

13

3D Scatter Plot

三维散点

利用颜色、大小可视化其他特征



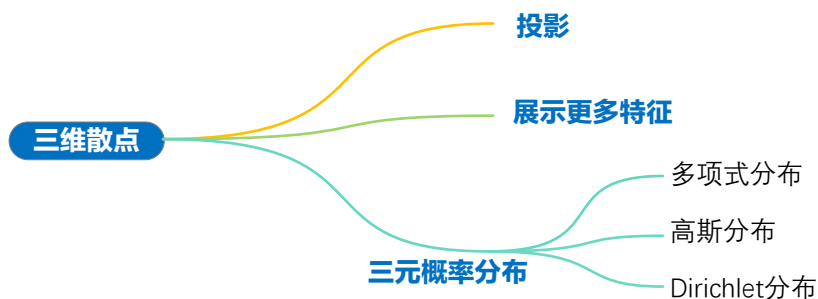
当一扇门关闭时，另一扇门打开；但是我们望眼欲穿、死死紧盯那扇关闭的门，看不到为我们打开的门。

When one door closes, another door opens; but we so often look so long and regretfully upon the closed door, that we do not see the ones which open for us.

—— 亚历山大·贝尔 (Alexander Bell) | 发明家、企业家 | 1847 ~ 1822



- ◀ matplotlib.pyplot.scatter() 绘制散点图
- ◀ numpy.dot() 计算向量标量积。值得注意的是，如果输入为一维数组，numpy.dot() 输出结果为标量积；如果输入为矩阵，numpy.dot() 输出结果为矩阵乘积，相当于矩阵运算符 \odot
- ◀ numpy.linalg.det() 计算行列式值
- ◀ numpy.linalg.inv() 矩阵求逆
- ◀ numpy.meshgrid() 创建网格化数据
- ◀ numpy.reshape() 是一个函数，用于重新重塑一个数组的形状，而不改变其数据内容
- ◀ numpy.where() 根据给定的条件返回输入数组中满足条件的元素的索引或值
- ◀ scipy.stats.dirichlet.pdf() Dirichlet 分布概率密度函数
- ◀ scipy.stats.multinomial.pmf() 多项分布概率质量函数



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

13.1 三维散点

本书前文，大家已经看过用散点可视化 RGB 色彩空间。本章深入聊一下如何用三维散点可视化各种场景。

各种样式的三维散点

我们首先用三维散点图可视化鸢尾花数据，如图 1 所示。其中， x 轴代表花萼长度， y 轴代表花萼宽度， z 轴代表花瓣长度。

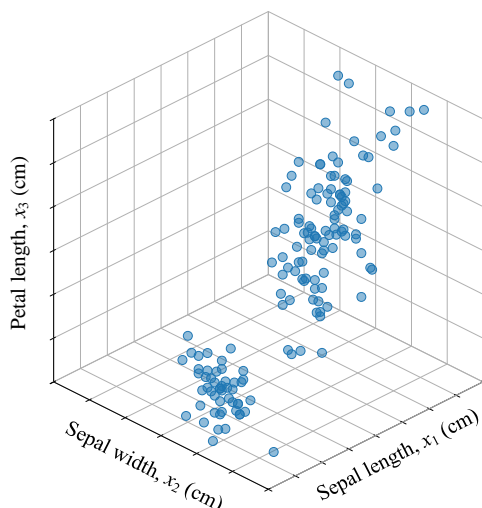



图 1. 三维散点可视化样本数据 |  BK_2_Ch13_01.ipynb

BK_2_Ch19_1.ipynb 绘制本节和下一节散点图，下面首先聊聊代码 1。

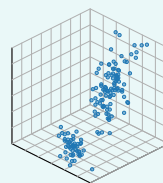
- a 用 `matplotlib.pyplot.figure()`，简作 `plt.figure()`，创建图形对象 `fig`。
- b 用 `add_subplot()` 在 `fig` 上添加三维轴对象，参数设置 `projection='3d'`。
- c 美化三维轴对象，请大家逐行注释。

```

a fig = plt.figure()
b ax = fig.add_subplot(projection='3d')
c ax.scatter(x1, x2, x3)

d ax.set_xlabel('Sepal length, $x_1$ (cm)')
  ax.set_ylabel('Sepal width, $x_2$ (cm)')
  ax.set_zlabel('Petal length, $x_3$ (cm)')
  ax.set_proj_type('ortho')
  ax.view_init(azim=-135, elev=30)
  ax.set_box_aspect([1,1,1])
  ax.w_xaxis.set_pane_color((1.0, 1.0, 1.0, 1.0))
  ax.w_yaxis.set_pane_color((1.0, 1.0, 1.0, 1.0))
  ax.w_zaxis.set_pane_color((1.0, 1.0, 1.0, 1.0))
  ax.set_xlim(4, 8)
  ax.set_ylim(2, 5)
  ax.set_zlim(1, 7)
  ax.set_xticklabels([])
  ax.set_yticklabels([])
  ax.set_zticklabels([])

```



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>

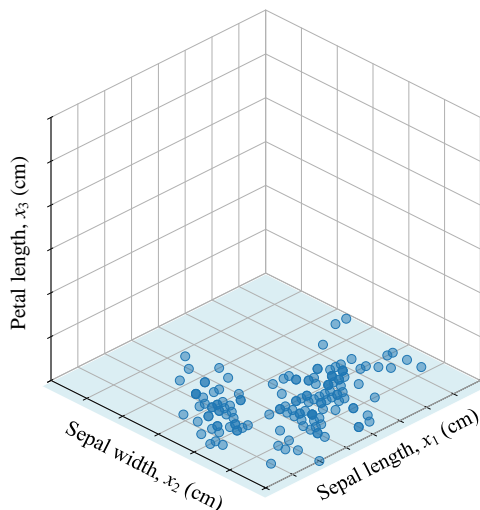
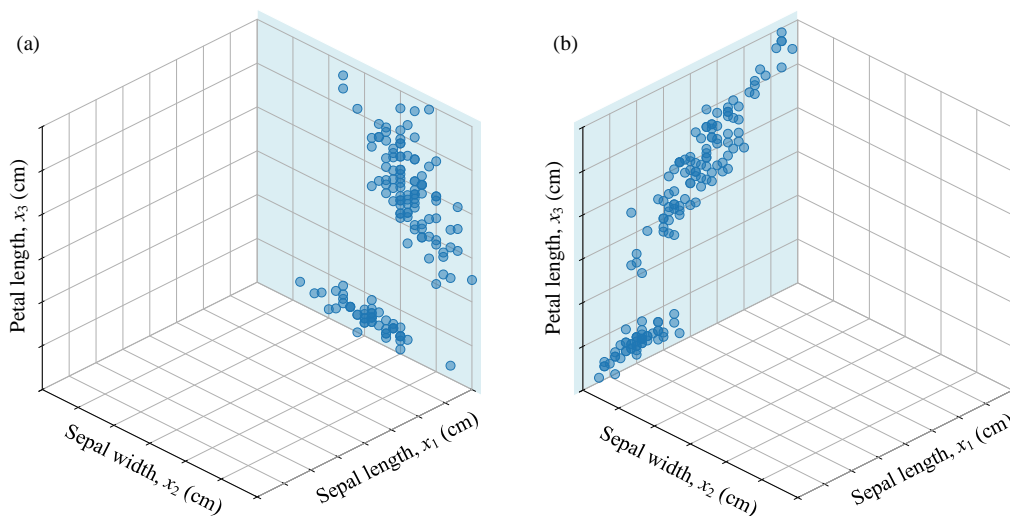
本书配套微课视频均发布在 B 站——生姜 DrGinger: <https://space.bilibili.com/513194466>

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

代码 1. 绘制三维散点图 |  BK_2_Ch13_01.ipynb

在不同平面上的投影

三维散点可以投影到不同平面上。图 2 所示为三维散点投影在 $x_3 = 1$ 平面上，即 $z = 1$ 。图 3 (a) 所示为散点投影在 $x_1 = 8$ 平面上，即 $x = 8$ 。图 3 (b) 所示为散点投影在 $x_2 = 5$ 平面上，即 $y = 5$ 。

图 2. 三维散点投影在 $x_3 = 1$ ($z = 1$)图 3. 三维散点投影在 $x_2 = 5$ ($y = 5$)、 $x_2 = 5$ ($y = 5$)

下面聊聊代码 2。这段代码实际上绘制了三幅散点图，不同的是它们的投影方向各不相同。

a 在三维轴对象 `ax` 上用 `scatter()` 绘制散点图。设定 `zdir='z'` 表示在 z 轴特定高度上绘制散点图。`zs=1` 表示所有散点都将位于 z 轴高度为 1 的平面上。

这个过程相当于将三维散点投影到 $z = 1$ 平面上。

b 使用 `scatter()` 绘制散点图时，设定 `zdir='y'` 和 `zs = 5` 在 $y = 5$ 平面上绘制散点。

c 使用 `scatter()` 绘制散点图时，设定 `zdir='x'` 和 `zs = 8` 在 $x = 8$ 平面上绘制散点。

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。

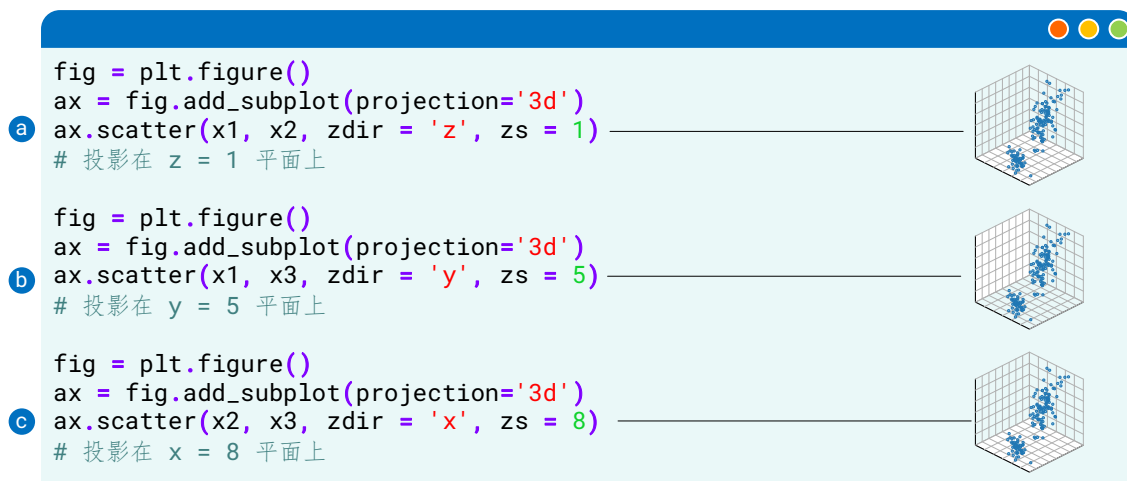
版权归清华大学出版社所有，请勿商用，引用请注明出处。

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

本书后文在绘制网格曲面、三维等高线等图像时，我们还会用到相似的投影方法，请大家注意。



代码 2. 绘制三维散点图，在三个平面上投影 | BK_2_Ch13_01.ipynb

13.2 展示更多特征

Matplotlib

类似平面散点图，我们可以用散点大小、颜色可视化更多特征。图 4 所示为用散点大小可视化鸢尾花花瓣宽度。

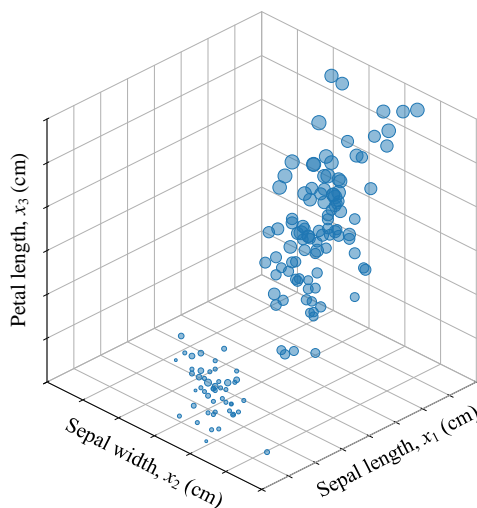


图 4. 用散点大小可视化鸢尾花花瓣宽度

图 5 (a) 所示为用颜色可视化鸢尾花类别。结合图 4、图 5 (a)，我们便得到图 5 (b)。图 5 (b) 可视化鸢尾花四个特征和分类标签。

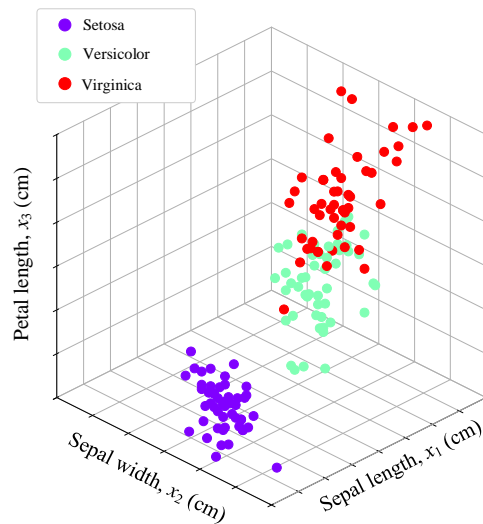


图 5. 用颜色可视化鸢尾花类别

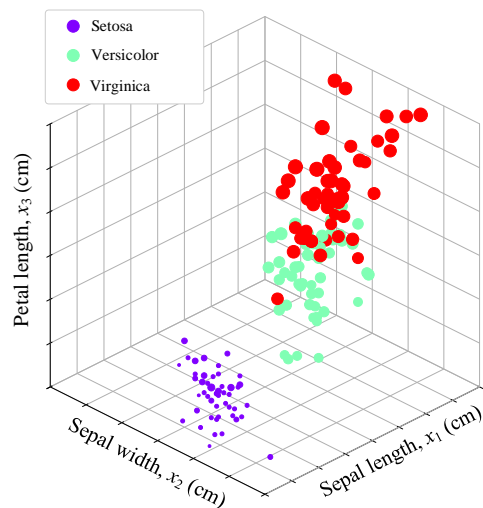


图 6. 同时可视化鸢尾花花萼宽度、类别

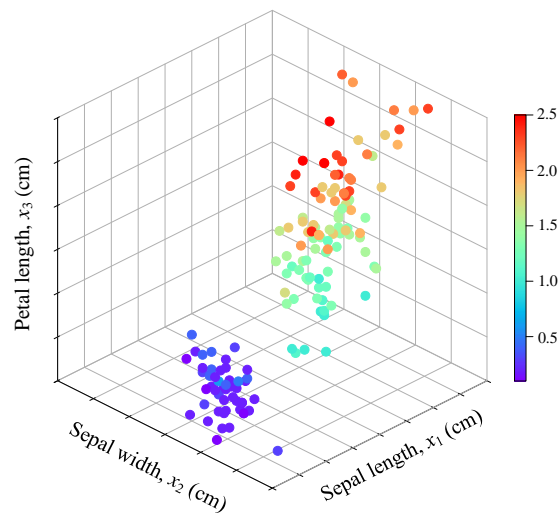


图 7. 用色谱可视化鸢尾花宽度

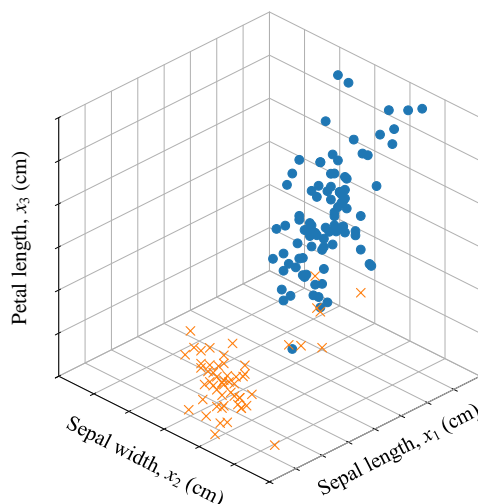


图 8. 用标记类型展示特征

下面聊聊 BK_2_Ch13_01.ipynb 中代码 3 这几句代码。

a 在使用 `scatter()` 绘制三维散点图时，设置 `s=x4*20` 表示每个散点的大小由 `x4` 决定，并且通过乘以 20 进行缩放。

b 中，参数 `c=labels` 表示使用 `labels` 数组的值来确定每个散点的颜色。`labels` 是一个包含鸢尾花样本数据标签信息的数组。`cmap=rainbow` 指定了颜色映射，它是一个从标签值到颜色的映射关系。

c 则结合 **a** 和 **b**。

d 通过设置 `c = x4`，用渐变颜色映射展示花瓣长度这个特征样本数值。

e 实际上是两个散点图，满足 `x4 > 1` 的用 `marker='o'` 来展示散点；满足 `x4 <= 1` 的数据则用 `marker='x'` 来展示。

⚠ 注意，目前 Seaborn 只能绘制二维散点图，还不支持三维散点图。

```

# 利用散点大小展示第四个特征
fig = plt.figure()
ax = fig.add_subplot(projection='3d')
ax.scatter(x1, x2, x3,
           s = x4*20)


# 利用颜色展示分类标签
fig = plt.figure()
ax = fig.add_subplot(projection='3d')
scatter_h = ax.scatter(x1, x2, x3,
                      c = labels,
                      cmap=rainbow)

# 颜色分类 + 散点大小
fig = plt.figure()
ax = fig.add_subplot(projection='3d')
ax.scatter(x1, x2, x3,
          s = x4*20,
          c = labels,
          cmap=rainbow)

# 利用色谱展示第四维特征
fig = plt.figure()
ax = fig.add_subplot(projection='3d')
scatter_plot = ax.scatter(x1, x2, x3,
                        c = x4,
                        cmap=rainbow)

# 用标记类型展示特征
fig = plt.figure()
ax = fig.add_subplot(projection='3d')
ax.scatter(x1[x4>1], x2[x4>1], x3[x4>1],
          marker='o')
ax.scatter(x1[x4<=1], x2[x4<=1], x3[x4<=1],
          marker='x')

```

代码 3. 绘制三维散点图，用颜色、大小展示更多特征 |  BK_2_Ch13_01.ipynb

Plotly

在 Plotly 中，我们可以用 `plotly.express.scatter_3d()` 绘制三维散点图。这个可视化工具具有很多好处。第一，图像就有可交互性；第二，展示标签很方便，特别是机器学习中展示样本标签时；第三，函数直接支持 Pandas DataFrame 类型数据。

图 9 所示为利用 Plotly 绘制的三维散点图。

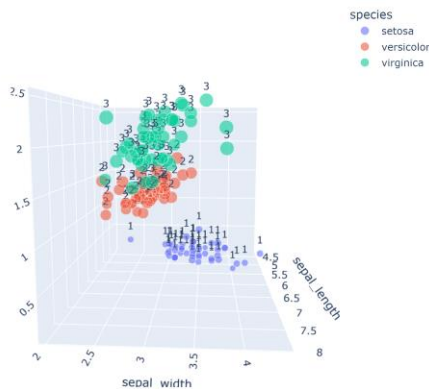


图 9. 用 Plotly 绘制三维散点图

下面聊聊代码 4。

a 利用 `plotly.express.data.iris()`，简作 `px.data.iris()`，来导入鸢尾花数据。

b 用 `plotly.express.scatter_3d()`，简作 `px.scatter_3d()`，绘制散点图。

`df` 是一个保存鸢尾花数据的 Pandas DataFrame。

参数 `x='sepal_length'`、`y='sepal_width'`、`z='petal_width'` 指定三维散点图中使用的三个坐标轴，分别是花萼长度、花萼宽度和花瓣宽度。

参数 `color='species'` 表示按照鸢尾花的种类来给每个散点着色。不同的物种将有不同的颜色。

参数 `size='petal_length'` 决定了每个散点的大小，即散点大小由花瓣长度决定。

参数 `text='species_id'` 指定每个散点上显示的文本信息，这里显示的是鸢尾花的类别数字标识。

参数 `size_max=28` 规定了散点的最大大小，设置为 28。

参数 `opacity=0.58` 控制了散点的透明度，设置为 0.58。

```
import plotly.express as px
# 导入数据
a df = px.data.iris()
df.head()

# 用三维散点可视化
b fig = px.scatter_3d(df,
                      x='sepal_length',
                      y='sepal_width',
                      z='petal_width',
                      color='species',
                      size='petal_length',
                      text='species_id',
                      size_max=28, opacity=0.58)
fig.update_layout(autosize=False, width=600, height=600)
fig.show()
```

代码 4. 用 Plotly 绘制三维散点图 | BK_2_Ch13_01.ipynb

13.3 可视化三元概率分布

至此，我们已经掌握了很多可视化一元、二元概率分布的绘图方案。本节要介绍如何用三维散点展示三元概率分布。本书后文还会介绍更多展示三元概率分布的可视化方案。

多项分布

图 15 所示为用三维散点可视化多项分布。**多项分布 (Multinomial Distribution)** 是一种离散型概率分布，用于描述在多项试验中各个可能结果出现次数的概率分布。多项试验是指在一个试验中，每次试验有多个可能的结果，每个结果出现的概率是固定的。

图 15 所示的多项分布中， x_1 、 x_2 、 x_3 的取值范围为 $[0, 20]$ 区间内的整数。我们用散点的颜色代表多项分布的概率质量值。

《统计至简》第 5 章将讲解多项分布。



Jupyter 笔记 BK_2_Ch13_02.ipynb 绘制图 15，请大家自行分析代码。

高斯分布

三元高斯分布 (trivariate Gaussian distribution) 概率密度函数本质上是四维数据。如图 16 所示，分层散点图可以可视化三元高斯分布。打个比方，如图 10 所示，这种可视化方法相当于断层扫描来观察不同截面数据。更通俗地说，就好比用不同刀法切豆腐。本书后文还会用“切豆腐”可视化更多数据，请大家格外注意。

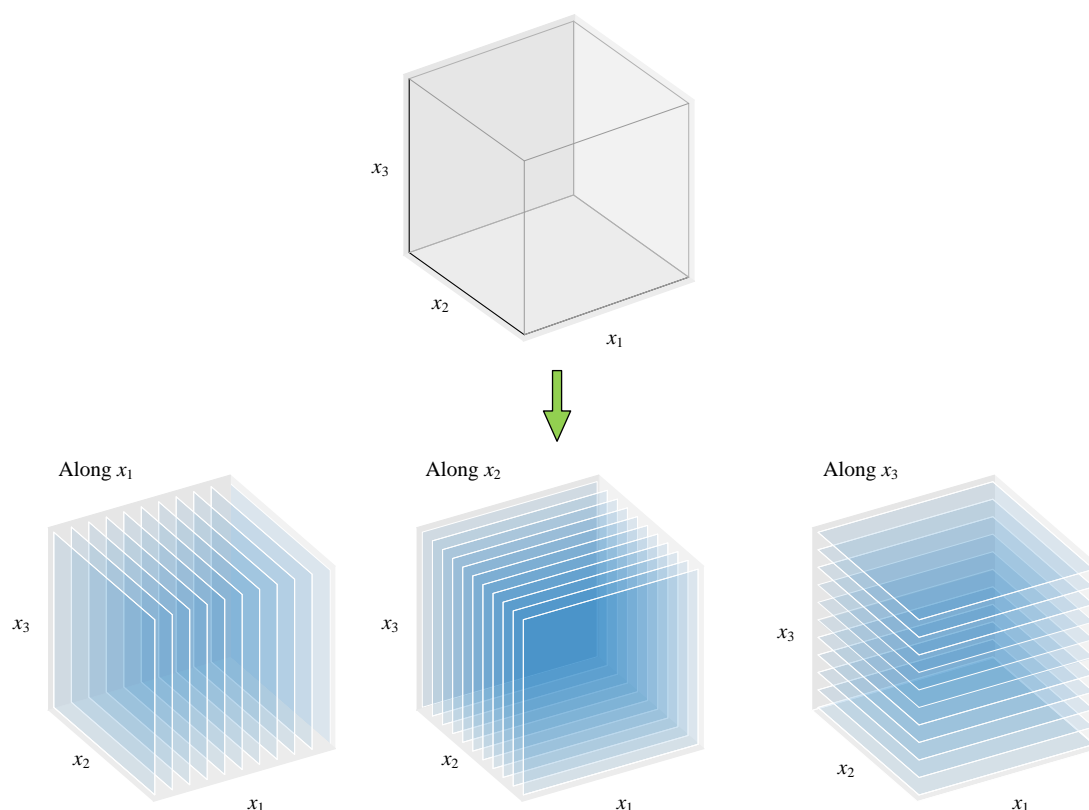


图 10. 三种不同刀法“切豆腐”

▲ 注意，对于三元高斯分布 PDF， x_1 、 x_2 、 x_3 的取值范围均为 $(-\infty, \infty)$ 。

本书后文会用分层等高线可视化三元高斯分布的 PDF。

➡ 鸢尾花书《矩阵力量》第 20 章会专门从线性代数运算角度展开讲解如何理解多元高斯分布 PDF。《统计至简》第 11 章将专门讲解多元高斯分布。

Jupyter 笔记 BK_2_Ch13_03.ipynb 绘制图 16，我们有必要聊聊代码 5 和代码 6 这两个自定义函数。

读过鸢尾花书《编程不难》的同学，对马氏距离应该不陌生。大家应该已经清楚，平面上，欧氏距离等高线为正圆，而马氏距离等高线多为旋转椭圆。图 11 来自《编程不难》第 31 章，我们用这幅图中数据讲过**主成分分析** (Principal Component Analysis, PCA) 这种降维算法。图 11 中的等高线就是马氏距离。很容易发现，马氏距离考虑了数据分布的形态。

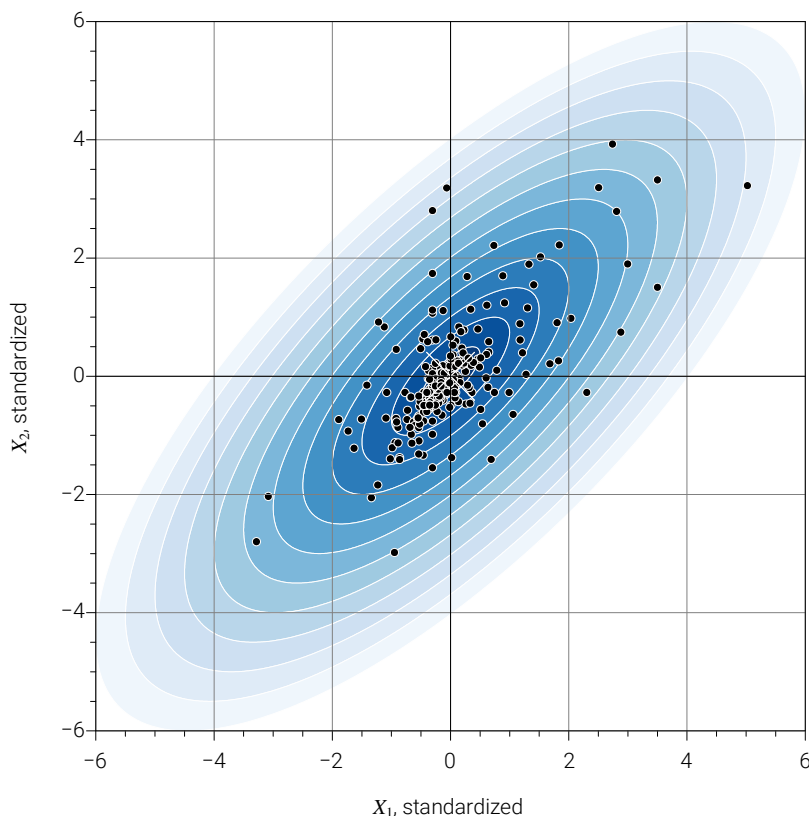


图 11. 标准化数据的散点图，等高线为马氏距离

首先看代码 5，它用来计算马氏距离。我们借助图 12 和图 13 来帮助我们理解马氏距离运算过程。

- a 首先完成中心化；从几何角度来看，这就是平移。
- b 计算协方差矩阵 Σ 的逆，得到 Σ^{-1} 。大家想要理解 Σ^{-1} 到底起到怎样作用的话，就要移步《矩阵力量》深入学习各种线性代数工具了。

对于单一坐标点 $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$ (列向量)，c 计算马氏距离平方 $d^2 = (\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu})$ ，对应的计算过

程如图 12 所示。

对了一组坐标点，为了方便运算向量化，我们采用了图 13。

在图 13 中，矩阵 X 是 n 个行向量 $(x - \mu)^T$ 构成的矩阵。换个角度来看， X 的每一行都是一个（中心化）坐标点；而转置后 X^T 的每一列代表一个坐标点。对于矩阵 X ，**c** 计算的结果为 $n \times n$ 方阵。如图 14 所示，这个方阵主对角线上的元素才是我们想要的马氏距离平方。

- d** 提取主对角线元素，即马氏距离平方。
- e** 开平方得到一组马氏距离；后文代码还会用 `numpy.reshape()` 将其重塑为合适的形状，以便后续可视化。

本书后文在介绍如何可视化**瑞利商** (Rayleigh quotient) 时，也会用到类似运算，请大家务必掌握。此外，本书后文还会可视化包括马氏距离在内的其他距离度量。

```
def Mahal_d(Mu, Sigma, x):
    # 计算马哈距离

    # 中心化, mu为质心
    x_demeaned = x - Mu

    # 协方差矩阵求逆
    inv_covmat = np.linalg.inv(Sigma)

    # 计算马氏距离平方
    mahal_sq = x_demeaned @ inv_covmat @ x_demeaned.T

    # 仅保留对角线元素
    mahal_sq = np.diag(mahal_sq)

    # 对角线元素开平方，得到马氏距离
    mahal_d = np.sqrt(mahal_sq)

    return mahal_d
```

代码 5. 自定义函数计算马氏距离 | BK_2_Ch13_03.ipynb

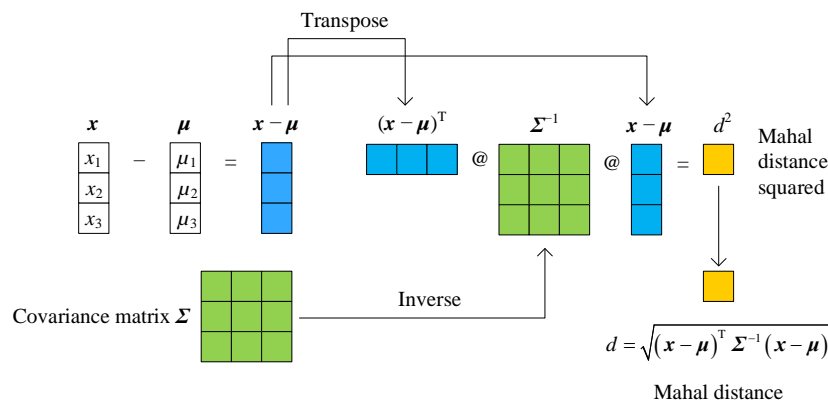


图 12. 计算单一点马氏距离过程

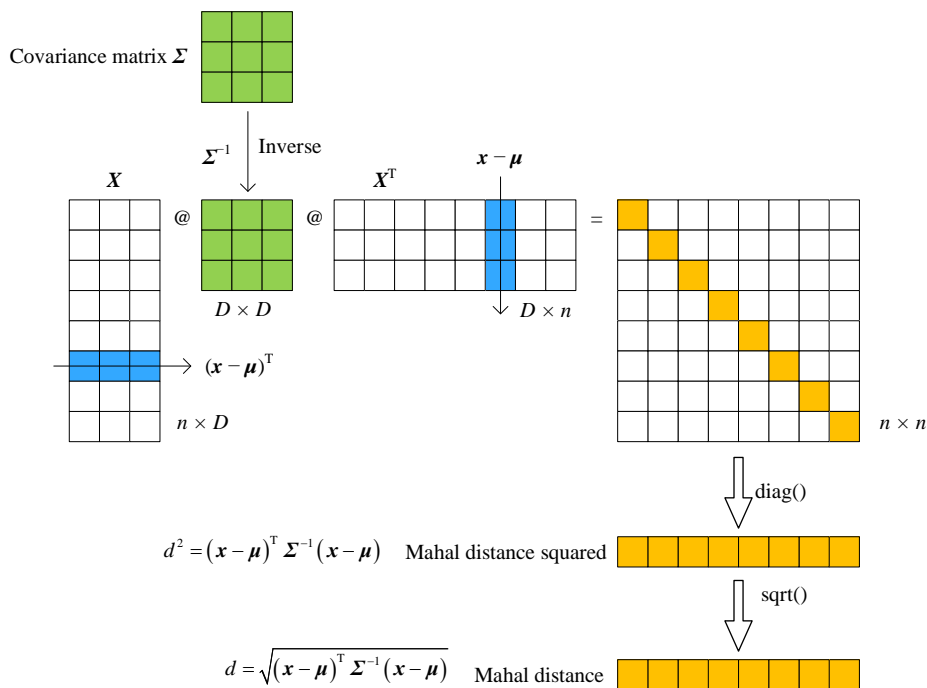


图 13. 计算一组坐标点的马氏距离

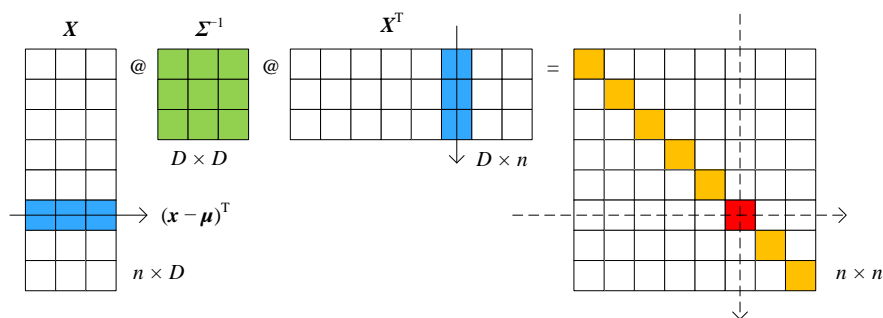


图 14. 主对角线上的元素为马氏距离平方

代码 6 自定义函数计算马氏距离，下面聊聊其中主要语句。

- 计算缩放因子 $\sqrt{|\Sigma|}$ 。| Σ | 表示对协方差矩阵计算**行列式** (determinant)。几何角度来看，| Σ | 和几何变换过程的缩放有关。鸢尾花书《矩阵力量》第 20 章将专门介绍这个知识点。
- 计算缩放因子 $\sqrt{(2\pi)^D}$ 。其中，本例中 $D = 3$ ，代表三元高斯分布。 $\sqrt{(2\pi)^D}$ 和高斯函数积分有关，这个因子完成概率归一化。《数学要素》会专门介绍高斯函数和高斯函数积分。而概率归一化这个知识点将在《统计至简》中讲解。
- 利用高斯函数将马氏距离转化为亲近度，《矩阵力量》第 20 章也会介绍这个知识点。
- 将算式各个部分整合起来计算多元高斯分布的概率密度函数 PDF。

这两段代码看上去简单，但是每一句背后都是一个个数学工具在支撑运算。

这个例子再次告诉我们，“调包”是不够的，仅仅会码代码也是不够的，数学、逻辑这些内核的工具才是知识的内核。

```
def Mahal_d_2_pdf(d, Sigma):
    # 将马氏距离转化为概率密度

    # 计算第一个缩放因子，和协方差行列式有关
    a scale_1 = np.sqrt(np.linalg.det(Sigma))


    # 计算第二个缩放因子，和高斯函数有关；D = 3，三元高斯分布
    b scale_2 = (2*np.pi)**(3/2)

    # 高斯函数，马氏距离转为亲近度
    c gaussian = np.exp(-d**2/2)

    # 完成缩放，得到概率密度值
    d pdf = gaussian/scale_1/scale_2

    return pdf
```

$$f_x(x) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^D} \sqrt{|\boldsymbol{\Sigma}|}}$$

$$= \frac{1}{\sqrt{|\boldsymbol{\Sigma}|}} \frac{1}{(2\pi)^{D/2}} \exp\left(-\frac{1}{2}d^2\right)$$
代码 6. 自定义函数计算马氏距离 |  BK_2_Ch13_03.ipynb

Dirichlet 分布

Dirichlet 分布是一种概率分布，用于描述多维随机变量的概率分布。Dirichlet 分布通常用于处理多元分类和多元回归问题，是多项分布的共轭先验分布。

Dirichlet 分布的定义域是 D 维单位超立方体，即所有分量都在 $[0, 1]$ 之间且它们之和等于 1；也就是说这些散点都在 $\theta_1 + \theta_2 + \theta_3 = 1$ 平面上。如图 17 所示， θ_1 、 θ_2 、 θ_3 的取值范围为 $[0, 1]$ 实数。

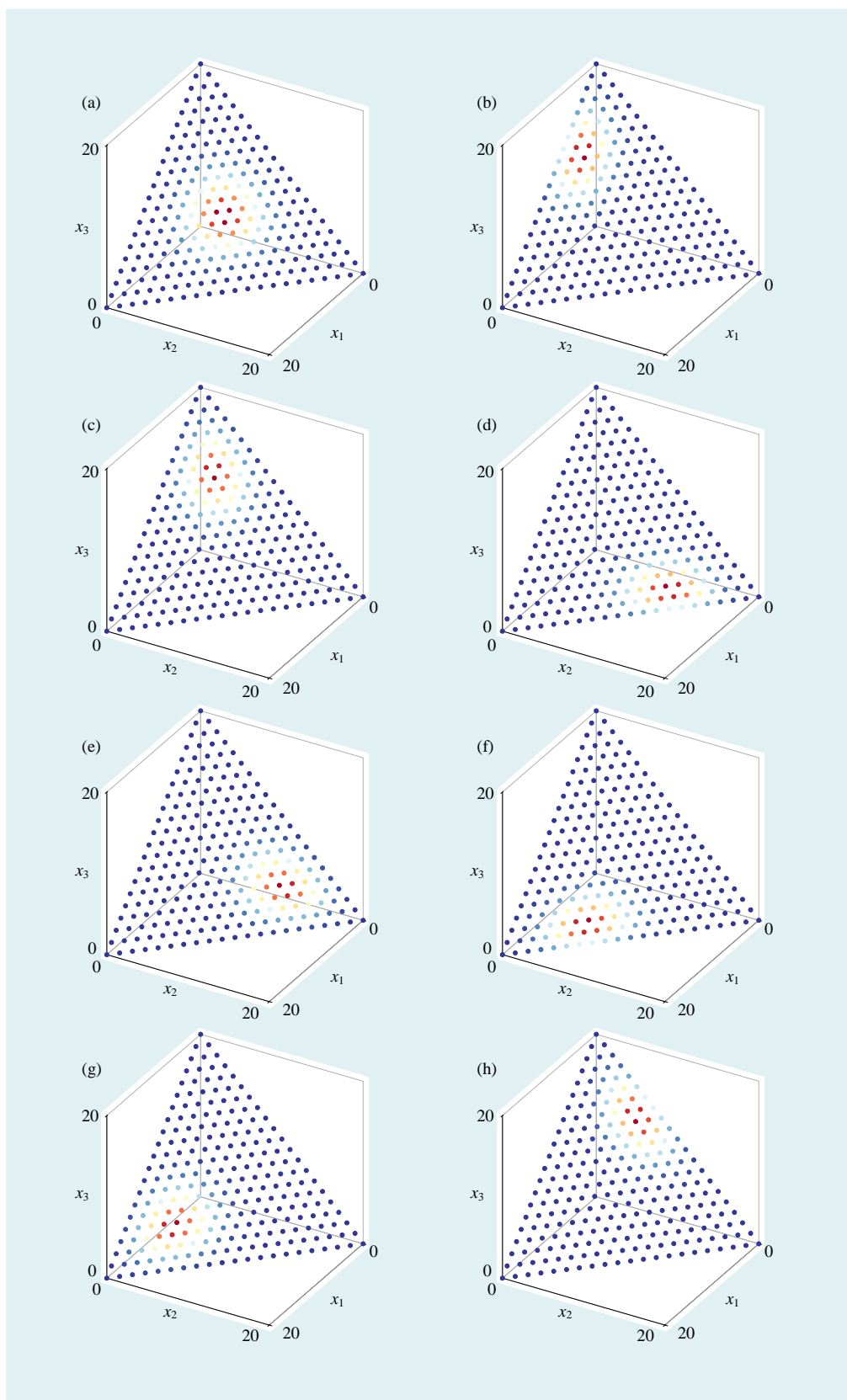

图 18 所示为利用三维散点图可视化满足特定 Dirichlet 分布的随机数。



Jupyter 笔记 BK_2_Ch13_04.ipynb 绘制图 17，BK_2_Ch13_05.ipynb 绘制图 18，请大家自行分析这两个代码文件。



本章利用三维散点图这个很普通的可视化方案作了很多有趣的案例。请大家格外关注多项分布、高斯分布、Dirichlet 分布。特别地，本书后续会专门讲解 Dirichlet 分布。

图 15. 用三维散点可视化多项分布 |  BK_2_Ch13_02.ipynb

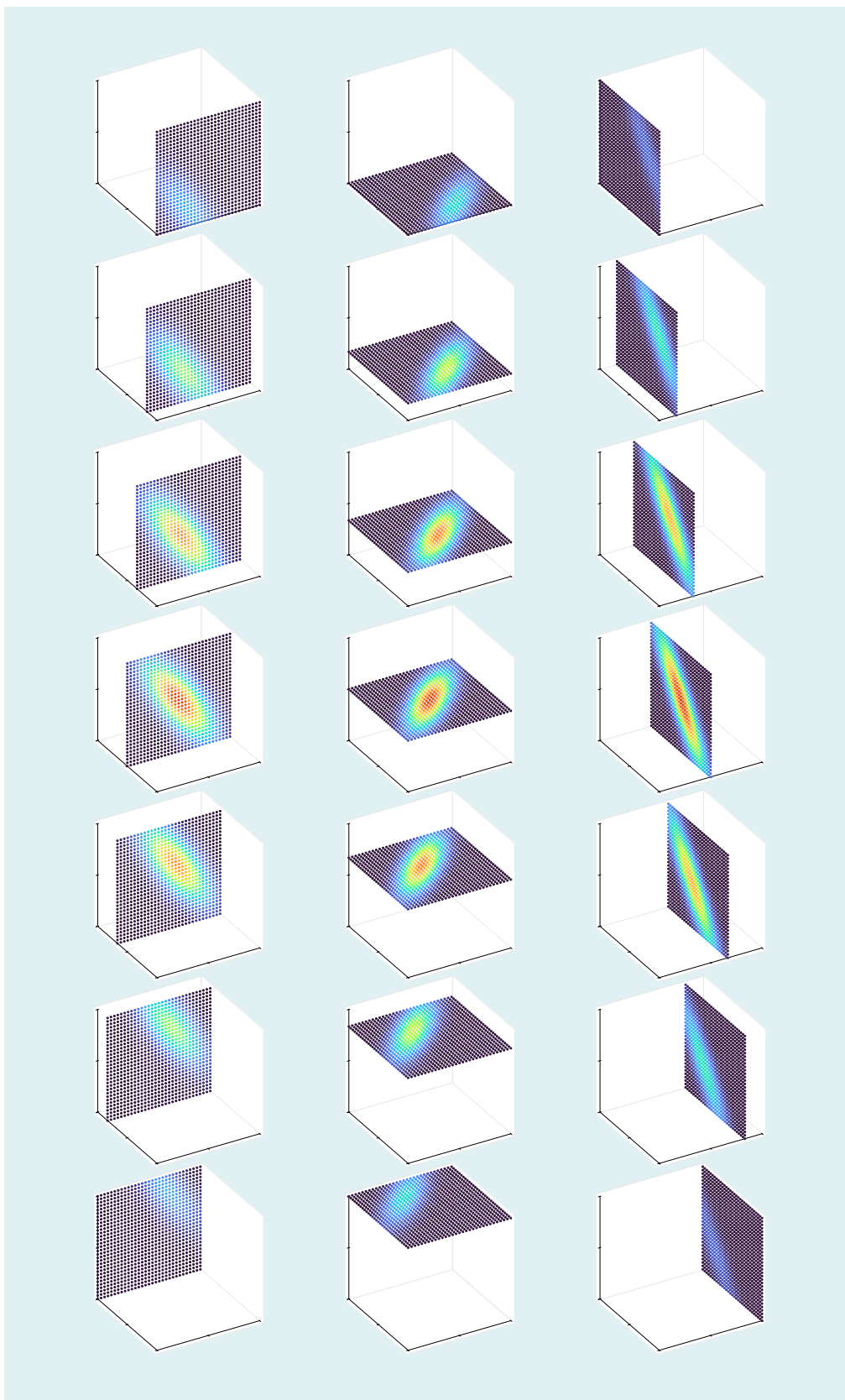

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

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 16. 用三维散点切片可视化高斯分布 |  BK_2_Ch13_03.ipynb

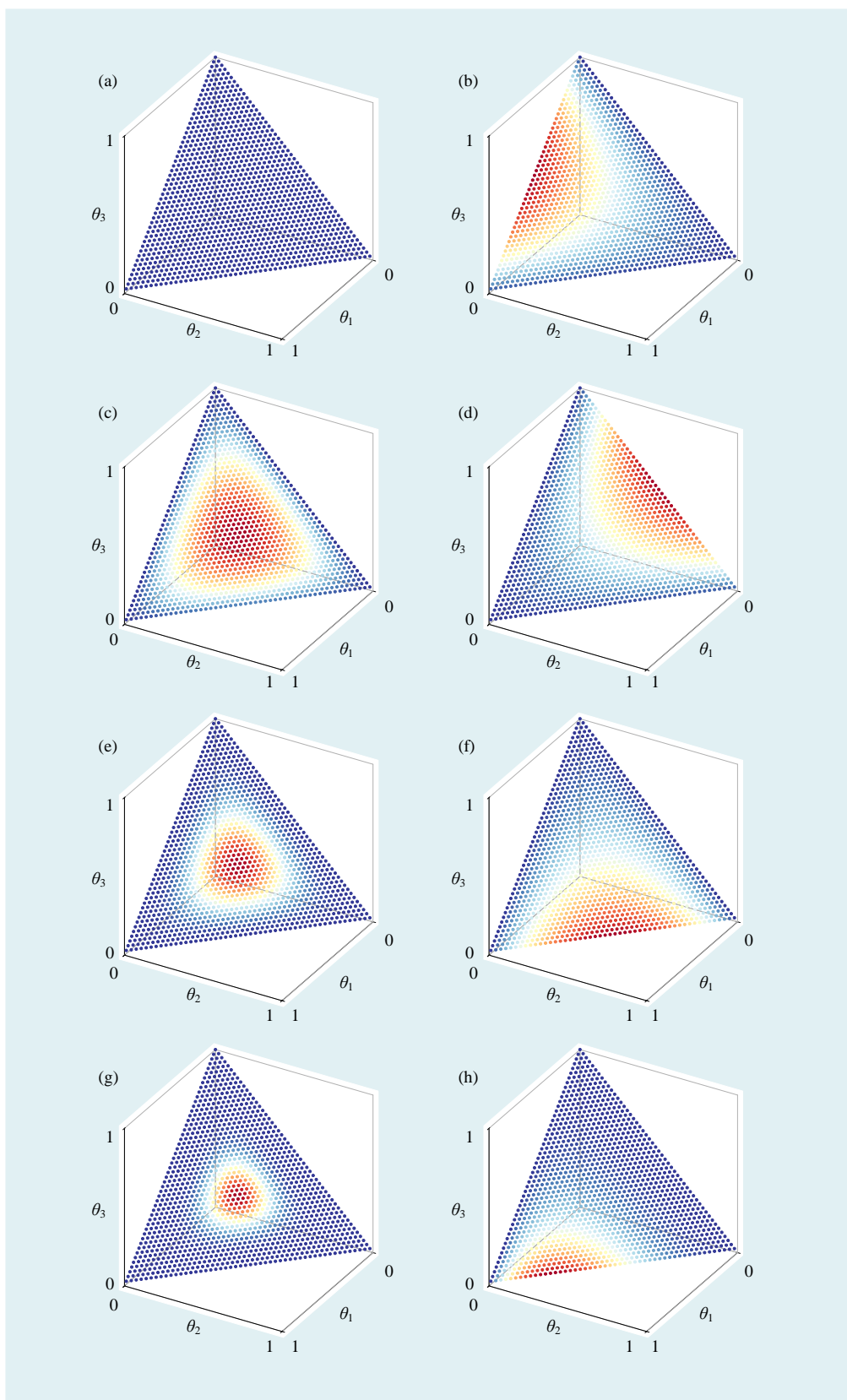

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

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 17. 用三维散点可视化 Dirichlet 分布 |  BK_2_Ch13_04.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

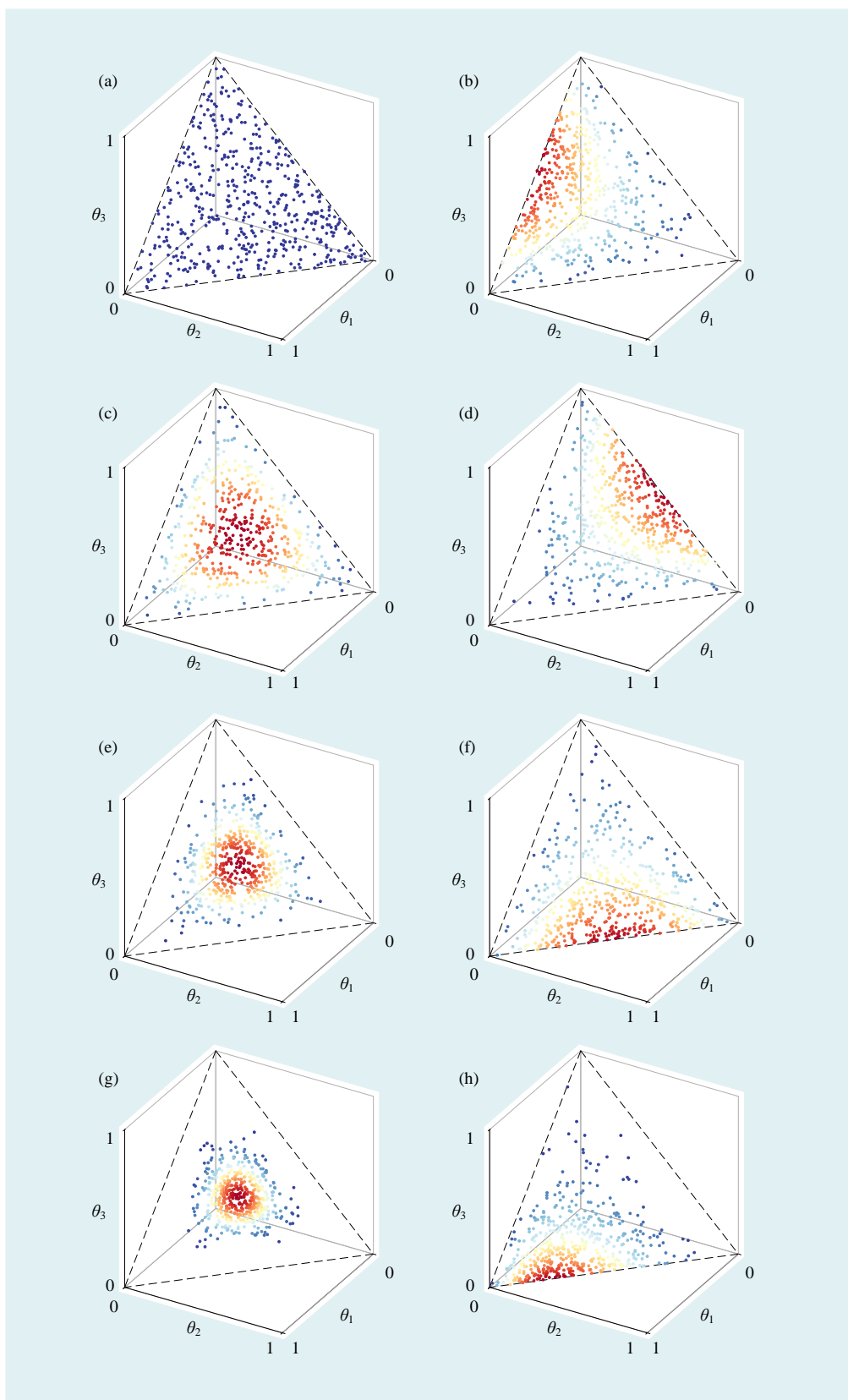



图 18. 用三维散点可视化满足 Dirichlet 分布的随机数 |  BK_2_Ch13_05.ipynb

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