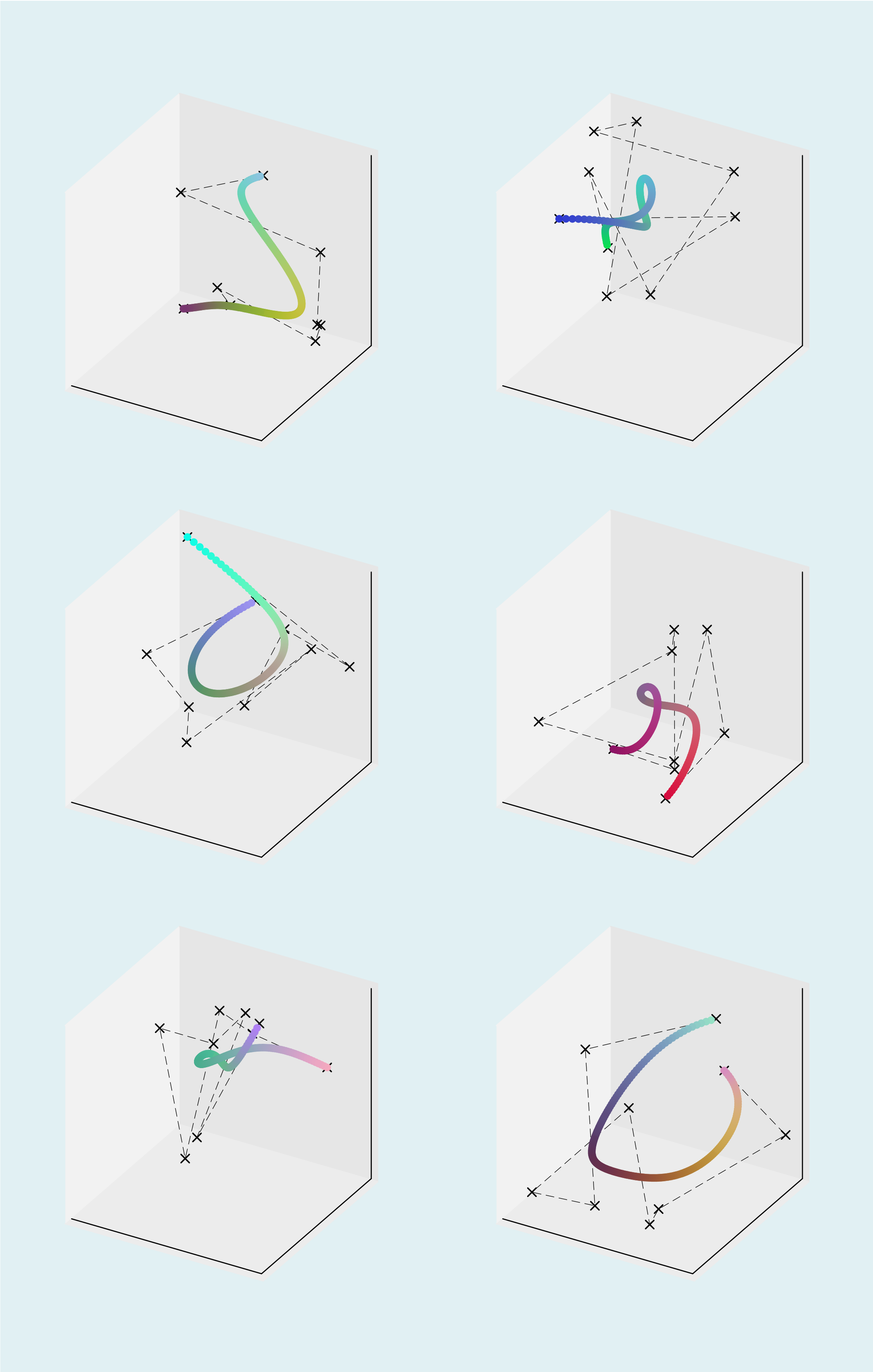


图 10．RGB 色彩空间中的几个 8 阶贝塞尔曲线，9 个控制点均由随机数发生器生成 | Bk_2_Ch33_04.ipynb

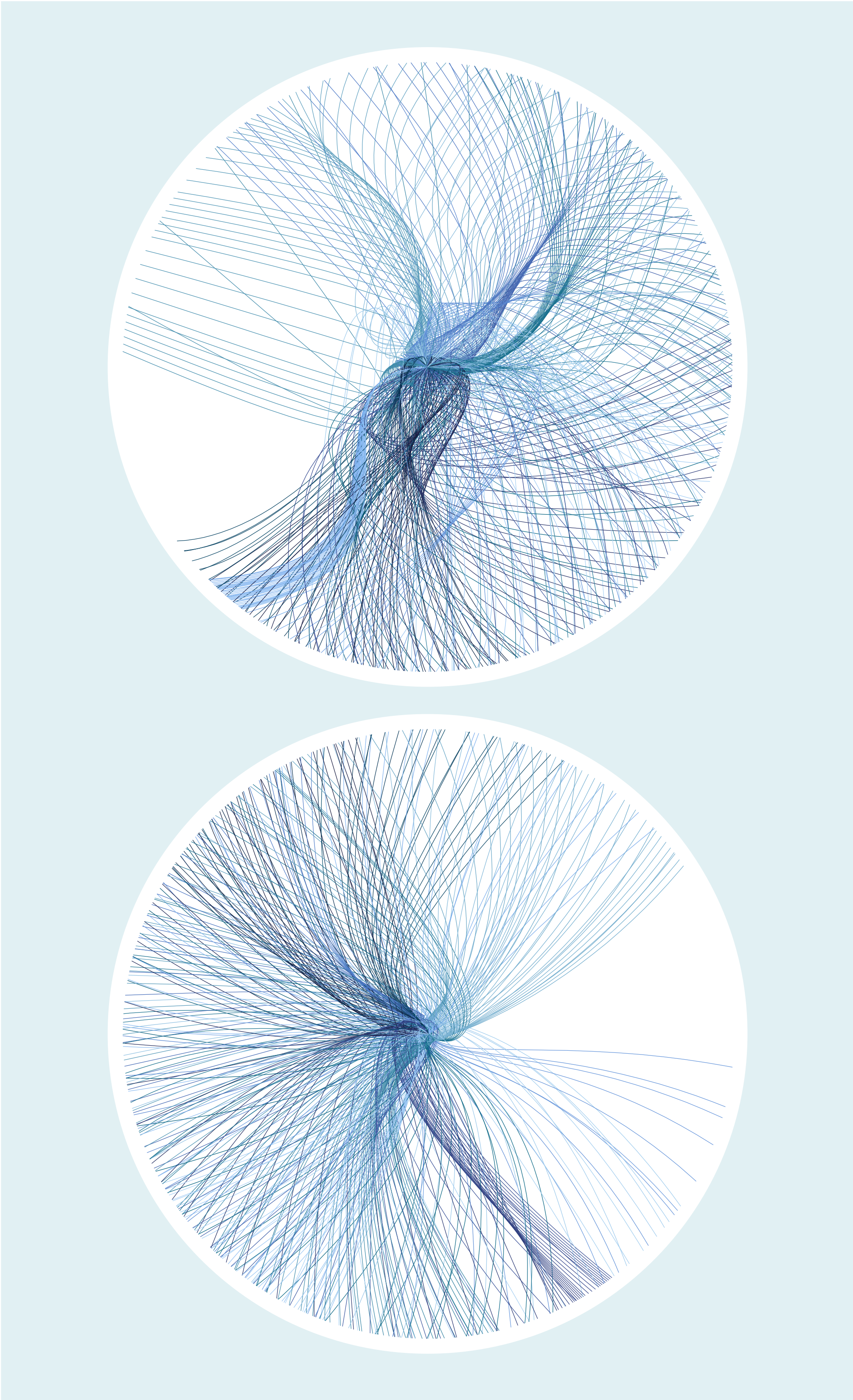


图 11. 用贝塞尔曲线绘制的“鸢尾花曲线” | Bk_2_Ch33_05.ipynb