

12

Descriptive Statistics Using Seaborn

Seaborn 可视化数据

使用 Seaborn 完成样本数据统计描述



理性永恒，其他一切皆有终结之时。

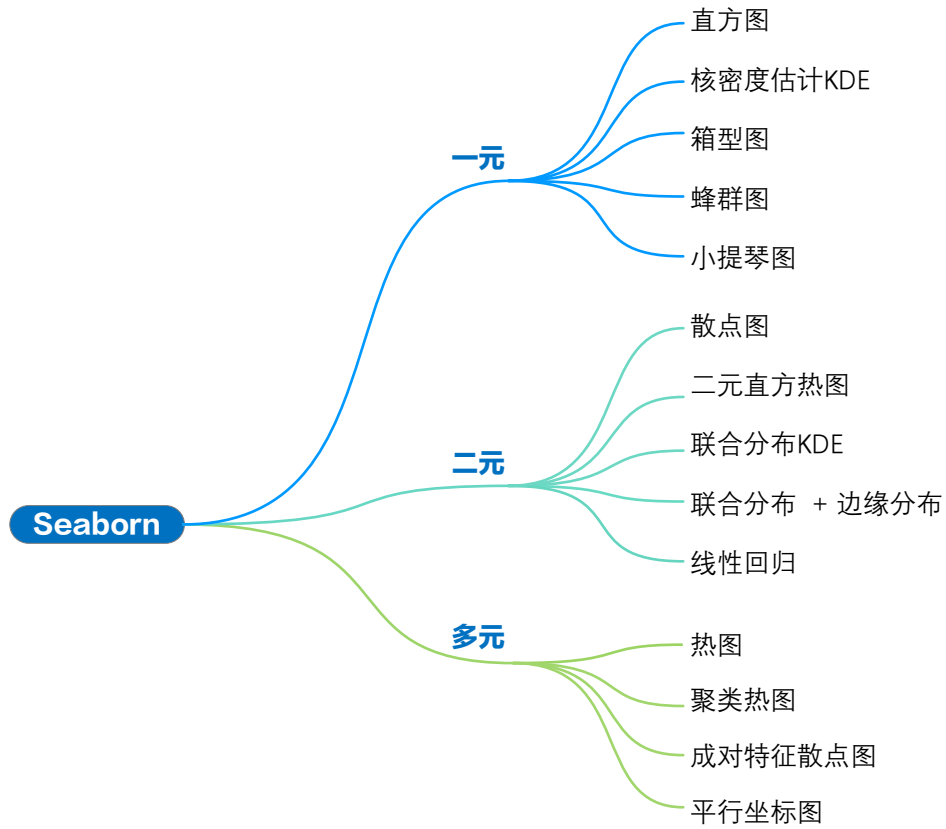
Reason is immortal, all else mortal.

—— 毕达哥拉斯 (Pythagoras) | 古希腊哲学家、数学家 | 570 ~ 495 BC



```
◀ pandas.plotting.parallel_coordinates() 绘制平行坐标图
◀ seaborn.boxplot() 绘制箱型图
◀ seaborn.heatmap() 绘制热图
◀ seaborn.histplot() 绘制频数/概率/概率密度直方图
◀ seaborn.jointplot() 绘制联合分布和边缘分布
◀ seaborn.kdeplot() 绘制 KDE 核概率密度估计曲线
◀ seaborn.lineplot() 绘制线图
◀ seaborn.lmplot() 绘制线性回归图像
◀ seaborn.pairplot() 绘制成对分析图
◀ seaborn.swarmplot() 绘制蜂群图
◀ seaborn.violinplot() 绘制小提琴图
```





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

12.1 Seaborn：统计可视化利器

本书前文介绍用 Seaborn 绘制热图。实际上，Seaborn 的真正价值体现在统计可视化上。简单来说，Seaborn 是一个用于数据可视化的 Python 库，它基于 Matplotlib，并提供了一组高级的绘图函数和样式设置，可以轻松创建具有吸引力和专业外观的统计图表。

Seaborn 提供了多种可视化方案，包括但不限于。

- ▶ 分布图：包括直方图、核密度图、箱线图，用于展示数据的分布情况。
- ▶ 散点图：用于观察两个变量之间的关系，可以通过散点图添加颜色或大小编码第三个变量。
- ▶ 线性关系图：通过绘制线性回归模型的置信区间，展示两个变量之间的线性关系。
- ▶ 分类图：包括条形图、点图、计数图等，用于比较不同类别之间的数值关系。
- ▶ 矩阵图：如热图和聚类图，用于显示数据的相似性和聚类结构。

本章以鸢尾花数据为例介绍如何用 Seaborn 可视化样本数据分布。

样本数据分布是指在统计学中，对于一组收集到的数据，对其进行统计和描述的方式。

一元样本数据分布是指只包含一个随机变量的样本数据分布，例如鸢尾花花萼长度。可视化一元样本分布的方法有：**直方图** (histogram)、**核密度估计** (Kernel Density Estimation, KDE)、**毛毯图** (rug plot)、**分散图** (strip plot)、**小提琴图** (violin plot)、**箱型图** (box plot)、**蜂群图** (swarm plot) 等等。

二元样本数据分布则涉及两个随机变量，例如鸢尾花花萼长度、花萼关系之间的关系。这种分布一般叫**联合分布** (joint distribution)。我们可以通过相关性系数量化联合分布。

边缘分布 (marginal distribution) 是指在多元数据分布中，对某一个或几个变量进行统计，而忽略其他变量的分布。例如，在花萼长度、花萼关系的二元数据分布中，对花萼长度的边缘分布就是仅考虑花萼长度变量的数据分布。

可视化二元样本分布的方法有**散点图** (scatter plot)、散点图 + 边缘直方图、散点图 + 毛毯图、散点图 + 回归图、频率热图、二元 KDE 等等图形和图形组合。

多元样本数据分布则涉及两个以上随机变量，例如鸢尾花花萼长度、花萼宽度、花瓣长度、花瓣宽度。多元样本数据的可视化方案有**热图** (heatmap)、**聚类热图** (cluster map)、**平行坐标图** (parallel plot)、成对特征散点图、Radviz 等等。特别地，我们还可以用协方差矩阵、相关性系数矩阵来量化随机变量之间的关系。而热图可以用来可视化协方差矩阵、相关性系数矩阵。

除此之外，我们在采用上述可视化方案时，还可以考虑分类，比如鸢尾花种类。

下面我们来逐一展示这些统计可视化方案。

12.2 一元特征数据

直方图

直方图是一种常用的数据可视化图表，用于显示数值变量的分布情况。

如图 1 所示，将数据划分为不同的区间（也称“柱子”），一般计算每个区间内的数据频数（样本数量）；简单来说，这个过程就是“查数”。然后，通过绘制每个区间的柱状条形来表示相应的频数。

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

比如，图 1 中深蓝的“柱子”对应区间的样本数量为 25，因此“柱子”的高度为 25。

直方图的 x 轴表示变量的取值范围，而 y 轴表示频数、概率、概率密度。图 1 中深蓝的“柱子”对应的频数为 25，样本总数为 150，因此这个柱子对应的概率为 $25/150$ 。柱子的宽度为 0.2，因此这个深蓝色柱子的概率密度为 $25/150/0.2$ 。

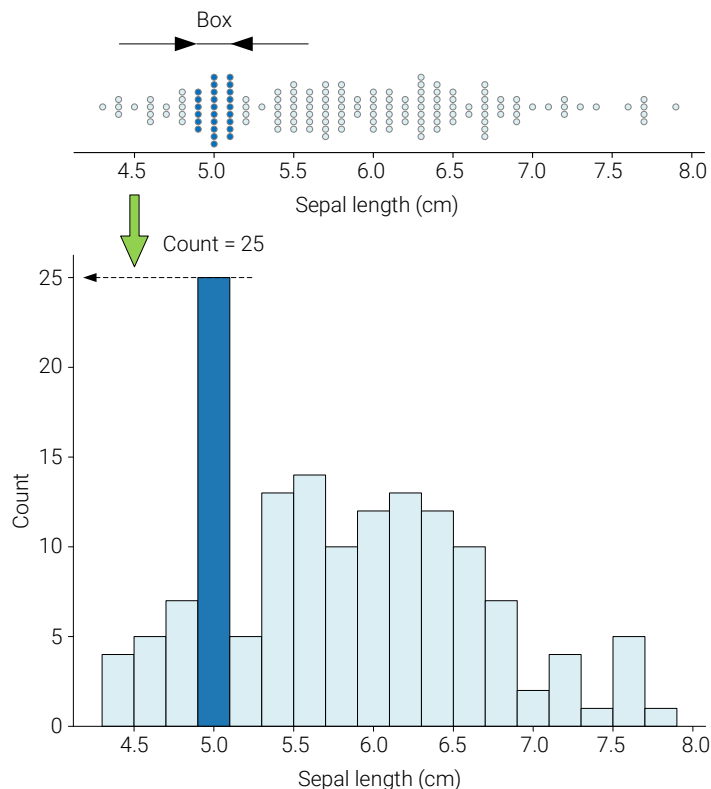
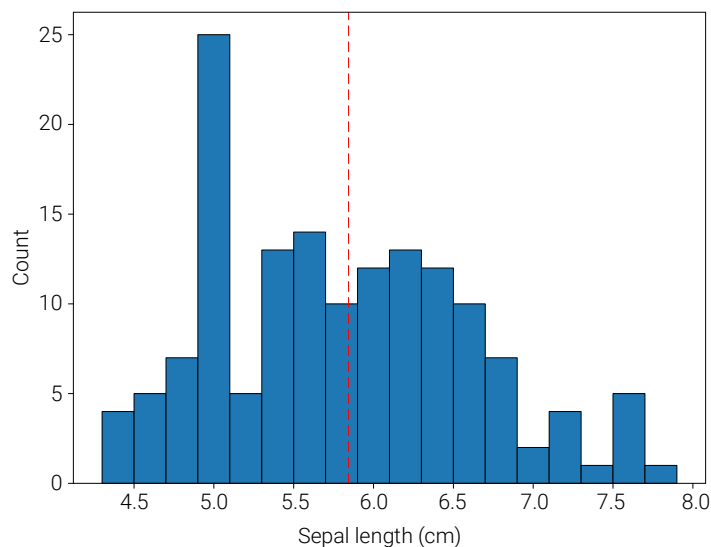


图 1. 直方图原理

图 2 所示为鸢尾花花萼长度样本数据的直方图，纵轴为频数。



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

图 2. 鸢尾花花萼长度样本数据直方图，纵轴为频数

如果图 2 的纵轴为概率，图 2 的这些“柱子”的高度之和为 1。如果图 2 的纵轴为概率密度，图 2 的这些“柱子”的面积之和为 1。

▲ 注意，标准差是方差的平方根。样本、样本均值、样本标准差、三者的单位相同。

代码 1 绘制图 2，下面聊聊其中的关键语句。

a 将 seaborn 导入，简作 sns。

b 利用 `seaborn.load_dataset()`，简作 `sns.load_dataset()`，导入鸢尾花数据。保存在 Seaborn 中的鸢尾花数据类型是 Pandas DataFrame。

请大家自己解释 **c**，简述 `fig` 和 `ax` 两个对象都有哪些用途。

d 利用 `seaborn.histplot()` 绘制直方图。参数与 `data` 一般为 Pandas 数据帧，`x` 为横轴对应的数据帧的列标签。

请大家在 JupyterLab 尝试分别绘制鸢尾花数据其他三个量化特征（花萼宽度、花瓣长度、花瓣宽度）的直方图。

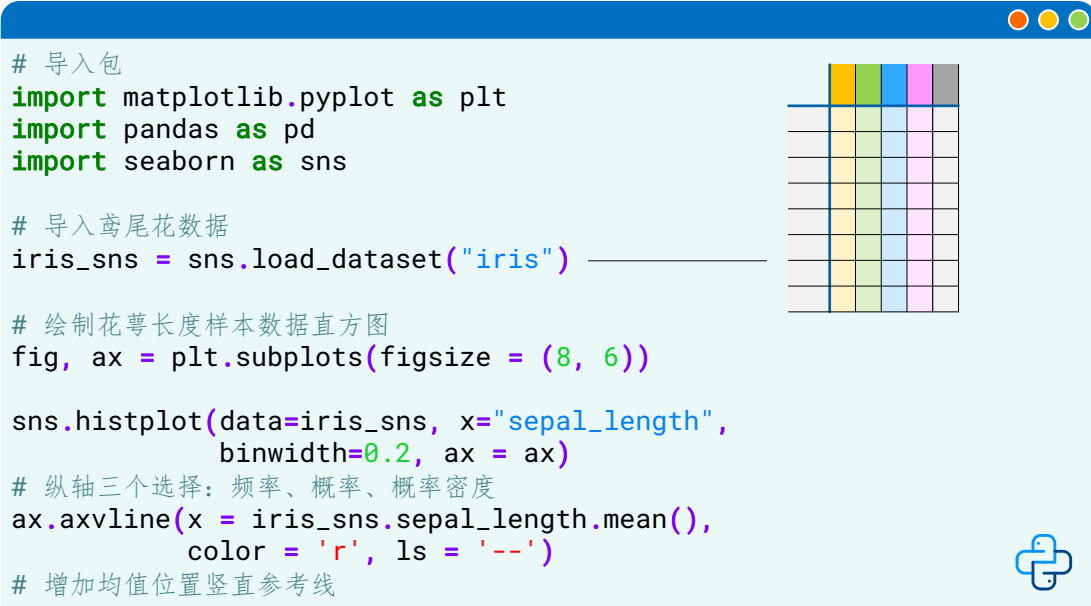
此外，参数 `stat` 指定纵轴类型，比如 `'count'` 对应频数，`'probability'` 对应概率，`'density'` 对应概率密度。可以用 `bins` 指定直方图区间数量，`binwidth` 定义区间宽度。

e 利用 `axvline()` 在轴对象 `ax` 绘制了花萼长度样本均值的位置。

这段代码中 `iris_sns.sepal_length` 取出数据帧的特定列，`sepal_length` 是列标签。而 `iris_sns.sepal_length.mean()` 则计算这一列的均值。

这是 Pandas 数据帧重要的计算方法——**链式法则** (method chaining)。简单来说，Pandas 链式法则是一种编程风格，旨在通过将多个操作链接在一起，以更清晰、紧凑的方式执行数据处理任务。

请大家修改本章配套 Jupyter Notebook，将“均值 ± 标准差”这两条直线也画上去。




```
# 导入包
import matplotlib.pyplot as plt
import pandas as pd
a import seaborn as sns

# 导入鸢尾花数据
b iris_sns = sns.load_dataset("iris")

# 绘制花萼长度样本数据直方图
c fig, ax = plt.subplots(figsize = (8, 6))

d sns.histplot(data=iris_sns, x="sepal_length",
               binwidth=0.2, ax = ax)
# 纵轴三个选择：频率、概率、概率密度
e ax.axvline(x = iris_sns.sepal_length.mean(),
             color = 'r', ls = '--')
# 增加均值位置竖直参考线
```

代码 1. 用 Seaborn 绘制直方图;  Bk1_Ch12_01.ipynb

如代码 2 所示，利用 `seaborn.histplot()` 绘制鸢尾花数据直方图时，如果指定 `hue = 'species'`，我们便得到每个类别鸢尾花单独的直方图，具体如图 3 所示。

`seaborn.histplot()` 还可以用来绘制二维直方图，本章后文将介绍。此外，本章配套的 Jupyter Notebook 还给出函数其他用法。

▲ 注意，图 3 直方图纵轴为概率密度值。

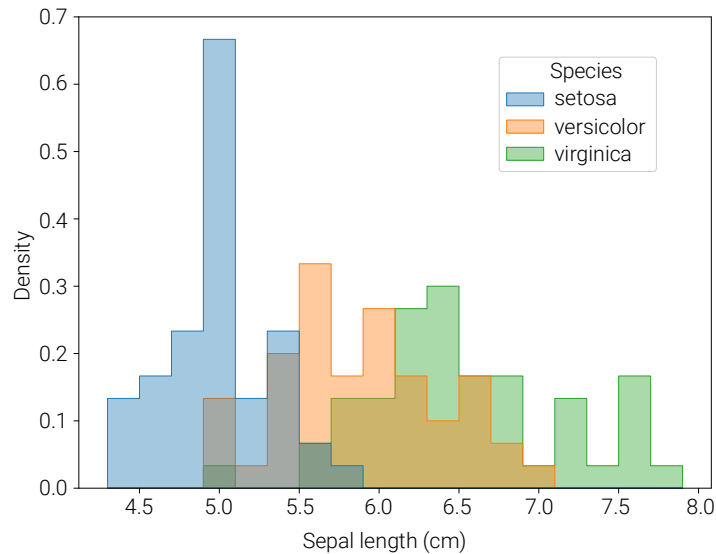


图 3. 鸢尾花花萼长度样本数据直方图，考虑鸢尾花分类，纵轴为概率密度

```
# 绘制花萼长度样本数据直方图，考虑鸢尾花分类
fig, ax = plt.subplots(figsize = (8,6))

sns.histplot(data = iris_sns, x="sepal_length",
             hue = 'species', binwidth=0.2, ax = ax,
             element="step", stat = 'density')

# 纵轴为概率密度
```

代码 2. 用 Seaborn 绘制直方图，考虑鸢尾花分类，使用时须配合前文代码； Bk1_Ch12_01.ipynb

在直方图中，以下是频数、概率和概率密度的确切定义如下。

直方图中每个区间内的样本数量被称为**频数** (frequency)。它表示了数据落入该区间的次数或计数。

概率 (probability) 是指某个事件发生的可能性。在直方图中，可以将频数除以总观测值的数量，得到每个区间的概率。这样计算得到的概率是相对频率，表示该区间中的观测值出现的相对概率。

概率密度 (probability density): 是指在概率分布函数中某一点附近单位自变量取值范围内的概率。在直方图中, 概率密度可以通过将每个区间的频数除以该区间的宽度得到。概率密度函数描述了变量的分布形状, 而不是具体的概率值。

直方图可以显示数据的分布形状, 如**对称 (symmetry)**、**偏态 (skewness)**、**峰度 (kurtosis)**等, 以及数据的中心趋势和离散程度。通过观察直方图, 我们可以直观地了解数据的分布特征, 如数据的集中程度、范围和异常值等。



《统计至简》第 1 章将专门讲解直方图、偏态、峰度等概念。

核密度估计 KDE

核密度估计 (Kernel Density Estimation, KDE) 是一种非参数方法, 用于估计连续变量的**概率密度函数 (Probability Density Function, PDF)**。它通过将每个数据点视为一个核函数 (通常是高斯核函数), 在整个变量范围内生成一系列核函数, 然后将这些核函数进行平滑和叠加, 从而得到连续的概率密度估计曲线。具体原理如图 4 所示。

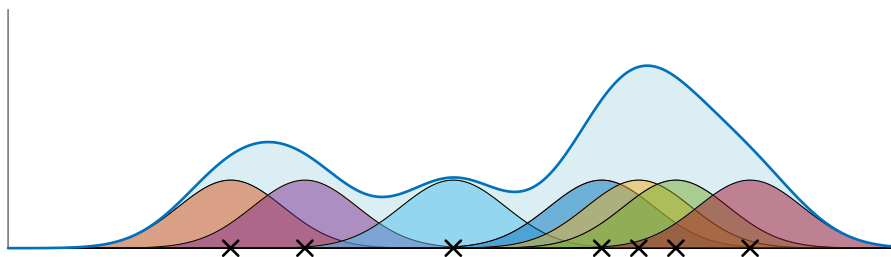


图 4. 高斯核密度估计原理

核密度估计的目标是通过在数据点附近生成高斯分布的核函数, 捕捉数据的分布特征和结构。具体地说, 每个数据点的核函数会在其附近产生一个高的小高斯分布, 然后将所有核函数叠加在一起。通过调整核函数的带宽参数, 可以控制估计曲线的平滑程度和敏感度。



本书第 27 章将介绍如何使用 Statsmodels 中的核密度估计函数; 《统计至简》第 17 章将专门讲解核密度估计原理。

图 5 所示为利用 `seaborn.kdeplot()` 绘制的鸢尾花花萼长度数据高斯核密度估计 PDF。可以这样理解, 图 5 是图 2 直方图的“平滑”处理结果。

图 5 的横轴还有用 `seaborn.rugplot()` 绘制的毛毯图。毛毯图常用于展示数据在一维空间上的分布。它通过在坐标轴上绘制短线, 或称为“毛毯”, 表示数据点的位置和密度。这种图形通常用于辅助其他类型的图表, 如直方图或密度图, 以更清晰地显示数据的分布特征。

在用 `seaborn.kdeplot()` 绘制花萼长度样本数据核密度估计曲线时, 我们还可以用 `hue` 来绘制三类鸢尾花种类各自的分布, 具体如图 6 所示。

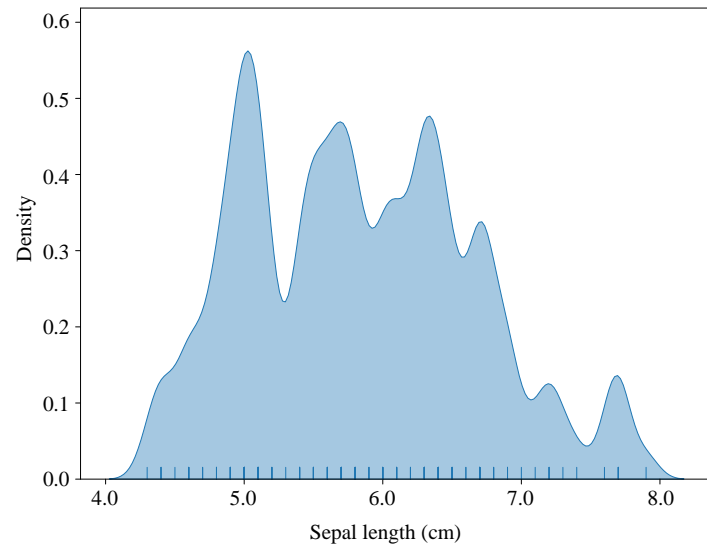


图 5. 鸢尾花花萼长度样本数据核密度估计

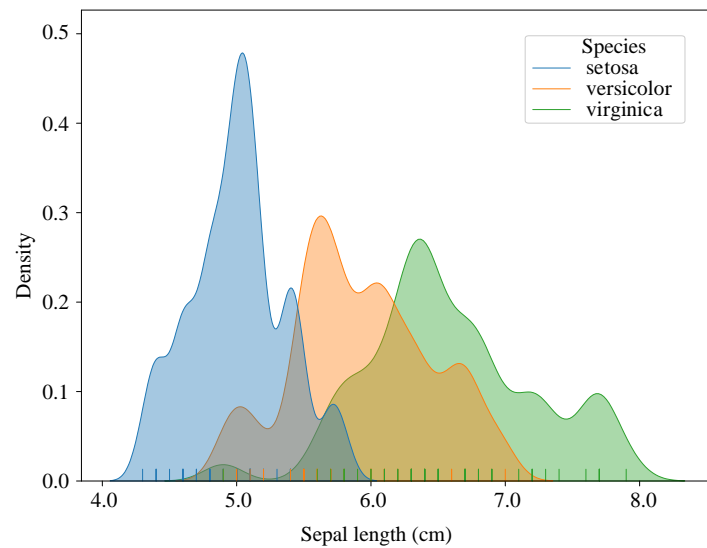


图 6. 鸢尾花花萼长度样本数据核密度估计，考虑鸢尾花分类

换个角度理解图 6，图 6 中三条曲线叠加便得到图 5。图 7 这幅图更好地解释了这一点。用 `seaborn.kdeplot()` 绘制这幅图时，需要设置 `multiple="stack"`。

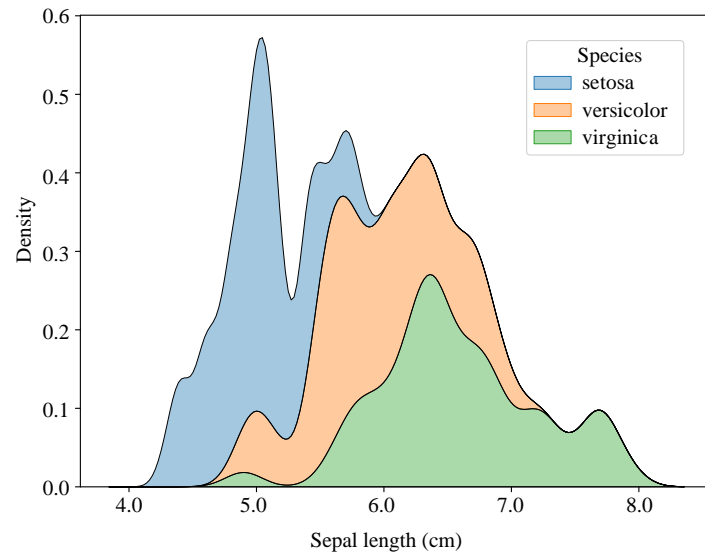


图 7. 三条 KDE 曲线叠加

特别地，在利用绘制核密度估计曲线时，如果设置 `multiple = 'fill'`，我们便获得图 8。图中每条曲线准确来说，都是“**后验概率 (posterior)**”。而这个后验概率值可以用来完成分类。

也就是说，给定具体花萼长度，比较该点处红蓝绿三条曲线对应的宽度，最宽的曲线对应的鸢尾花种类可以作为该点的鸢尾花分类预测值。因此，这个后验概率值也叫“**成员值 (membership score)**”。

➡ 想要理解后验概率这个概念，需要大家深入理解贝叶斯定理，《统计至简》第 18、19 章将专门介绍利用贝叶斯定理完成分类。

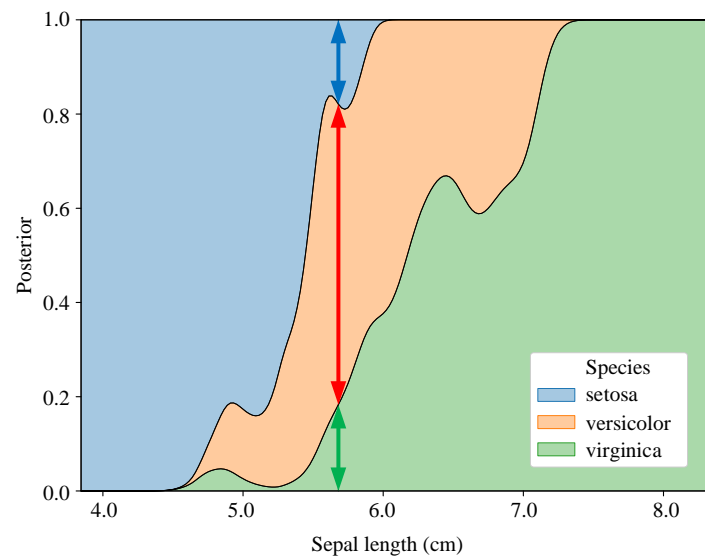


图 8. 后验概率曲线

代码 3 绘制图 5 ~ 图 8，请大家自行分析这段代码，并逐行注释。

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

```


# 绘制花萼长度样本数据，高斯核密度估计
fig, ax = plt.subplots(figsize = (8,6))
a sns.kdeplot(data=iris_sns, x='sepal_length',
b           bw_adjust=0.3, fill = True)
b sns.rugplot(data=iris_sns, x='sepal_length')

# 绘制花萼长度样本数据，高斯核密度估计，考虑鸢尾花类别
fig, ax = plt.subplots(figsize = (8,6))
c sns.kdeplot(data=iris_sns, x='sepal_length', hue = 'species',
d           bw_adjust=0.5, fill = True)
c sns.rugplot(data=iris_sns, x='sepal_length', hue = 'species')

# 绘制花萼长度样本数据，高斯核密度估计，考虑鸢尾花类别，堆叠
fig, ax = plt.subplots(figsize = (8,6))
d sns.kdeplot(data=iris_sns, x='sepal_length', hue= 'species',
e           multiple='stack', bw_adjust=0.5)

# 绘制后验概率（成员值）
fig, ax = plt.subplots(figsize = (8,6))
e sns.kdeplot(data=iris_sns, x='sepal_length',
             hue='species', bw_adjust=0.5, multiple = 'fill')

```

代码 3. 用 Seaborn 绘制高斯核密度估计，使用时须配合前文代码；  Bk1_Ch12_01.ipynb



什么是贝叶斯定理？

贝叶斯定理是一种用于更新概率推断的数学公式。它描述了在获得新信息后如何更新我们对某个事件发生概率的信念。贝叶斯定理基于先验概率（我们对事件发生的初始信念）和条件概率（给定新信息的情况下事件发生的概率），通过计算后验概率（在获得新信息后事件发生的概率）来实现更新。贝叶斯定理在统计学、机器学习和人工智能等领域具有广泛应用。

分散点图

分散点图（strip plot）一般用来可视化一组分类变量与连续变量的关系。在分散图中，每个数据点通过垂直于分类变量的轴上的一个点表示，连续变量的取值则沿着水平轴展示。

这种图形通常用于可视化分类变量和数值变量之间的关系，以观察数据的分布、聚集和离散程度，同时也可以用于比较不同分类变量水平下的数值变量。

代码 4 中的 `seaborn.stripplot()` 是 Seaborn 库中用于绘制分散点图的函数。需要注意的是，分散点图适用于较小的数据集，当数据点重叠较多时，可考虑使用 `seaborn.swarmplot()` 函数来避免重叠点问题。

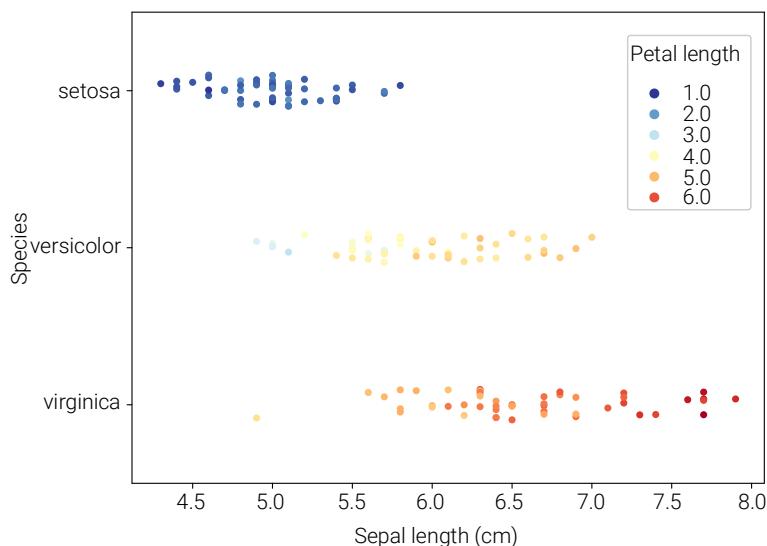



图 9. 分散点图

```
# 绘制鸢尾花花萼长度分散点图
fig, ax = plt.subplots(figsize = (8,6))
sns.stripplot(data=iris_sns, x='sepal_length', y='species',
              hue='petal_length', palette='RdYlBu_r', ax = ax)
```

代码 4. 用 Seaborn 绘制分散点图, 考虑鸢尾花分类, 使用时须配合前文代码;  Bk1_Ch12_01.ipynb

蜂群图

蜂群图 (swarm plot) 是一种用于可视化分类变量和数值变量关系的图表类型。它通过在分类轴上对数据进行分散排列, 避免数据点的重叠, 以展示数值变量在不同类别下的分布情况。每个数据点在分类轴上的位置表示其对应的数值大小, 从而呈现出数据的密度和分布趋势。

蜂群图可以帮助我们比较不同类别之间的数值差异和趋势, 适用于数据探索、特征分析和可视化报告等场景。

图 10 所示为利用 `seaborn.swarmplot()` 绘制蜂群图。图 11 所示为考虑鸢尾花分类的蜂群图。请大家自行分析代码 5。

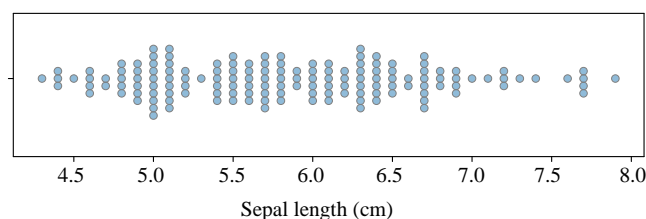


图 10. 蜂群图

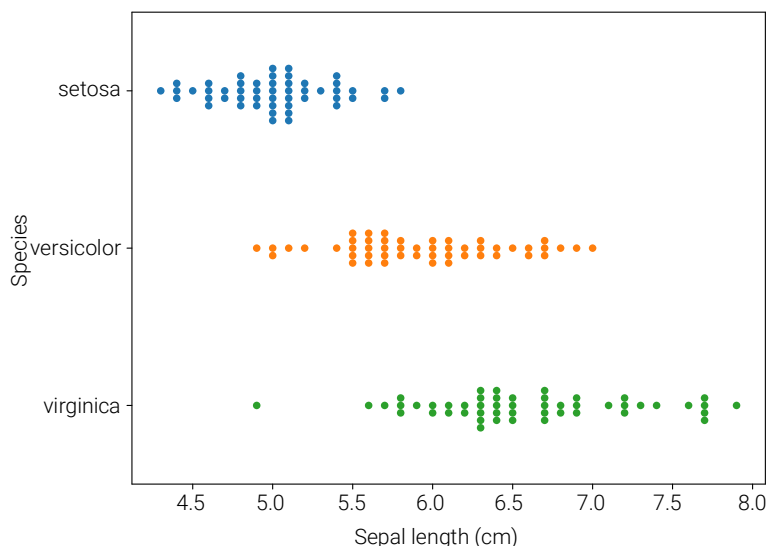



图 11. 蜂群图，考虑鸢尾花分类

```
# 绘制花萼长度样本数据，蜂群图
fig, ax = plt.subplots(figsize = (8,4))
sns.swarmplot(data=iris_sns, x="sepal_length", ax = ax)

# 绘制花萼长度样本数据，蜂群图，考虑分类
fig, ax = plt.subplots(figsize = (8,4))
sns.swarmplot(data=iris_sns, x="sepal_length", y = 'species',
              hue = 'species', ax = ax)
```

代码 5. 用 Seaborn 绘制蜂群图，使用时须配合前文代码；  Bk1_Ch12_01.ipynb

箱型图

箱型图 (box plot) 是一种常用的统计图表，用于展示数值变量的分布情况和异常值检测。

箱型图通过绘制数据的五个关键统计量（最小值、第一四分位数 Q_1 、中位数 Q_2 、第三四分位数 Q_3 、最大值）以及可能存在的异常值来提供对数据的直观概览。

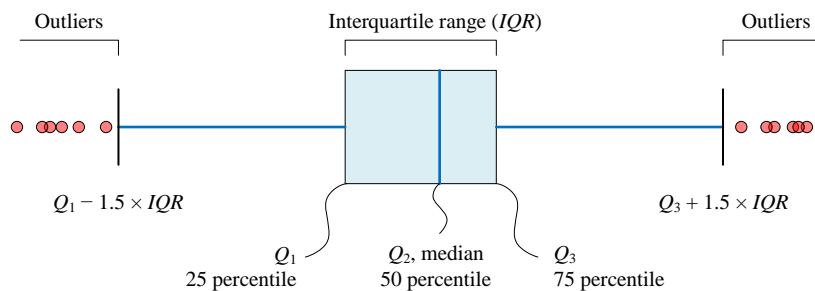


图 12. 箱型图原理

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



什么是四分位？

四分位是统计学中用于描述数据集分布的概念，将数据按大小顺序分成四等份。第一个四分位数 Q_1 表示 25% 的数据小于或等于它，第二个四分位数 Q_2 是中位数，表示 50% 的数据小于或等于它，第三个四分位数 Q_3 表示 75% 的数据小于或等于它。四分位可以帮助了解数据的中心趋势、分散程度和异常值。四分位与盒须图、离群值检测等统计分析方法密切相关。

图 13 所示为利用 `seaborn.boxplot()` 绘制的鸢尾花花萼长度样本数据的箱型图。图 14 所示为考虑鸢尾花分类的箱型图。

箱型图的主要元素包括：

- **箱体 (box)**：由第一四分位数 Q_1 和第三四分位数 Q_3 之间的数据范围组成。箱体的高度表示数据的四分位距 $IQR = Q_3 - Q_1$ ，箱体的中线表示数据的中位数。
- **须 (whisker)**：延伸自箱体的线段，表示数据的整体分布范围。通常，须的长度为 1.5 倍的四分位距。但是，仔细观察图 13，我们会发现用 Seaborn 绘制的箱型图左须距离 Q_1 、右须距离 Q_3 宽度并不相同。根据 Seaborn 的技术文档，左须、右须延伸至该范围 $[Q_1 - 1.5 \times IQR, Q_3 + 1.5 \times IQR]$ 内最远的样本点，具体如图 15 所示。更为极端的样本会被标记为异常值。
- **异常值 (outliers)**：范围 $[Q_1 - 1.5 \times IQR, Q_3 + 1.5 \times IQR]$ 之外的数据点，被认为是异常值，可能表示数据中的极端值或异常观测。

通过观察箱型图，可以快速了解数据的中心趋势、离散程度以及是否存在异常值等关键信息。

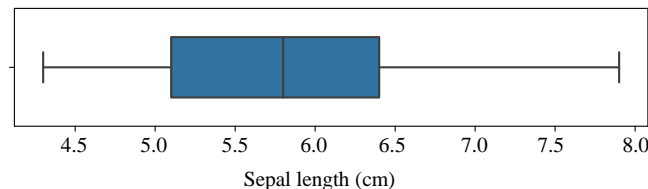


图 13. 箱型图

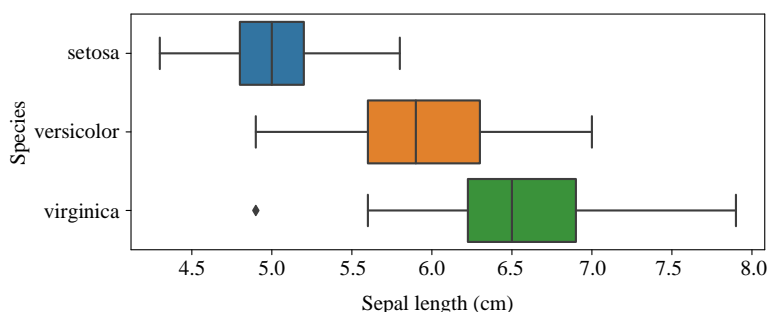


图 14. 箱型图，考虑鸢尾花分类

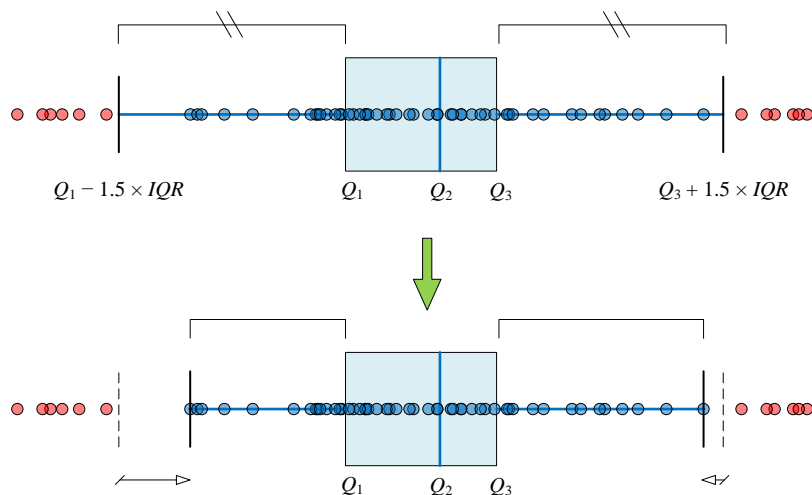



图 15. Seaborn 绘制箱型图左须、右须位置

请大家自行分析代码 6。

```
# 绘制鸢尾花花萼长度箱型图
fig, ax = plt.subplots(figsize = (8,2))
a sns.boxplot(data = iris_sns, x='sepal_length', ax = ax)

# 绘制鸢尾花花萼长度箱型图，考虑鸢尾花分类
fig, ax = plt.subplots(figsize = (8,3))
b sns.boxplot(data = iris_sns, x = 'sepal_length',
              y = 'species', ax = ax)
```

代码 6. 用 Seaborn 绘制箱型图，使用时须配合前文代码；  Bk1_Ch12_01.ipynb

小提琴图

小提琴图 (violin plot) 是一种用于可视化数值变量分布的图表类型。它结合了核密度估计曲线和箱型图的特点，可以同时展示数据的分布形状、中位数、四分位数和离群值等信息。

`seaborn.violinplot()` 是 Seaborn 库中用于绘制小提琴图的函数。

小提琴图的主要组成部分包括：

- ▶ 背景形状：由核密度估计曲线组成，表示数据在不同值上的概率密度。
- ▶ 中位数线：位于核密度估计曲线的中间位置，表示数据的中位数。
- ▶ 四分位线：分别位于核密度估计曲线的 25% 和 75% 位置，表示数据的四分位范围。
- ▶ 离群值点：位于核密度估计曲线之外的离群值数据点。

图 16 所示为用 `seaborn.violinplot()` 绘制的鸢尾花花萼长度样本数据的小提琴图。图 17 为考虑鸢尾花分类的小提琴图。图 18 所示为“蜂群图 + 小提琴图”可视化方案。

请大家自行分析代码 7。

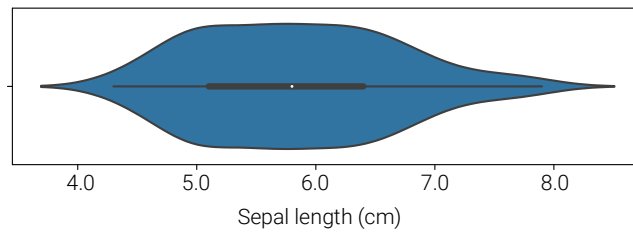


图 16. 小提琴图

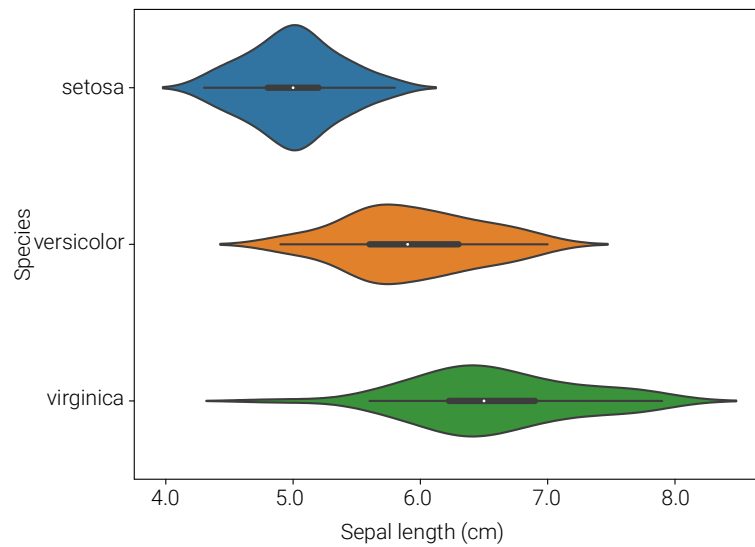


图 17. 小提琴图，考虑鸢尾花分类

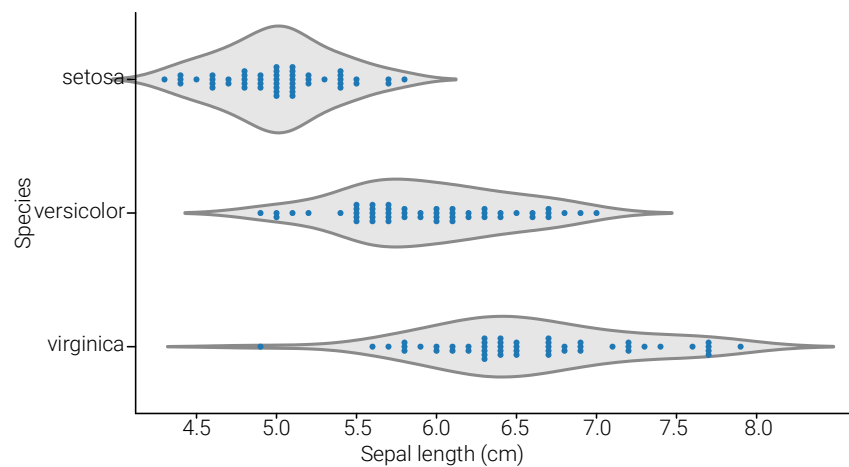


图 18. 蜂群图 + 小提琴图，考虑鸢尾花分类

```


# 绘制花萼长度样本数据，小提琴图
fig, ax = plt.subplots(figsize = (8,2))
a sns.violinplot(data=iris_sns, x='sepal_length', ax = ax)

# 绘制花萼长度样本数据，小提琴图，考虑分类
fig, ax = plt.subplots(figsize = (8,4))
b sns.violinplot(data=iris_sns, x='sepal_length',
                 y='species', ax = ax)

# 蜂群图 + 小提琴图，考虑鸢尾花分类
c sns.catplot(data=iris_sns, x='sepal_length', y='species',
              kind='violin', color='.9', inner=None)

d sns.swarmplot(data=iris_sns, x='sepal_length',
                y='species', size=3)

```

代码 7. 用 Seaborn 绘制小提琴图，使用时须配合前文代码； Bk1_Ch12_01.ipynb

12.3 二元特征数据

散点图

散点图是一种数据可视化图表，用于展示两个变量之间的关系。它通过在坐标系中以点的形式表示每个数据点，横轴代表一个变量，纵轴代表另一个变量。

散点图可以帮助我们观察和分析数据点之间的趋势、分布和相关性。通过观察点的聚集程度和分布形状，我们可以推断两个变量之间的关系类型，如线性正相关、线性负相关、线性无关，甚至是非线性关系。

图 19 所示为利用 `seaborn.scatterplot()` 绘制的散点图，散点图的横轴为花萼长度、纵轴为花萼宽度。通过观察散点趋势，可以发现花萼长度、花萼宽度似乎似乎存在线性正相关。但是实际情况可能并非如此。本章最后将通过线性相关性系数进行量化确认。

图 19 这幅图中，我们还用毛毯图分别可视化花萼长度、花萼宽度的分布情况。

用不同颜色散点代表鸢尾花分类，我们便得到图 20 所示散点图。观察这幅图中蓝色点，即 `setosa` 类，我们可以发现更强的线性正相关性。

请大家自行分析代码 8，并逐行注释。

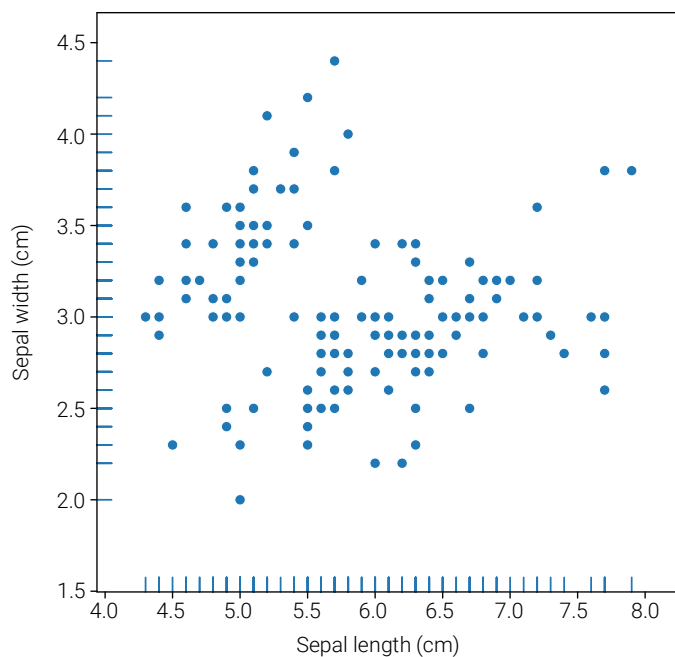


图 19. 散点图 + 毛毯图

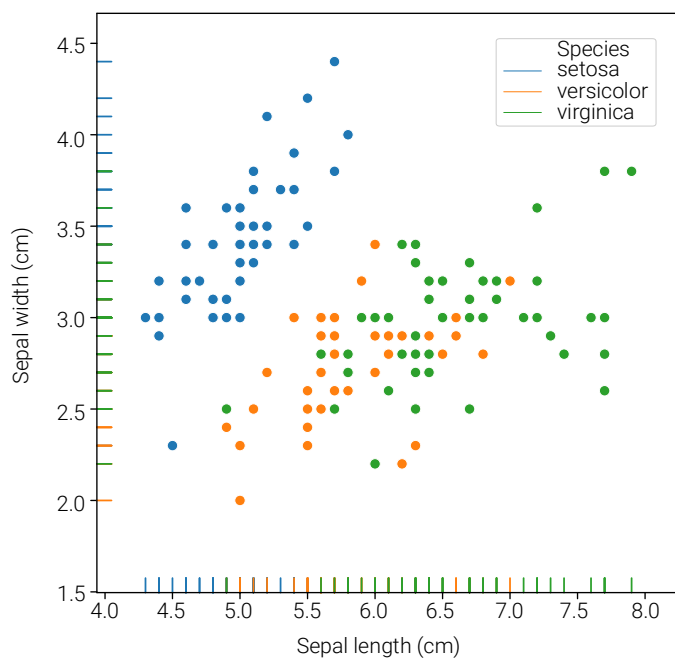


图 20. 散点图 + 毛毯图, 考虑鸢尾花分类

```

# 鸢尾花散点图 + 毛毯图
fig, ax = plt.subplots(figsize = (4,4))

a sns.scatterplot(data=iris_sns, x='sepal_length', y='sepal_width')
b sns.rugplot(data=iris_sns, x='sepal_length', y='sepal_width')

# 鸢尾花散点图 + 毛毯图，考虑鸢尾花分类
fig, ax = plt.subplots(figsize = (4,4))

c sns.scatterplot(data=iris_sns, x='sepal_length',
                  y='sepal_width', hue = 'species')
d sns.rugplot(data=iris_sns, x='sepal_length',
              y='sepal_width', hue = 'species')

```


代码 8. 用 Seaborn 绘制二元散点图 + 毛毯图，使用时须配合前文代码； Bk1_Ch12_01.ipynb

图 32 所示为几种用 `seaborn.scatterplot()` 绘制的鸢尾花数据集散点图。

图 32 (a) 的横轴是花萼长度，纵轴是花萼花萼宽度。调转横纵轴特征便得到图 32 (b)。

在图 32 (a) 的基础上，可以用色调代表花瓣长度。这样一幅二维散点图上，我们便可可视化三个量化特征。

图 32 (d) 在图 32 (c) 基础上又进一步，用散点大小代表花瓣宽度。

图 32 (e) 则用颜色可视化鸢尾花的分类标签。在此基础上，我们还可以用散点大小可视化花瓣宽度。

图 32 (g) 则集合前几幅散点图，并且用不同标识符号代表鸢尾花分类标签。这种散点图显然“信息过载”，并不推荐。

二元直方热图

本章前文，我们将一元样本数据划分成不同区间便可以绘制一元直方图。

类似地，如果我们把图 19 所示平面划分成如图 21 所示一系列格子，计算每个格子中的样本数，我们便可以绘制类似图 22 二元直方图。

显然，这种可视化方案并不理想。一方面“柱子”的高度很难确认，而且固定某个特定视角之后，一些较矮的“柱子”必定会被遮挡。因此，在实践中我们常常使用二元直方热图作为可视化方案。

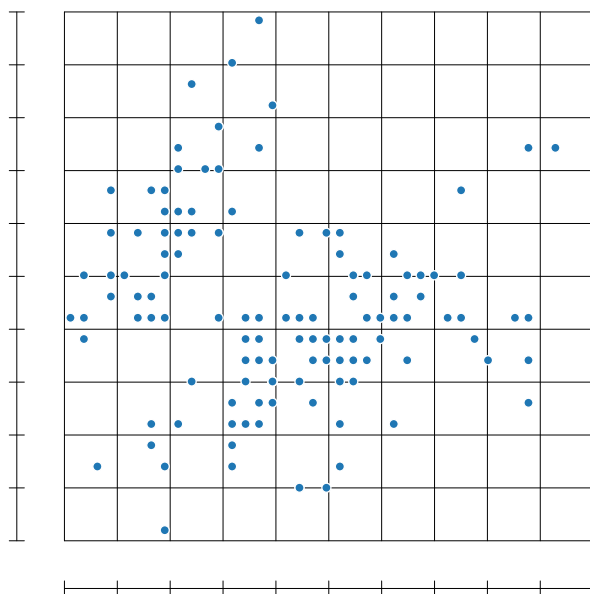


图 21. 二元直方图原理

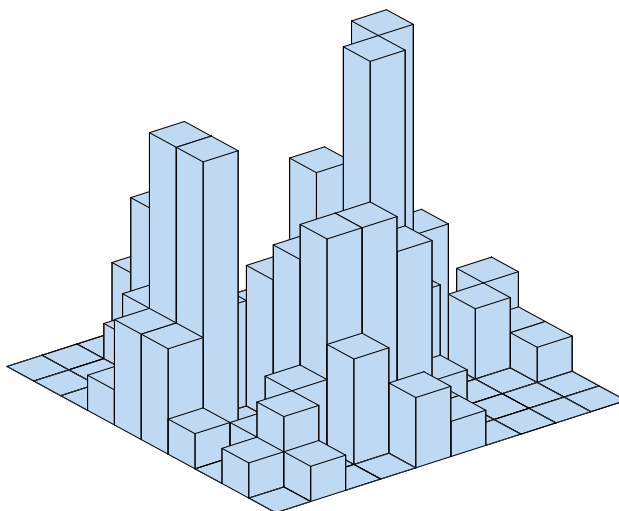


图 22. 二元直方图，柱状图可视化方案

二元直方热图由一个矩形网格组成，其中每个单元格的颜色代表了对应的数据频数、概率、概率密度。通常，行和列代表两个不同的随机变量，而单元格中的颜色强度表示频数、概率、概率密度。

二元直方热图可以帮助我们观察两个变量之间的关系以及它们的分布模式。通过观察颜色的变化和集中区域，我们可以得出关于两个变量之间的相关性、联合分布和潜在模式的初步结论。

所示为利用 `seaborn.displot()` 绘制的二元直方热图，横轴为鸢尾花花萼长度，纵轴为花萼宽度。如图 24 所示，二元直方热图沿着某个方向压缩便得到一元直方图；反过来看，直方图沿着特定方向展开便得到二元直方热图。

请大家自行分析代码 9。

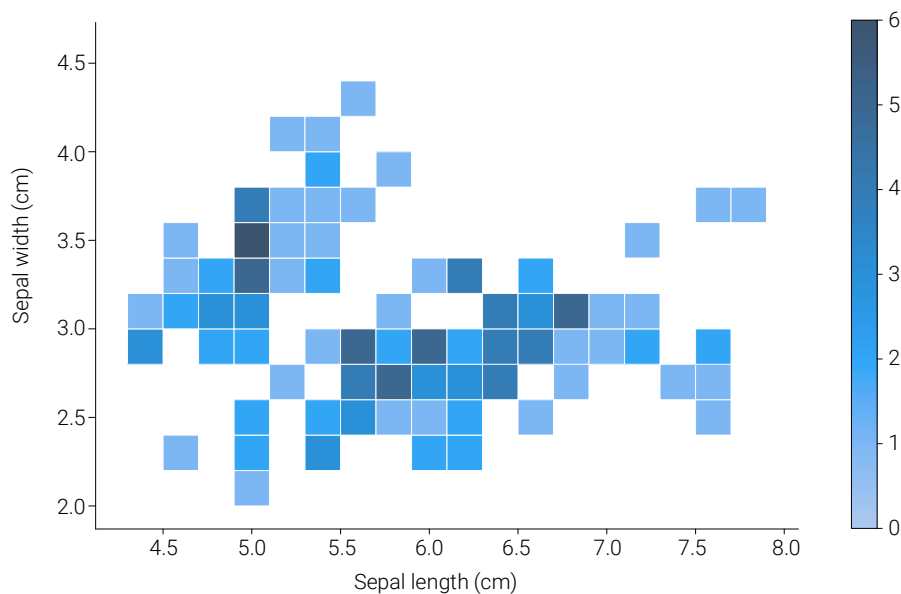


图 23. 鸢尾花花萼长度、花萼宽度的二元直方热图

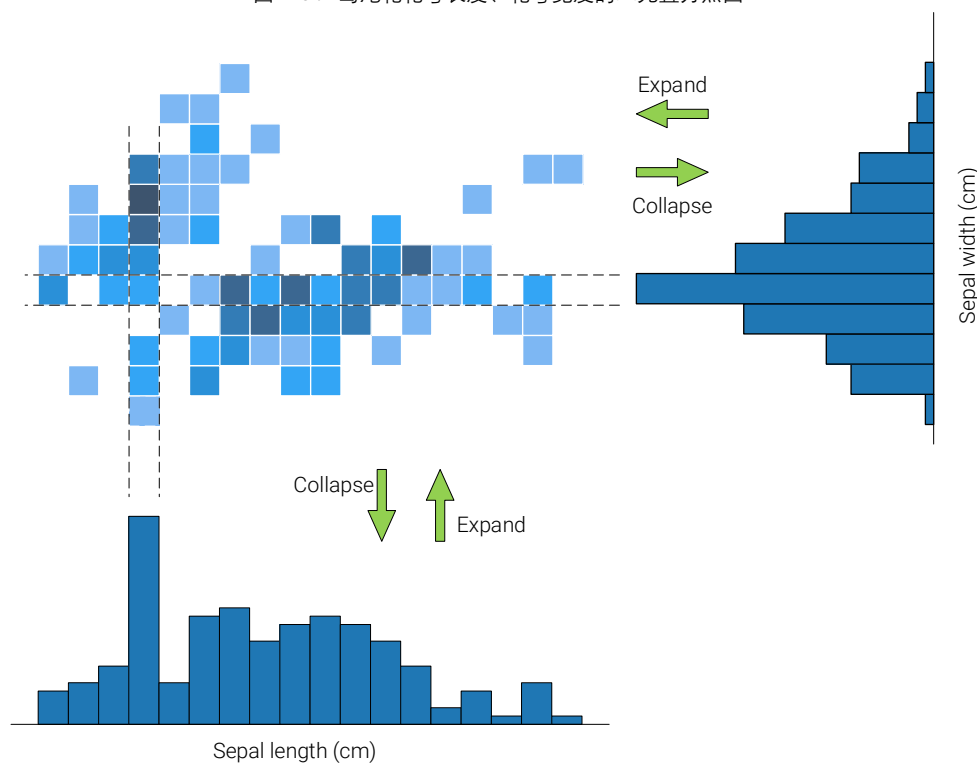


图 24. 一元直方图和二元直方热图之间关系

```
# 鸢尾花二元频率直方热图

sns.displot(data=iris_sns, x="sepal_length", y="sepal_width",
            binwidth=(0.2, 0.2), cbar=True)
```

代码 9. 用 Seaborn 绘制二元直方热图, 使用时须配合前文代码; Bk1_Ch12_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

联合分布 KDE

前文的高斯核函数 KDE 也可以用在估算二元联合分布。图 25 所示为 `seaborn.kdeplot()` 绘制鸢尾花花萼长度、花萼宽度联合分布概率密度估计等高线。图 25 (b) 还考虑了鸢尾花三个不同类别。请大家自行分析代码 10。

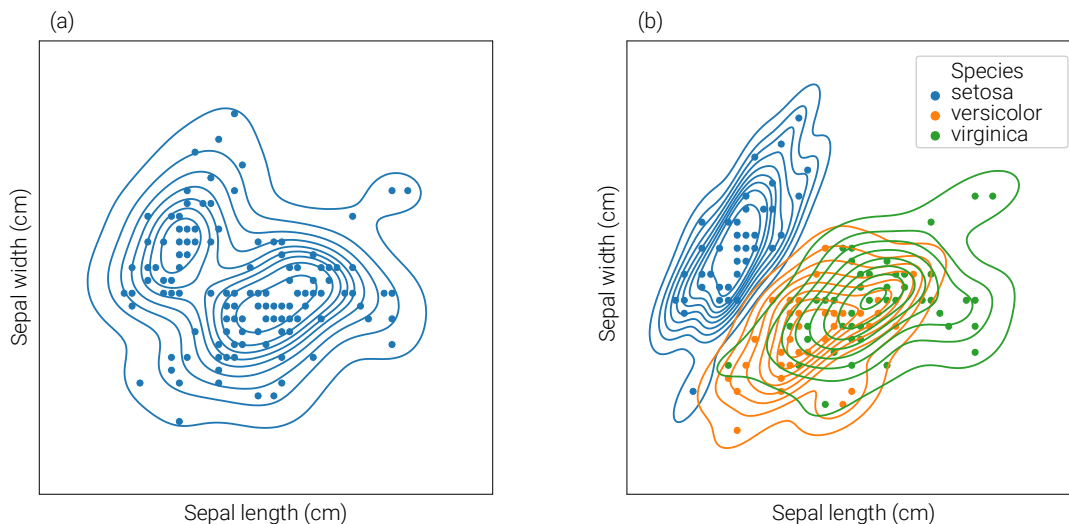


图 25. 鸢尾花花萼长度、花萼宽度的联合分布，高斯核密度估计



什么是联合分布？

联合分布是统计学中用于描述两个或多个随机变量同时取值的概率分布。它提供了关于多个变量之间关系的信息，包括它们的联合概率、相互依赖程度以及共同变化的模式。联合分布可以以多种形式呈现，如概率质量函数（离散变量）或概率密度函数（连续变量）。通过分析联合分布，我们可以洞察变量之间的相关性、条件概率以及预测和推断未来事件的可能性。联合分布在概率论、统计建模、数据分析和机器学习等领域具有广泛应用。

```
# 联合分布概率密度等高线
sns.displot(data=iris_sns, x='sepal_length',
            y='sepal_width', kind='kde')

# 联合分布概率密度等高线，考虑分布
sns.kdeplot(data=iris_sns, x='sepal_length',
            y='sepal_width', hue = 'species')
```

代码 10. 用 Seaborn 绘制联合分布概率密度等高线，使用时须配合前文代码； Bk1_Ch12_01.ipynb

联合分布 + 边缘分布

图 26 所示为利用 `seaborn.jointplot()` 可视化“联合分布 + 边缘分布”。

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

`seaborn.jointplot()` 函数用于创建联合图，结合了两个变量的散点图和各自的边缘分布图。它可以帮助我们同时可视化两个变量之间的关系以及它们的边缘分布。

`seaborn.jointplot()` 函数默认情况下会绘制散点图和边缘直方图。散点图展示了两个变量之间的关系，而边缘直方图则分别显示了每个变量的边缘分布情况。

请大家自行分析代码 11。

本章配套 Jupyter Notebook 还提供 `seaborn.jointplot()` 其他几种可视化方案，请大家自行学习。

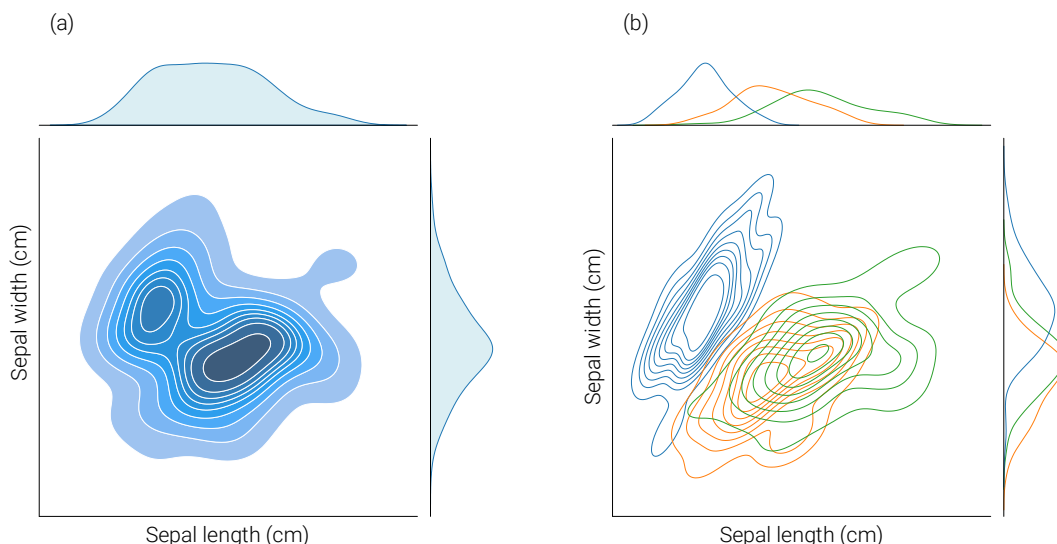


图 26. 鸢尾花花萼长度、花萼宽度的联合分布和边缘分布



什么是边缘分布？

边缘分布是指在多变量数据集中，针对单个变量的分布情况。它表示了某个特定变量在与其他变量无关时的概率分布。边缘分布可以通过将多变量数据集投影到某个特定变量的轴上来获得。通过分析边缘分布，我们可以了解每个变量单独的分布特征，包括均值、方差、偏度、峰度等统计量，以及分布的形状和模式。边缘分布对于探索数据集的特征、进行单变量分析和了解数据的单个方面非常有用。

```
# 联合分布、边缘分布
sns.jointplot(data=iris_sns, x='sepal_length', y='sepal_width',
              kind = 'kde', fill = True)

# 联合分布、边缘分布，考虑鸢尾花分类
sns.jointplot(data=iris_sns, x='sepal_length', y='sepal_width',
              hue = 'species', kind='kde')
```

代码 11. 用 Seaborn 绘制联合分布和边缘分布，使用时须配合前文代码； Bk1_Ch12_01.ipynb

线性回归

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 27 所示为利用 `seaborn.lmplot()` 绘制的鸢尾花花萼长度、花萼宽度之间的线性回归关系图。`seaborn.lmplot()` 函数默认情况下会绘制散点图和拟合的线性回归线。散点图展示了两个变量之间的关系，而线性回归线表示了拟合的线性关系。

除了基本语法外，`seaborn.lmplot()` 还支持其他参数，例如 `hue` 参数用于指定一个额外的分类变量，可以通过不同的颜色展示不同类别的数据点和回归线。请大家自行分析代码 12。



《数据有道》第 9、10 章专门介绍线性回归。

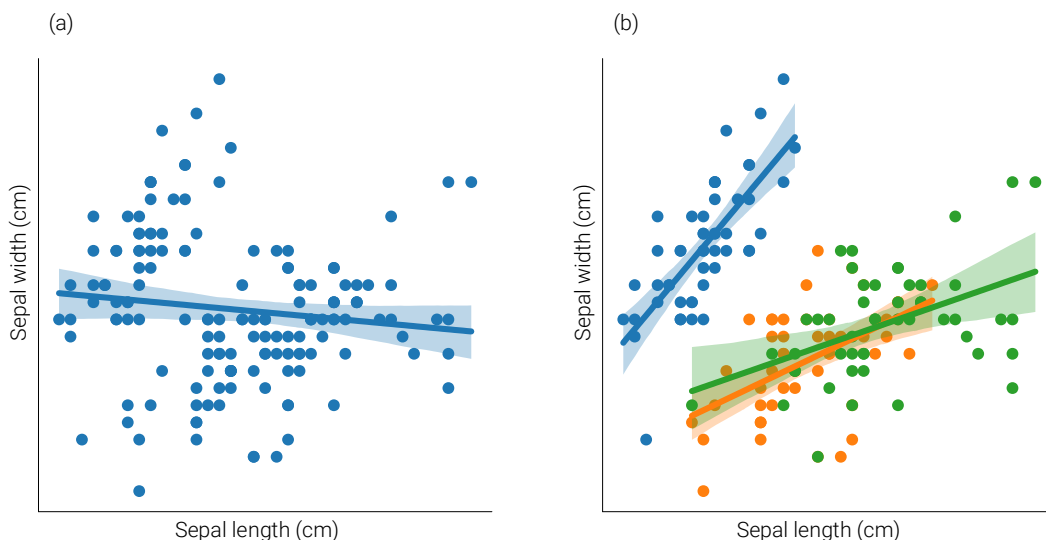



图 27. 鸢尾花花萼长度、花萼宽度的线性回归关系

```
# 可视化线性回归关系
a sns.lmplot(data=iris_sns, x='sepal_length', y='sepal_width')

# 可视化线性回归关系，考虑鸢尾花分类
b sns.lmplot(data=iris_sns, x='sepal_length', y='sepal_width',
             hue = 'species')
```

代码 12. 用 Seaborn 可视化线性回归关系，使用时须配合前文代码；  Bk1_Ch12_01.ipynb

12.4 多元特征数据

分散点图、小提琴图

我们当然可以使用一元可视化方案展示多元数据的特征，如图 28 所示。

代码 13 绘制图 28，下面咱们聊聊其中关键词句。

a 利用 `pandas.melt()`，简作 `pd.melt()`，将鸢尾花数据集从宽格式 (wide format) 转换为长格式 (long format)。

宽格式数据帧如表 1 所示，长格式数据帧如表 2 所示。函数输入 'species' 是要保留的标识变量，也就是不进行融合。参数 `var_name='measurement'` 指定了在融合过程中生成的新列的名称。

请大家自行分析 **b** 和 **c**，并逐行注释。

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



本书第 22 章会介绍包括 `pandas.melt()` 在内的各种常用数据帧规整方法。

但是图 28 这两幅图最致命的缺陷是仅仅展示单个特征分布，并没有展示特征之间的联系。下面我们聊聊其他能够可视化多元特征之间关系的可视化方案。

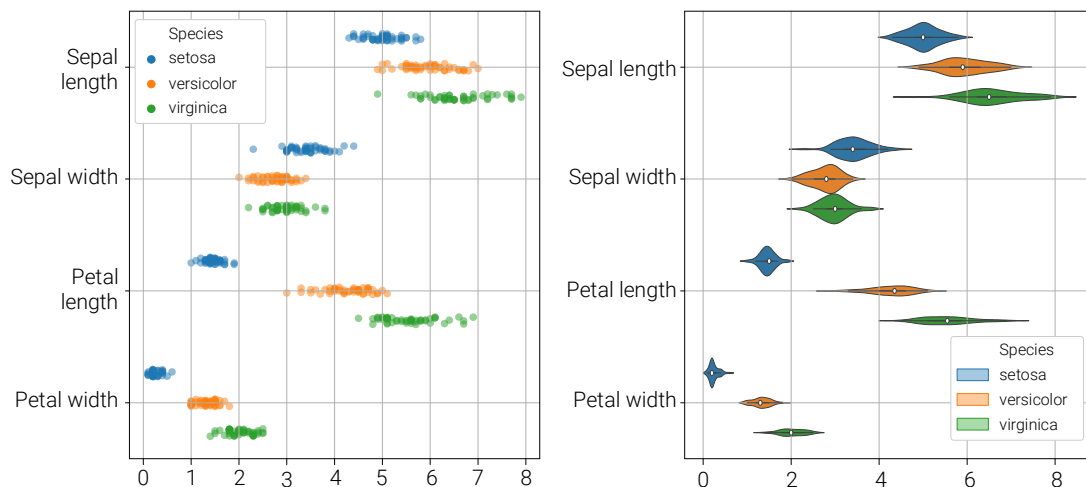


图 28. 分散点图、小提琴图，多特征

```

a iris_melt = pd.melt(iris_sns, 'species', var_name='measurement')
  # 数据从宽格式 (wide format) 转换为长格式 (long format)

  # 绘制多特征分散图
b sns.stripplot(data=iris_melt, x='value', y='measurement',
  hue='species', dodge=True, alpha=.25,
  zorder=1, legend=True)

plt.grid()

  # 绘制多特征小提琴图
c sns.violinplot(data=iris_melt, x='value', y='measurement',
  hue='species', dodge=True, alpha=.25,
  zorder=1, legend=True)

plt.grid()

```


代码 13. 用 Seaborn 绘制多特征分散点图、小提琴图，使用时须配合前文代码； Bk1_Ch12_01.ipynb

表 1. 宽格式

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
...
149	5.9	3	5.1	1.8	virginica

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

表 2. 长格式

	species	measurement	value
0	setosa	sepal_length	5.1
1	setosa	sepal_length	4.9
2	setosa	sepal_length	4.7
...
599	virginica	petal_width	1.8

聚类热图

`seaborn.clustermap()` 函数用于创建聚类热图，它能够可视化数据集中的聚类结构和相似性。聚类热图使用层次聚类算法对数据进行聚类，并以热图的形式展示聚类结果。

聚类热图的原理是通过计算数据点之间的相似性（例如欧几里得距离或相关系数），然后使用**层次聚类**（`hierarchical clustering`）算法将相似的数据点分组为聚类簇。层次聚类将数据点逐步合并并形成聚类树状结构，根据相似性的距离进行聚类的层次化过程。聚类热图将聚类树状结构可视化热图，同时显示数据点的排序和聚类关系。

代码 14 利用 `.iloc[:, :-1]` 方法索引和切片数据帧。简单来说，方法 `iloc` 是 `Pandas DataFrame` 的索引器之一，用于按照整数位置进行选择。第一个冒号 `:` 代表所有行，`:-1` 表示选择除了最后一列之外的所有列。



本书第 21 章将专门介绍 `Pandas` 数据帧索引和切片。



《机器学习》将专门介绍各种聚类算法。

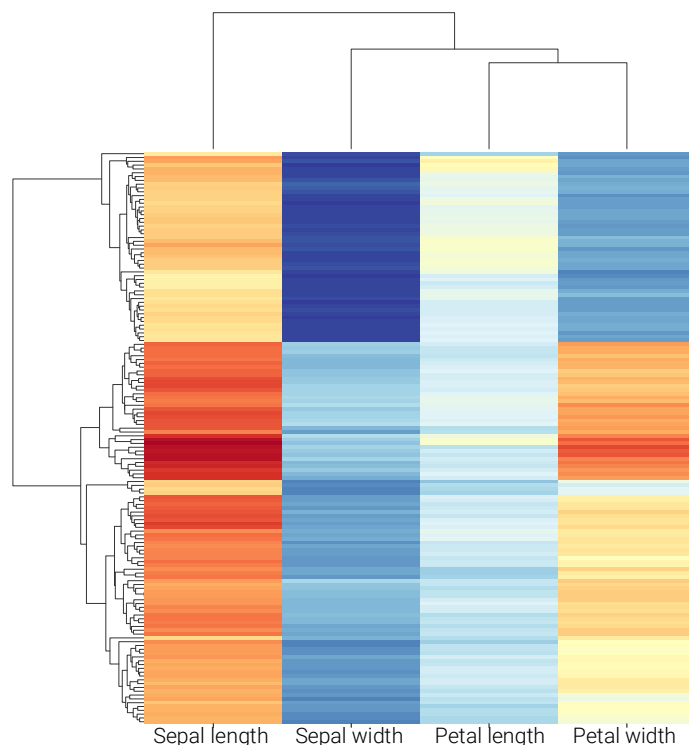


图 29. 鸢尾花数据集，聚类热图

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

```
# 聚类热图
sns.clustermap(iris_sns.iloc[:, :-1], cmap = 'RdYlBu_r',
               vmin = 0, vmax = 8)
```

代码 14. 用 Seaborn 绘制聚类热图，使用时须配合前文代码；  Bk1_Ch12_01.ipynb



什么是聚类？

机器学习中的聚类是一种无监督学习方法，用于将数据集中的样本按照相似性进行分组或聚集。聚类算法通过自动发现数据的内在结构和模式，将相似的样本归为一类，从而实现数据的分组和分类。聚类的目标是使得同一类别内的样本相似度高，而不同类别之间的样本相似度低。聚类算法通常基于样本之间的距离或相似性度量进行操作，例如欧几里得距离、余弦相似度等。常见的聚类算法包括 K 均值聚类、层次聚类、DBSCAN、高斯混合模型等。

成对特征散点图

`seaborn.pairplot()` 函数用于创建成对特征散点图矩阵，可视化多个变量之间的关系和分布。它会将数据集中的每对特征绘制为散点图，并展示变量之间的散点关系和单变量的分布。

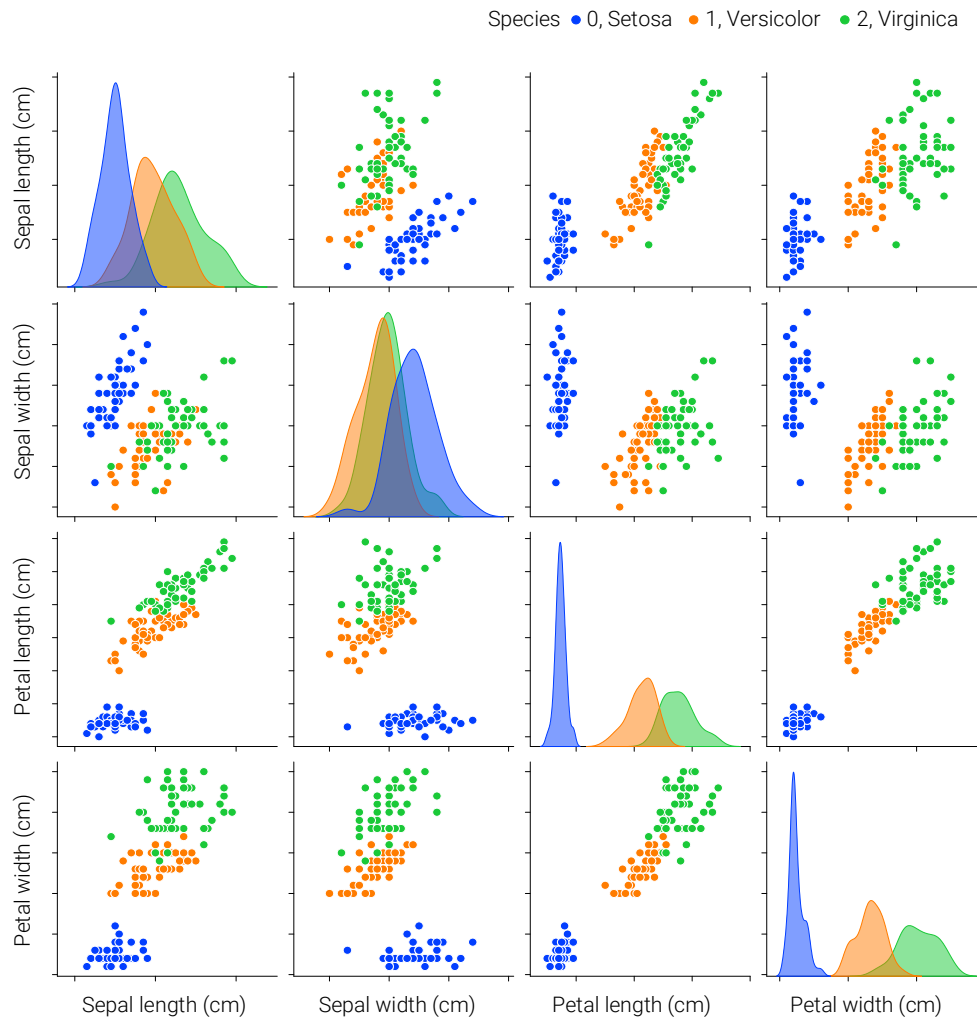


图 30. 鸢尾花数据成对特征散点图，考虑分类标签

代码 15 中，`seaborn.pairplot()` 函数会根据数据集中的每对特征生成散点图，并以网格矩阵的形式展示。对角线上的图形通常是单变量的直方图或核密度估计图，表示每个变量的分布情况。非对角线上的图形是两个变量之间的散点图，展示它们之间的关系。

此外，`seaborn.pairplot()` 函数还支持其他参数，例如 `hue` 参数用于根据一个分类变量对散点图进行颜色编码，使不同类别的数据点具有不同的颜色。

通过使用 `seaborn.pairplot()` 函数，我们可以轻松地可视化多个变量之间的关系和分布。这对于探索变量之间的相关性、识别数据中的模式和异常值等非常有用。

```
# 绘制成对特征散点图
sns.pairplot(iris_sns, hue = 'species')
```

代码 15. 用 Seaborn 绘制成对特征散点图，使用时须配合前文代码； [Bk1_Ch12_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

平行坐标图

平行坐标图 (parallel coordinates plot) 是一种可视化多个连续变量之间关系的图形方法。它使用平行的垂直线段来表示每个变量，这些线段相互平行并沿着水平轴排列。每个变量的值通过垂直线段在对应的轴上进行表示。

在平行坐标图中，每个数据样本由一条连接不同垂直线段的折线表示。这条折线的形状和走势反映了数据样本在不同变量之间的关系。通过观察折线的走势，我们可以识别出变量之间的相对关系，例如正相关、负相关或无关系。同时，我们也可以通过折线的位置和形状来比较不同样本之间的差异。

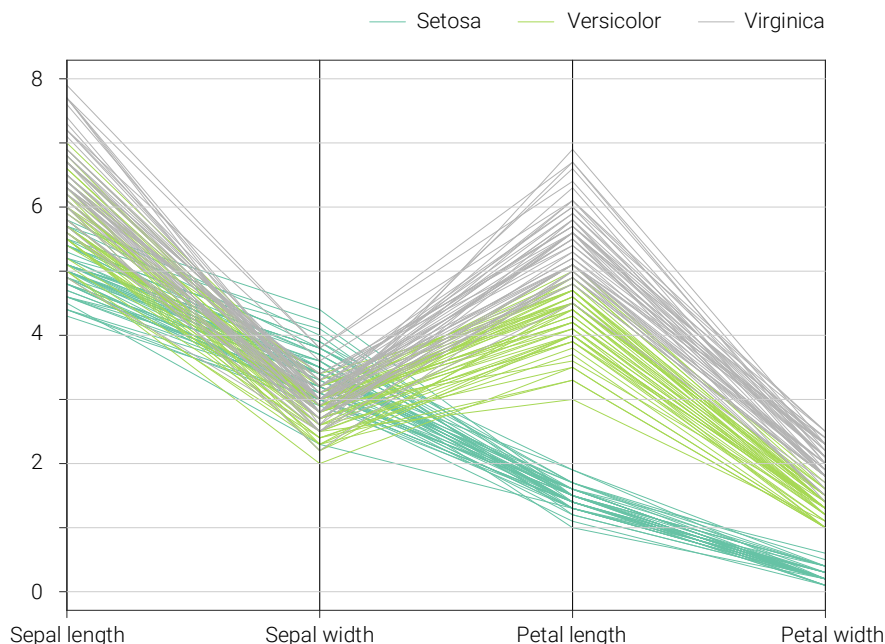



图 31. 鸢尾花数据，平行坐标图

平行坐标图常用于数据探索、特征分析和模式识别等任务。它能够帮助我们发现多个变量之间的关系、观察变量的分布模式，并对数据样本进行可视化比较。此外，通过添加颜色映射或其他可视化元素，还可以在平行坐标图中显示附加信息，例如类别标签或异常值指示。

⚠ 注意，目前 Seaborn 并没有绘制平行坐标图的工具，本章配套的 Jupyter Notebook 中采用的是 `pandas.plotting.parallel_coordinates()` 函数。

```
a from pandas.plotting import parallel_coordinates
# 可视化函数来自pandas
# 绘制平行坐标图
b parallel_coordinates(iris_sns, 'species',
                        colormap=plt.get_cmap("Set2"))
plt.show()
```

代码 16. 用 Pandas 绘制平行坐标图，使用时须配合前文代码；  Bk1_Ch12_01.ipynb

类似平行坐标图的可视化方案还有**安德鲁斯曲线** (Andrews curves)。在安德鲁斯曲线中，每个特征被映射为一个三角函数（通常是正弦函数和余弦函数），并按照给定的顺序排列。本章配套的 Jupyter Notebook 也用 `pandas.plotting.andrews_curves()` 绘制了鸢尾花样本数据的安德鲁斯曲线。

量化多特征样本数据任意两个随机变量关系的最方便的工具莫过于协方差矩阵、相关性系数矩阵。这是上一章已经介绍过的内容，本章不再赘叙。



请大家完成下面 3 道题目。

Q1. 请大家分别绘制鸢尾花花萼宽度、花瓣长度、花瓣宽度的直方图、KDE 概率密度估计。

Q2. 请大家绘制鸢尾花花萼长度、花瓣长度的散点图、二元直方热图、联合分布 KDE 等高线。

Q3. 请大家自行学习本章配套代码 `Bk1_Ch12_02.ipynb`。

* 本章题目不提供答案。



本章主要介绍了 Seaborn 库，这个库特别适合统计可视化。和 Matplotlib 一样，Seaborn 也是提供静态可视化方案。

Plotly 也有大量统计可视化方案，而且都具有交互属性。本书第 23、24 章将结合 Pandas 介绍 Plotly 的统计可视化工具。

本书专门介绍可视化的板块到此结束。鸢尾花书《可视之美》将介绍更多可视化方案。

下一板块用 6 章内容专门介绍 NumPy。

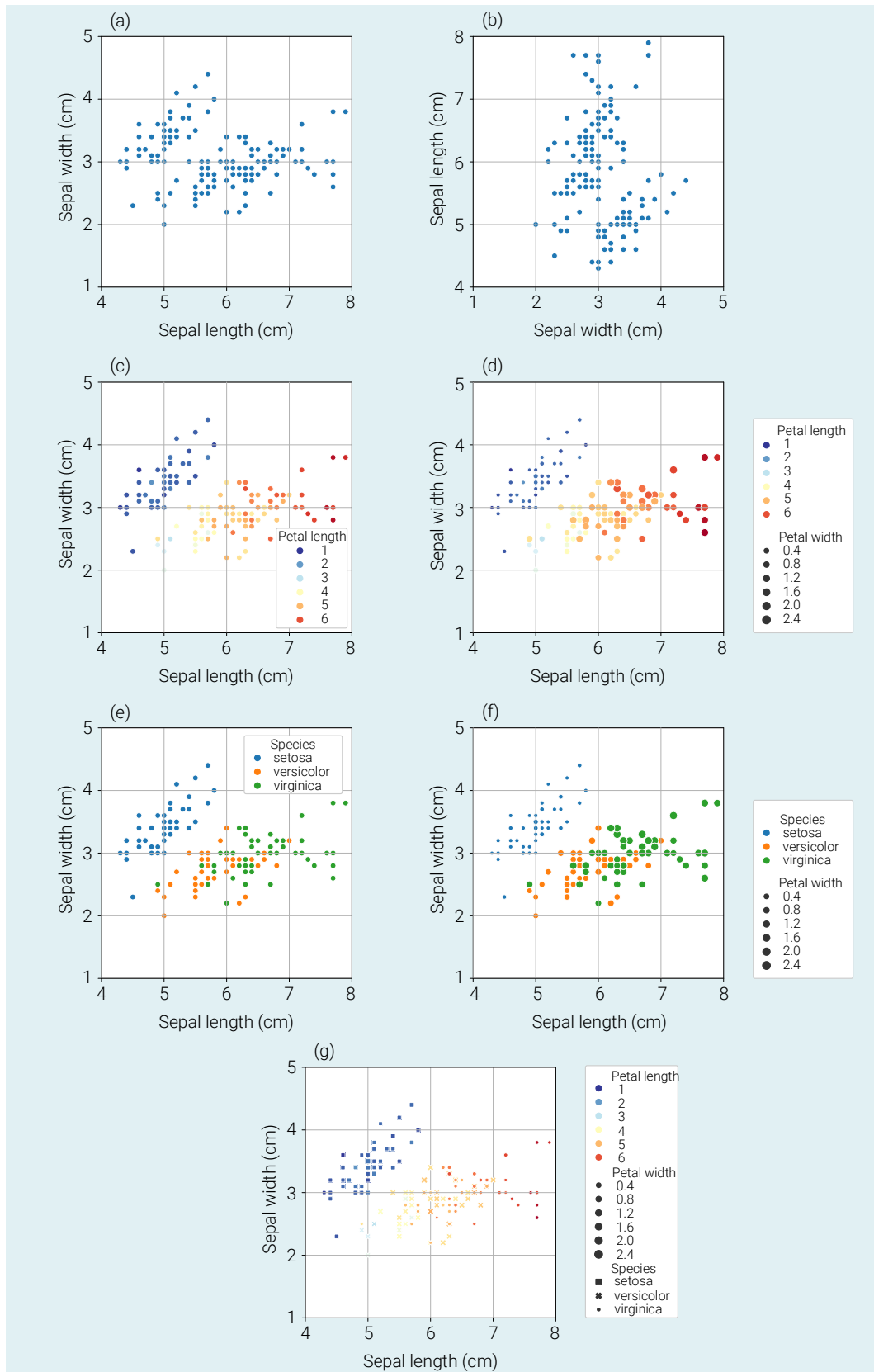


图 32. 几种用 seaborn 绘制的二元散点图; Bk1_Ch12_02.ipynb

整页排版，背景色采用奇数页网格笔记纸背景色，四面出血

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