21 Fundamentals of Statistics 统计入门

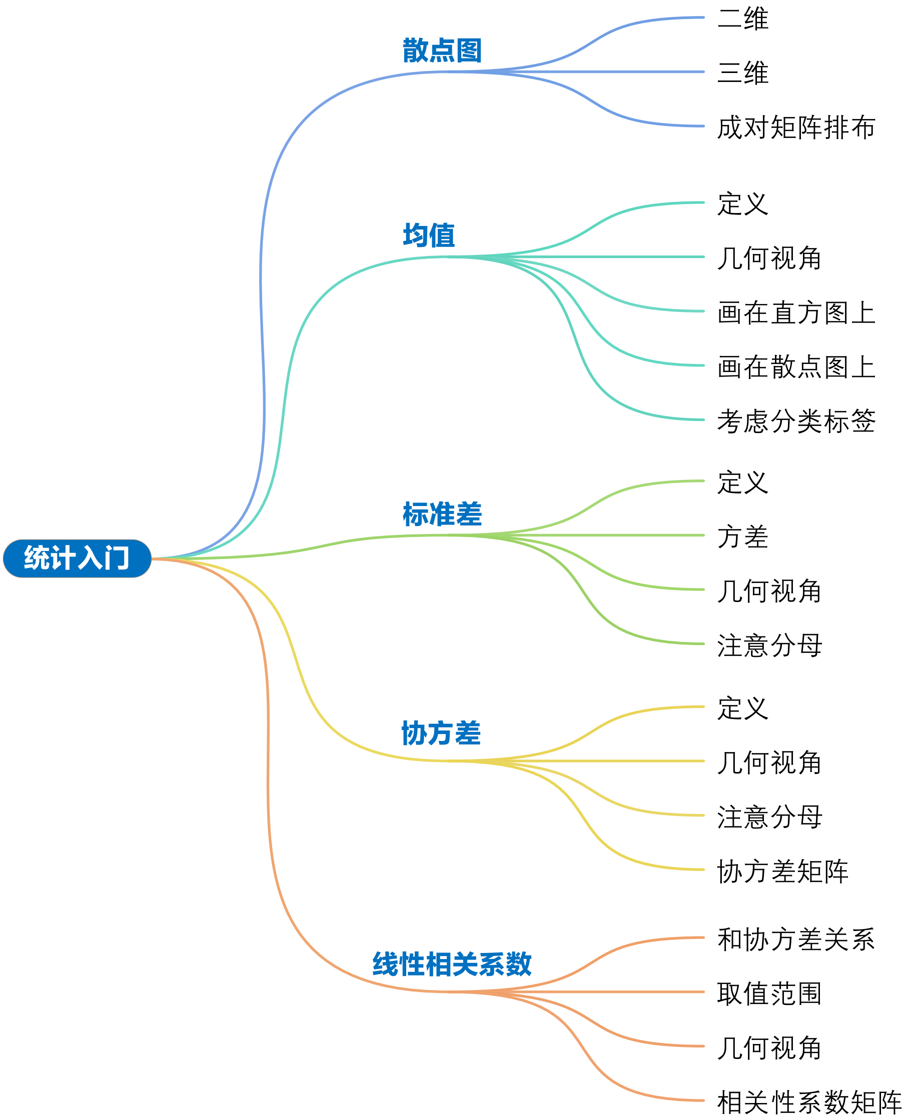
以鸢尾花数据为例

有朝一日，对于所有人，统计思维就像读写能力一样重要。

Statistical thinking will one day be as necessary for efficient citizenship as the ability to read and write.

—— 赫伯特·乔治·威尔斯 (H. G. Wells) | 英国科幻小说家 | 1866 ~ 1946

- `seaborn.heatmap()` 绘制热图
- `seaborn.histplot()` 绘制频率/概率直方图
- `seaborn.pairplot()` 绘制成对分析图
- `seaborn.lineplot()` 绘制线图


统计入门
散点图
二维
三维
成对矩阵排布
均值
定义
几何视角
画在直方图上
画在散点图上
考虑分类标签
标准差
定义
方差
几何视角
注意分母
协方差
定义
几何视角
注意分母
协方差矩阵
线性相关系数
和协方差关系
取值范围
几何视角
相关性系数矩阵

21.1 统计的前世今生：强国知十三数

现在，“概率”和“统计”两个词如影随形。统计搜集、整理、分析、研究数据，从而寻找规律；概率论是统计推断的基础，基于特定条件，概率预测事件的可能性。

现代统计学的主要数学基础是概率论；但是，统计的出现远早于概率。

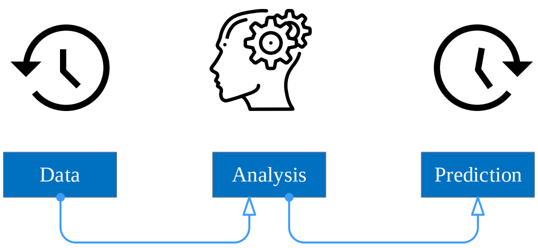


图 1. 统计和概率关系

通过上一章学习，我们了解了概率出生草莽；但是，统计学却是衔玉而生。

统计学的初衷就是为国家管理提供可靠数据。英语中 statistics 是源于现代拉丁语 statisticum collegium (国会)。

战国思想家商鞅 (390 BC ~ 338 BC) 提出“强国知十三数”，他为秦国制定的统计内容包含“十三数”——“竟内仓、口之数，壮男、壮女之数，老、弱之数，官、士之数，以言说取食者之数，利民之数，马、牛、刍藁之数。欲强国，不知国十三数，地虽利，民虽众，国愈弱至削。”

简单说，商鞅认为和国家存亡攸关的统计数字包括粮仓、金库、壮年男子、壮年女子、老年人、体弱者、官吏、士卒、游说者、工商业者、牲畜和饲料。刍藁 (chú gǎo) 为饲养牲畜的草料。

商鞅强调统计数字对王朝兴亡至关重要。他说，“数者，臣主之术而国之要也。故万国失数而国不危，臣主失数而不乱者，未之有也。”大意是，统计数字是治国之术和国家根本；没有统计数字，君主便无法治国理政，国家就要危乱。

阿拉伯学者肯迪 (Al-Kindi, 801 ~ 873) 创作的《密码破译》 (Manuscript on Deciphering Cryptographic Messages) 书中，介绍如何使用统计数据和频率分析进行密码破译。肯迪和本书前文介绍的花拉子密 (Muhammad ibn Musa al-Khwarizmi) 都供职于巴格达“智慧宫 (House of Wisdom)”。

英国经济学家约翰·葛兰特 (John Graunt, 1620 ~ 1674) 在 1663 年发表了 《对死亡率表的自然与政治观察》 (Natural and Political Observations Made Upon the Bills of Mortality)，被誉为人口统计学的开山之作，他本人也常被称作“人口统计学之父”。

本章内容以鸢尾花数据为例，用最少的公式，尽量从几何可视化视角给大家介绍统计的入门知识。

21.2 散点图：当数据遇到坐标系

本书前文以表格的形式介绍过鸢尾花数据；有了坐标系，类似鸢尾花这样的样本数据就可以在纸面飞跃。

本节介绍样本数据重要的可视化方案之一——散点图 (scatter plot)。散点图将样本数据以点的形式展现在直角坐标系上。

图 2 (a) 所示为鸢尾花数据中花萼长度和花萼宽度两个特征的散点图。散点图中每一个点代表一朵鸢尾花，横坐标值代表花萼长度，纵坐标值代表花萼宽度。

我们知道鸢尾花数据集一共有 150 个数据点，分成 3 大类，也就是对应 3 个不同的标签；在图 2 (a) 散点图基础上，用不用颜色区分分类标签，我们可以得到图 2 (b)。

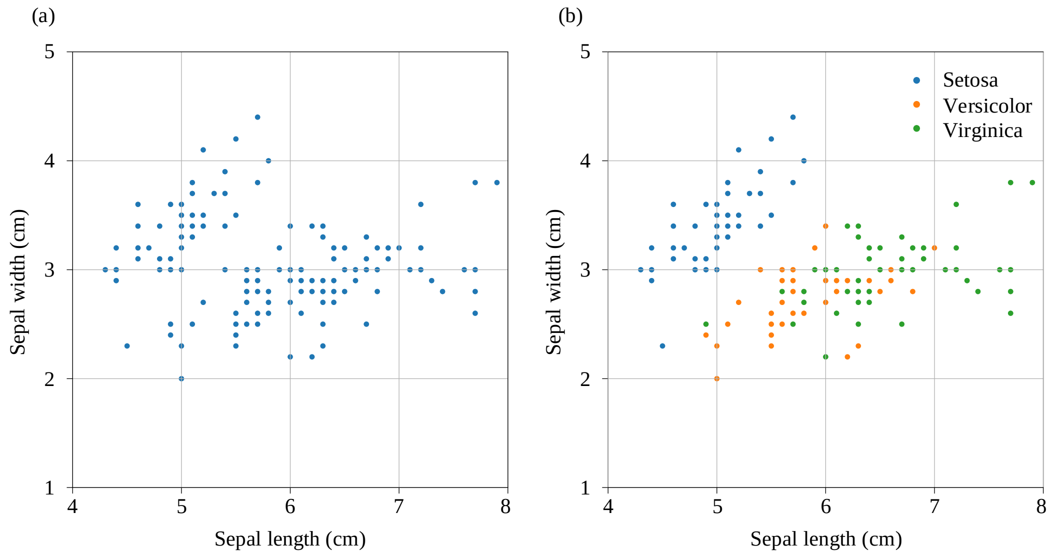


图 2. 花萼长度、花萼宽度特征数据散点图

我们也可以在三维直角坐标系中绘制散点图；图 3 (a) 所示为花萼长度、花萼宽度、花瓣长度三个特征的散点图；在图 3 (a) 基础上，如果加上分类标签，我们可以得到图 3 (b)。

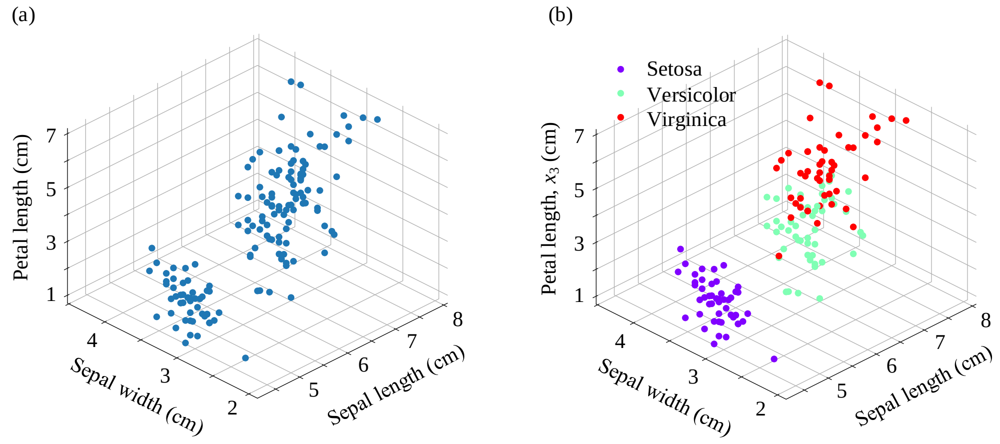


图 3. 花萼长度、花萼宽度、花瓣长度散点图

成对特征散点图

大家可能会问，鸢尾花有 4 个特征 (花萼长度、花萼宽度、花瓣长度、花瓣宽度)；有没有什么可视化方案能够展示所有的特征？

答案是成对特征散点图。

如图 4 所示，16 幅子图被安排成 4 × 4 矩阵的形式；12 幅散点图为成对特征关系；对角线上的 4 幅图像叫做概率密度估计 (probability density estimation) 曲线。

简单来说，概率密度估计曲线展示数据分布情况，类似于上一章介绍的频率直方图。本系列丛书《概率统计》一册将专门讲解概率密度估计。

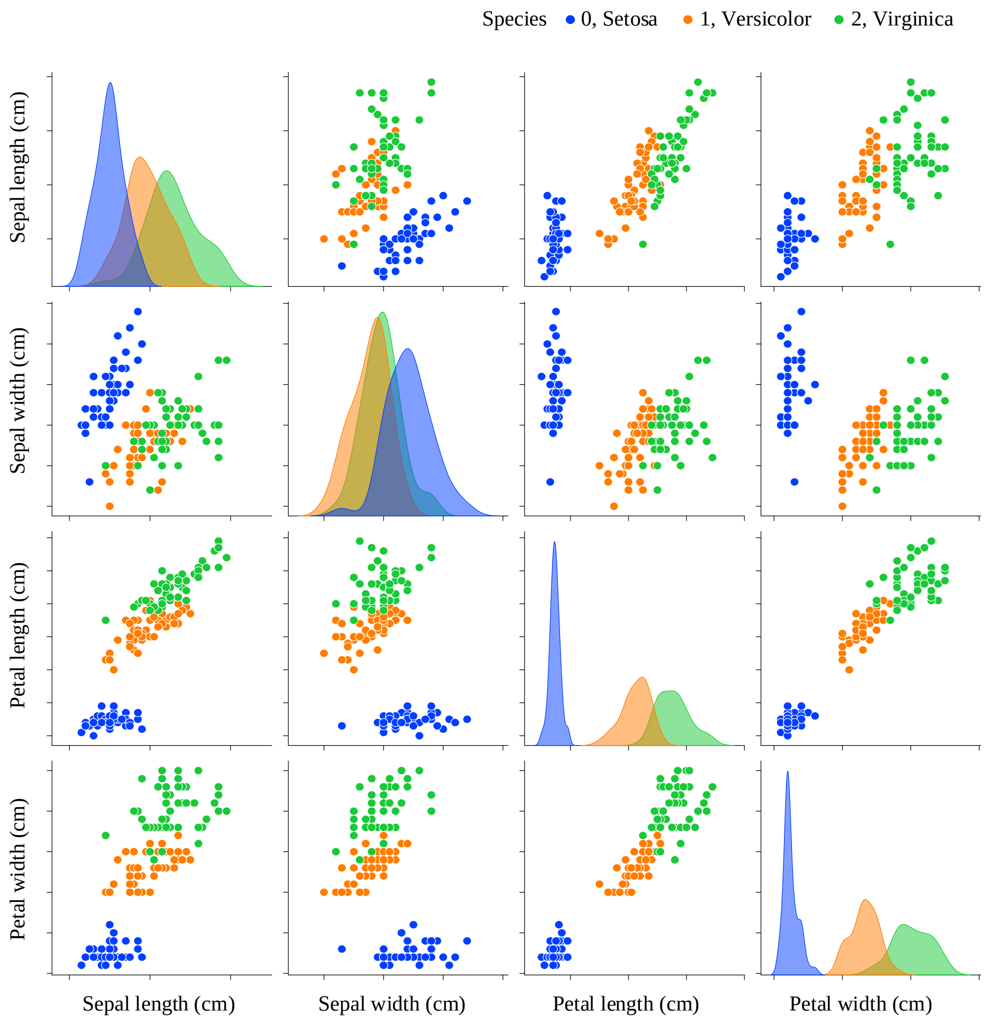


图 4. 鸢尾花数据成对特征散点图，考虑分类标签

散点图的作用

利用散点图，我们可以发现数据的集中、分布程度，比如数据主要集中在哪些区域。

散点图也会揭示不同特征之间可能存在的量化关系，比如图 4 中花瓣长度和宽度数据关系似乎能够用一条直线来表达；这就是线性回归 (linear regression) 的思路。

此外，我们还可以利用散点图发现数据是否存在离群值。离群值 (outlier) 指的是，和其他数据相比，数据中有一个或几个样本数值差异较大。本系列丛书《数据科学》一册将讲解常用发现数据中离群值的算法。

本节采用可视化的方式来描绘数据，实际应用中，我们经常需要量化数据的集中、分散程度，以及不同特征之间的关系；这就需要大家了解均值、方差、标准差、协方差、相关性这些概念。这是本章后续要介绍的内容。

代码文件 Bk3_Ch21_1.py 中 Bk3_Ch21_1_A 部分绘制本节图像。

```python
# Bk3_Ch21_1_A

import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
from sklearn.datasets import load_iris

# Load the iris data
iris_sns = sns.load_dataset("iris")

#%% Scatter plot of x1 and x2
fig, ax = plt.subplots()

ax = sns.scatterplot(data=iris_sns, x="sepal_length", y="sepal_width")

ax.set_xlabel('Sepal length, $x_1$ (cm)')
ax.set_ylabel('Sepal width, $x_2$ (cm)')
ax.set_xticks(np.arange(4, 8 + 1, step=1))
ax.set_yticks(np.arange(1, 5 + 1, step=1))
ax.axis('scaled')
ax.grid(linestyle='--', linewidth=0.25, color=[0.7,0.7,0.7])
ax.set_xbound(lower = 4, upper = 8)
ax.set_ybound(lower = 1, upper = 5)

fig, ax = plt.subplots()

ax = sns.scatterplot(data=iris_sns, x="sepal_length",
                     y="sepal_width", hue = "species")

ax.set_xlabel('Sepal length, $x_1$ (cm)')
ax.set_ylabel('Sepal width, $x_2$ (cm)')
ax.set_xticks(np.arange(4, 8 + 1, step=1))
ax.set_yticks(np.arange(1, 5 + 1, step=1))
ax.axis('scaled')
ax.grid(linestyle='--', linewidth=0.25, color=[0.7,0.7,0.7])
ax.set_xbound(lower = 4, upper = 8)
ax.set_ybound(lower = 1, upper = 5)

#%% 3D scatter plot

x1=iris_sns['sepal_length']
x2=iris_sns['sepal_width']
x3=iris_sns['petal_length']
labels = iris_sns['species'].copy()
labels[labels == 'setosa']      =1
labels[labels == 'versicolor'] =2
labels[labels == 'virginica']  =3
rainbow = plt.get_cmap("rainbow")

fig = plt.figure()
ax = fig.gca(projection='3d')
ax.scatter(x1, x2, x3)

ax.set_xlabel('Sepal length, $x_1$ (cm)')
ax.set_ylabel('Sepal width, $x_2$ (cm)')
ax.set_zlabel('Petal length, $x_3$ (cm)')
plt.show()
ax.set_proj_type('ortho')
ax.view_init(azim=-135, elev=30)

fig = plt.figure()
ax = fig.gca(projection='3d')
```

```
scatter_h = ax.scatter(x1, x2, x3, c = labels, cmap=rainbow)

classes = ['Setosa', 'Versicolor', 'Virginica']
plt.legend(handles=scatter_h.legend_elements()[0], labels=classes)
ax.set_xlabel('Sepal length, $x_1$ (cm)')
ax.set_ylabel('Sepal width, $x_2$ (cm)')
ax.set_zlabel('Petal length, $x_3$ (cm)')
plt.show()
ax.set_proj_type('ortho')
ax.view_init(azim=-135, elev=30)

#%% pairwise plot
sns.pairplot(iris_sns)
sns.pairplot(iris_sns, hue = 'species')
```

21.3 均值：集中程度

大家对均值这个概念应该不陌生。

均值 (average 或 mean)，也叫平均值，算数平均数 (as arithmetic average 或 arithmetic mean)。均值代表一组数据集中趋势。

均值对应的运算是，一组数据中所有数据先求和，再除以这组数据的个数。比如鸢尾花花萼特征数据 $\left\{x_1^{(1)}, x_1^{(2)}, \ldots, x_1^{(150)}\right\}$ 有 150 个值，它们的平均值为：

$$\mu_1 = \frac{1}{n}\left(\sum_{i=1}^{n} x_1^{(i)}\right) = \frac{x_1^{(1)} + x_1^{(2)} + \cdots + x_1^{(150)}}{150} \tag{1}$$

从几何角度，如图 5 所示，算数平均值相当于找到一个平衡点。

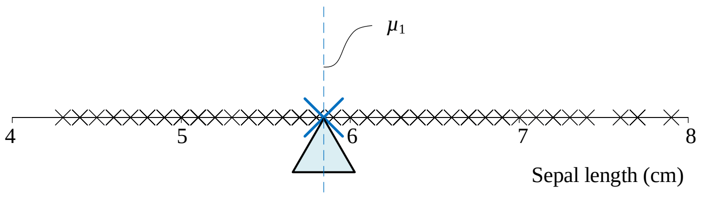


图 5. 均值相当于找到数据的平衡点

以鸢尾花为例，它的样本数据在花萼长度、花萼宽度、花瓣长度和花瓣宽度四个特征的均值分别为：

$$\mu_1 = 5.843,\quad \mu_2 = 3.057,\quad \mu_3 = 3.758,\quad \mu_4 = 1.199 \tag{2}$$

图 6 所示为鸢尾花四个特征均值在频数直方图位置。注意在计算这四个均值时，我们并没有考虑鸢尾花的分类标签。

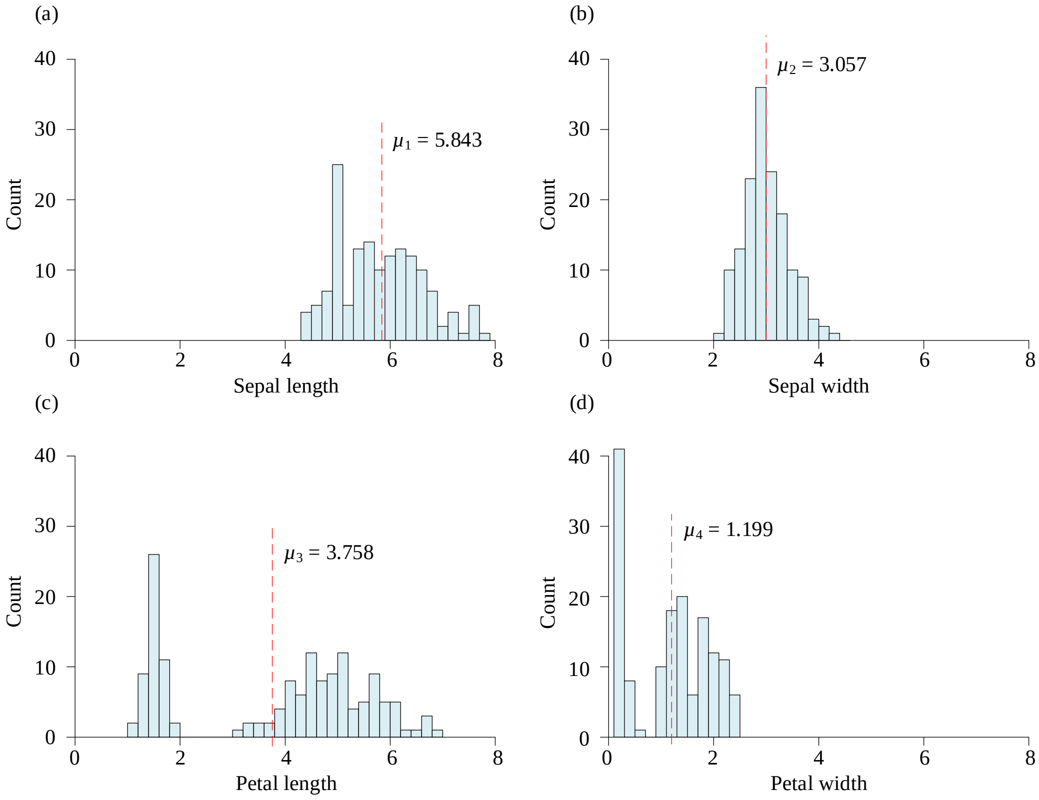


图 6. 鸢尾花四个特征数据均值在直方图位置

考虑分类

当然，我们在计算均值的时候，也可以考虑分类。

以鸢尾花数据为例，很多应用场合需要计算满足某个条件的均值，比如标签为 virginica 样本数据的花萼长度。

在图 4 基础上，我们可以三类不同标签条样本数据均值位置可视化，这样便得到图 7。图 7 中 ×、×、× 分别代表 setosa、versicolor、virginica 三个不同标签均值的位置。

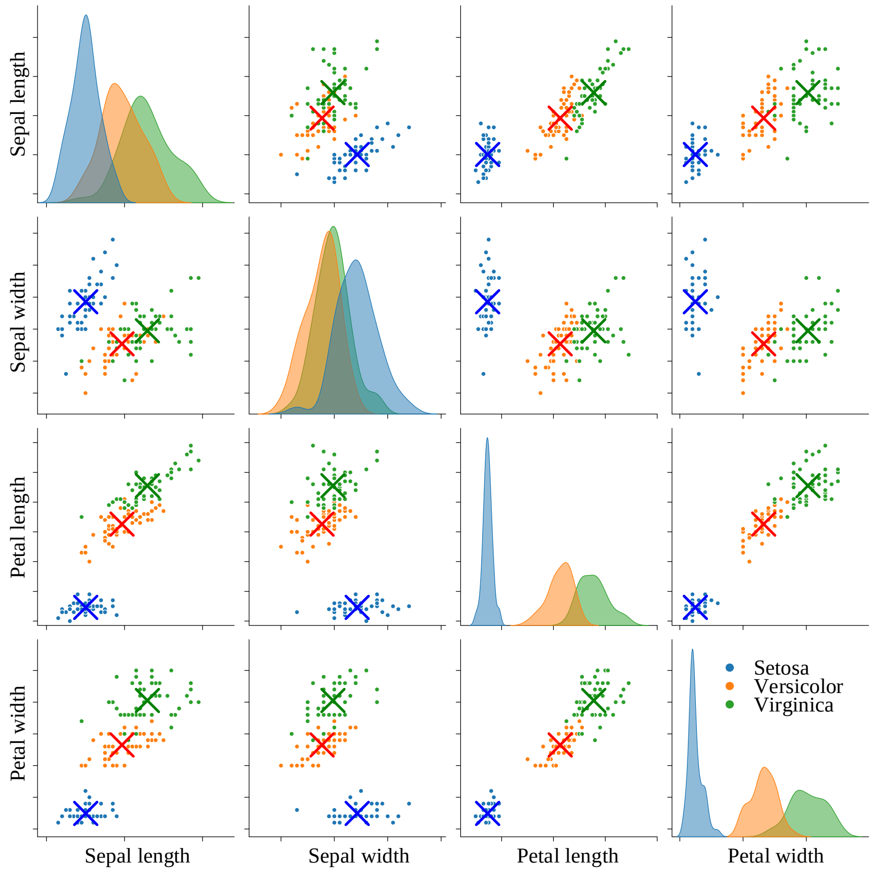


图 7. 均值在散点图上的位置，考虑分类标签

代码文件 Bk3_Ch21_1.py 中 Bk3_Ch21_1_B 部分计算均值并绘制图 6。

```
# Bk3_Ch21_1_B

#%% add mean values to the histograms

fig, axes = plt.subplots(2,2)

mu_1 = iris_sns['sepal_length'].mean()
sns.histplot(data=iris_sns, x = 'sepal_length', binwidth = 0.2, ax = axes[0][0])
axes[0][0].set_xlim([0,8]); axes[0][0].set_ylim([0,40])
axes[0][0].vlines(x = mu_1,
                  ymin = 0, ymax = 40, color = 'r')

mu_2 = iris_sns['sepal_width'].mean()
sns.histplot(data=iris_sns, x = 'sepal_width', binwidth = 0.2, ax = axes[0][1])
axes[0][1].set_xlim([0,8]); axes[0][1].set_ylim([0,40])
axes[0][1].vlines(x = mu_2,
                  ymin = 0, ymax = 40, color = 'r')

mu_3 = iris_sns['petal_length'].mean()
sns.histplot(data=iris_sns, x = 'petal_length', binwidth = 0.2, ax = axes[1][0])
axes[1][0].set_xlim([0,8]); axes[1][0].set_ylim([0,40])
axes[1][0].vlines(x = mu_3,
```

```
                   ymin = 0, ymax = 40, color = 'r')

mu_4 = iris_sns['petal_width'].mean()
sns.histplot(data=iris_sns, x = 'petal_width', binwidth = 0.2, ax = axes[1][1])
axes[1][1].set_xlim([0,8]); axes[1][1].set_ylim([0,40])
axes[1][1].vlines(x = mu_4,
                   ymin = 0, ymax = 40, color = 'r')
```

21.4 标准差：离散程度

标准差 (standard deviation) 描述一组数值以均值 μ 为基准的分散程度。如果数据为样本，比如鸢尾花花萼数据 $\left\{x_1^{(1)}, x_1^{(2)}, \dots, x_1^{(150)}\right\}$ 标准差为：

$$\sigma_1 = \sqrt{\frac{1}{150-1}\sum_{i=1}^{150}\left(x_1^{(i)} - \mu_1\right)^2} \tag{3}$$

注意，分母为 (150 – 1)，不是 150。

标准差的平方为方差 (variance)：

$$\operatorname{var}(X_1) = \sigma_1^2 = \frac{1}{150-1}\sum_{i=1}^{150}\left(x_1^{(i)} - \mu_1\right)^2 \tag{4}$$

如图 8 所示， $x_1^{(i)} - \mu_1$ 代表 $x_1^{(i)}$ 和 μ_1 距离；而 $\left(x_1^{(i)} - \mu_1\right)^2$ 代表以 $\left|x_1^{(i)} - \mu_1\right|$ 为边长正方形的面积。(4) 相当于这些正方形面积求平均值。

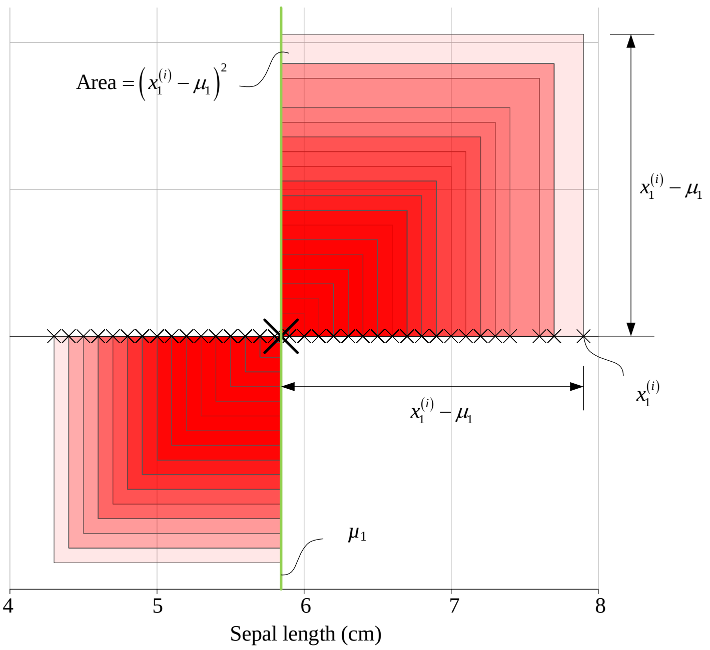


图 8. 几何视角看方差

注意，标准差的单位和样本数据相同；但是方差的单位是样本数据单位的平方。比如，鸢尾花花萼长度的单位是厘米 cm，因此这个特征上样本数据的标准差对应的单位也是厘米 cm，而方差的单位是平方厘米 cm^2。所以在同一幅图上，我们常会看到 μ、$\mu \pm \sigma$、$\mu \pm 2\sigma$、$\mu \pm 3\sigma$ 等。

计算鸢尾花样本数据四个特征的标准差：

$$\sigma_1 = 0.825, \quad \sigma_2 = 0.434, \quad \sigma_3 = 1.759, \quad \sigma_4 = 0.759 \tag{5}$$

图 9 所示为鸢尾花四个特征数据均值 μ、标准差 $\mu \pm \sigma$ 在频数直方图位置。

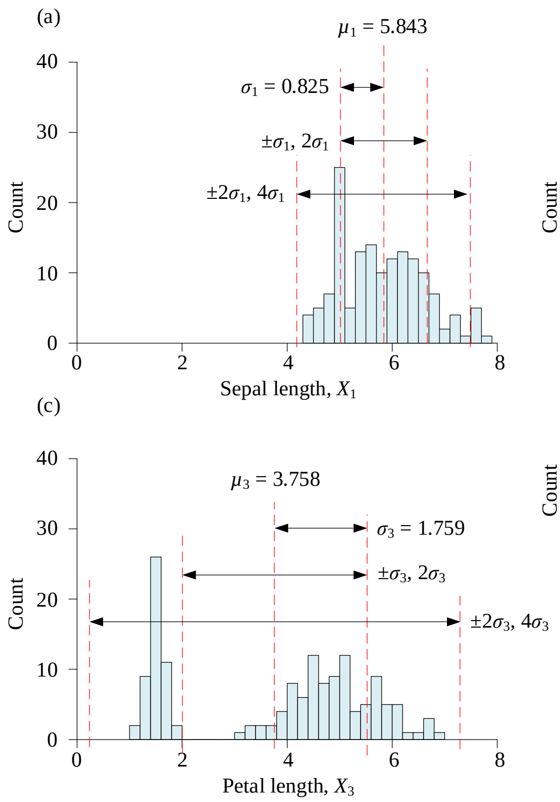


图 9. 鸢尾花四个特征数据均值、标准差在直方图位置

代码文件 Bk3_Ch21_1.py 中 Bk3_Ch21_1_C 部分计算标准差并绘制图 9。

```
# Bk3_Ch21_1_C

#%% add mean values and std bands to the histograms

num = 0
fig, axes = plt.subplots(2,2)
for i in [0,1]:
```

```
    for j in [0,1]:

        sns.histplot(data=iris_sns, x = iris_sns.columns[num],
                     binwidth = 0.2, ax = axes[i][j])
        axes[i][j].set_xlim([0,8]); axes[0][0].set_ylim([0,40])

        mu  = iris_sns.iloc[:,num].mean()
        std = iris_sns.iloc[:,num].std()

        axes[i][j].axvline(x=mu,
                           color = 'r', linestyle = '--')
        axes[i][j].axvline(x=mu - std,
                           color = 'r', linestyle = '--')
        axes[i][j].axvline(x=mu + std,
                           color = 'r', linestyle = '--')
        axes[i][j].axvline(x=mu - 2*std,
                           color = 'r', linestyle = '--')
        axes[i][j].axvline(x=mu + 2*std,
                           color = 'r', linestyle = '--')
        num = num + 1
```

21.5 协方差：联合变化程度

协方差 (covariance) 描述的是随机变量联合变化程度。白话讲，以图 4 中花瓣长度和宽度数据关系为例，我们发现如果样本数据的花瓣长度越长，其花瓣宽度很大可能性也越宽；这就是联合变化。而协方差以量化的方式来定量分析这种联合变化程度。

定义第 i 朵花的花萼长度和花萼宽度的取值为 $\left(x_1^{(i)}, x_2^{(i)}\right)$ $(i = 1, \ldots, 150)$，花萼长度和宽度的协方差为：

$$\operatorname{cov}\left(X_1, X_2\right) = \frac{1}{150-1}\sum_{i=1}^{150}\left(x_1^{(i)} - \mu_1\right)\left(x_2^{(i)} - \mu_2\right) \tag{6}$$

如图 10 所示，从几何视角，$\left(x_1^{(i)} - \mu_1\right)\left(x_2^{(i)} - \mu_2\right)$ 相当于以 $\left(x_1^{(i)} - \mu_1\right)$ 和 $\left(x_2^{(i)} - \mu_2\right)$ 为边的矩形面积，注意这个面积有正负。

当 $\left(x_1^{(i)} - \mu_1\right)$ 和 $\left(x_2^{(i)} - \mu_2\right)$ 同号，面积为正，对应图 10 中红色矩形。也就是说，红色矩形越多说明，花萼长度越长，花萼宽度越宽；或者，花萼长度越短，花萼宽度越窄。

当 $\left(x_1^{(i)} - \mu_1\right)$ 和 $\left(x_2^{(i)} - \mu_2\right)$ 异号，面积为负，对应图 10 中蓝色矩形。蓝色矩形越多说明，花萼长度越长，花萼宽度越窄；花萼长度越短，花萼宽度越长。

这些矩形的面积的平均值便是协方差；同样在计算协方差时，对于样本，分母为 $n-1$；对于总体，分母为 n。

可以这样理解，当 X_1 和 X_2 联合变化越强，某个颜色 (红色或蓝色) 矩形面积之和越大；当 X_1 和 X_2 联合变化弱的时候，红色和蓝色矩形面积之和越趋向于 0，也就是颜色越“平衡”。

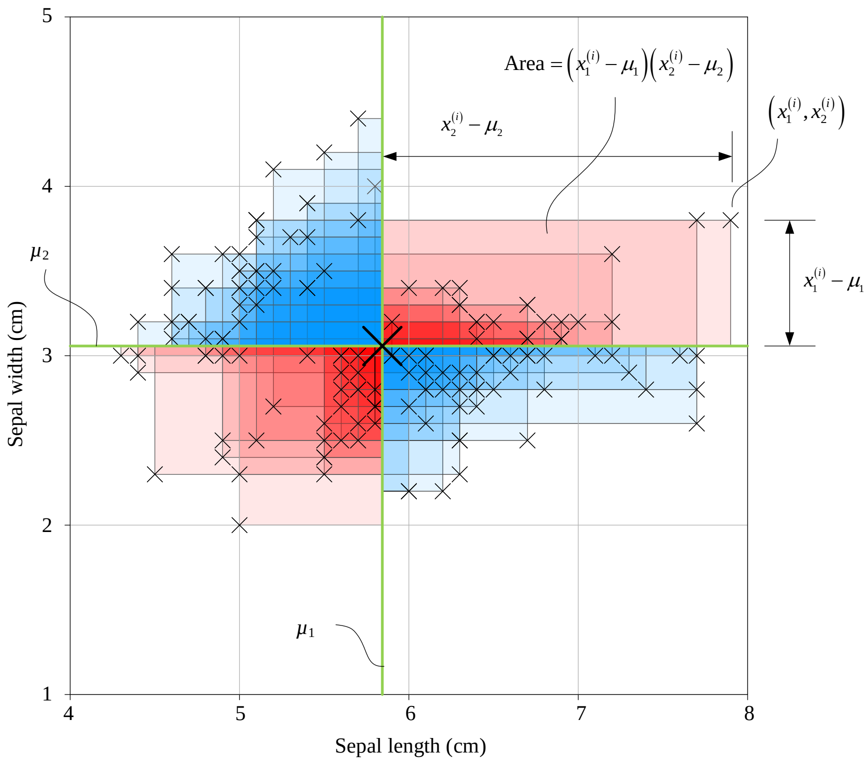


图 10. 几何视角看协方差

协方差矩阵

以鸢尾花为例，对于不同成对的特征，我们可以获得如下 6 (对应组合数 C_4^2) 个协方差值：

$$\begin{aligned}
\operatorname{cov}\left(X_1,X_2\right)&=-0.042\\
\operatorname{cov}\left(X_1,X_3\right)&=1.274\\
\operatorname{cov}\left(X_1,X_4\right)&=0.516\\
\operatorname{cov}\left(X_2,X_3\right)&=-0.330\\
\operatorname{cov}\left(X_2,X_4\right)&=-0.122\\
\operatorname{cov}\left(X_3,X_4\right)&=1.296
\end{aligned} \tag{7}$$

可以想象，如果我们有更多的特征，成对协方差值不计其数。整理和储存这些数据需要很好的结构。矩阵就是最好的解决办法。

由方差和协方差构成的矩阵叫做**协方差矩阵** (covariance matrix)，也叫方差-协方差矩阵 (variance-covariance matrix)；以鸢尾花四个特征为例，这个协方差矩阵为 4 × 4 矩阵：

$$\boldsymbol{\Sigma}=\left[\begin{matrix}
\operatorname{cov}\left(X_1,X_1\right) & \operatorname{cov}\left(X_1,X_2\right) & \operatorname{cov}\left(X_1,X_3\right) & \operatorname{cov}\left(X_1,X_4\right)\\
\operatorname{cov}\left(X_2,X_1\right) & \operatorname{cov}\left(X_2,X_2\right) & \operatorname{cov}\left(X_2,X_3\right) & \operatorname{cov}\left(X_2,X_4\right)\\
\operatorname{cov}\left(X_3,X_1\right) & \operatorname{cov}\left(X_3,X_2\right) & \operatorname{cov}\left(X_3,X_3\right) & \operatorname{cov}\left(X_3,X_4\right)\\
\operatorname{cov}\left(X_4,X_1\right) & \operatorname{cov}\left(X_4,X_2\right) & \operatorname{cov}\left(X_4,X_3\right) & \operatorname{cov}\left(X_4,X_4\right)
\end{matrix}\right] \tag{8}$$

协方差矩阵为方阵。矩阵中对角线上元素为方差。也就是说，某个随机变量和自身求协方差，得到的就是方差，比如下例：

$$\operatorname{cov}(X_1, X_1) = \operatorname{var}(X_1) \tag{9}$$

协方差矩阵中非对角线上元素为协方差。容易知道，下式成立：

$$\operatorname{cov}(X_i, X_j) = \operatorname{cov}(X_j, X_i) \tag{10}$$

也就是说，协方差矩阵为对称矩阵。

对于鸢尾花数据，它的协方差矩阵 $\boldsymbol{\Sigma}$ 具体值为：

$$\boldsymbol{\Sigma} = \begin{array}{c} \left[\begin{array}{cccc} 0.686 & -0.042 & 1.274 & 0.516 \\ -0.042 & 0.190 & -0.330 & -0.122 \\ 1.274 & -0.330 & 3.116 & 1.296 \\ 0.516 & -0.122 & 1.296 & 0.581 \end{array}\right] \\ \begin{array}{cccc} \text{Sepal length, } X_1 & \underbrace{}_{\text{Sepal width, } X_2} & \text{Petal length, } X_3 & \text{Petal width, } X_4 \end{array} \end{array} \begin{array}{l} \leftarrow \text{Sepal length, } X_1 \\ \leftarrow \text{Sepal width, } X_2 \\ \leftarrow \text{Petal length, } X_3 \\ \leftarrow \text{Petal width, } X_4 \end{array} \tag{11}$$

图 14 所示为鸢尾花数据协方差矩阵热图。

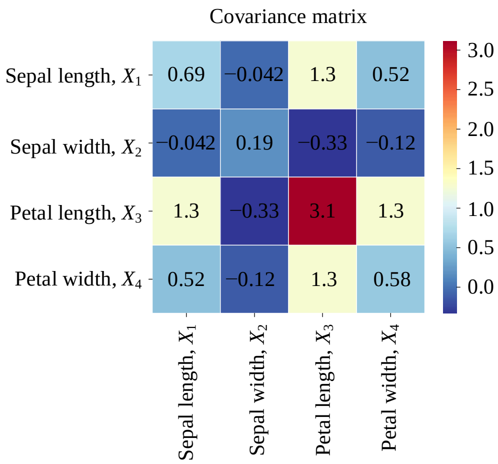


图 11. 鸢尾花数据协方差矩阵热图

考虑标签

当然，在计算协方差时，我们也可以考虑到数据标签。图 12 所示为三个不同标签数据各自的协方差矩阵热图。

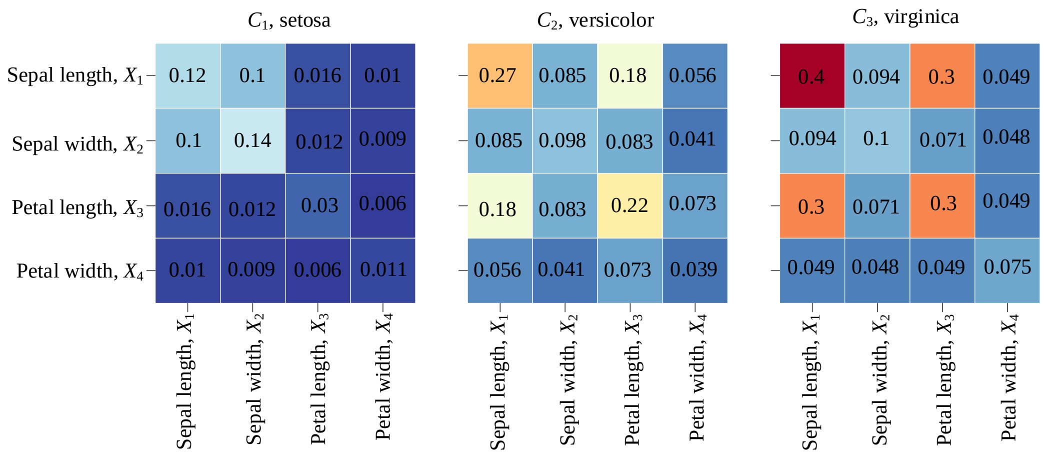


图 12. 协方差矩阵热图，考虑分类

代码文件 Bk3_Ch21_1.py 中 Bk3_Ch21_1_D 部分绘制本节热图。

```
# Bk3_Ch21_1_D

#%% covariance matrix

SIGMA = iris_sns.cov()

fig, axs = plt.subplots()

h = sns.heatmap(SIGMA,cmap='RdBu_r', linewidths=.05, annot = True)
h.set_aspect("equal")
h.set_title('$\Sigma$')

#%% compare covariance matrices, with class labels

f,(ax1,ax2,ax3) = plt.subplots(1,3,sharey=True)

g1 = sns.heatmap(iris_sns.loc[iris_sns['species'] == 'setosa'].cov(),
                 cmap="RdYlBu_r",
                 annot=True,cbar=False,ax=ax1,square=True,
                 vmax = 0.4, vmin = 0)
ax1.set_title('Y = 0, setosa')

g2 = sns.heatmap(iris_sns.loc[iris_sns['species'] == 'versicolor'].cov(),
                 cmap="RdYlBu_r",
                 annot=True,cbar=False,ax=ax2,square=True,
                 vmax = 0.4, vmin = 0)
ax2.set_title('Y = 1, versicolor')

g3 = sns.heatmap(iris_sns.loc[iris_sns['species'] == 'virginica'].cov(),
                 cmap="RdYlBu_r",
                 annot=True,cbar=False,ax=ax3,square=True,
                 vmax = 0.4, vmin = 0)
ax3.set_title('Y = 2, virginica')
```

21.6 线性相关性系数：线性关系强弱

有了上一节的协方差，我们就可以定义线性相关系数 (linear correlation coefficient 或 correlation coefficient)。线性相关系数，也叫皮尔逊相关系数 (Pearson correlation coefficient)，它刻画随机变量线性关系的强度，具体定义为：

$$\rho_{1,2} = \operatorname{corr}(X_1, X_2) = \frac{\operatorname{cov}(X_1, X_2)}{\sigma_1 \sigma_2} \tag{12}$$

ρ 的取值范围在 [−1, 1]；观察 (12)，可以发现 ρ 相当于协方差归一化。也相当于对两个随机变量的 z 分数求协方差：

$$\rho_{1,2} = \operatorname{corr}(X_1, X_2) = \operatorname{cov}\left(\frac{X_1 - \mu_1}{\sigma_1}, \frac{X_2 - \mu_2}{\sigma_2}\right) \tag{13}$$

归一化的线性相关系数比协方差更适合横向比较。

采用和图 10 一样的几何视角，我们来看一下在不同相关性系数条件下，红色和蓝色矩形面积的特征。如图 13 所示，当 ρ = 0.9 时，矩形的颜色几乎都是红色；当 ρ 逐步减小到 0.3 时，红色矩形依然主导，但是蓝色矩形不断变多，也就是红蓝色趋向均衡。

相反，当 ρ = −0.9 时，矩形的颜色中蓝色居多，而且面积和的比例明显压倒优势；当 ρ 逐步增大到−0.3 时，红色矩形增多，面积增大。

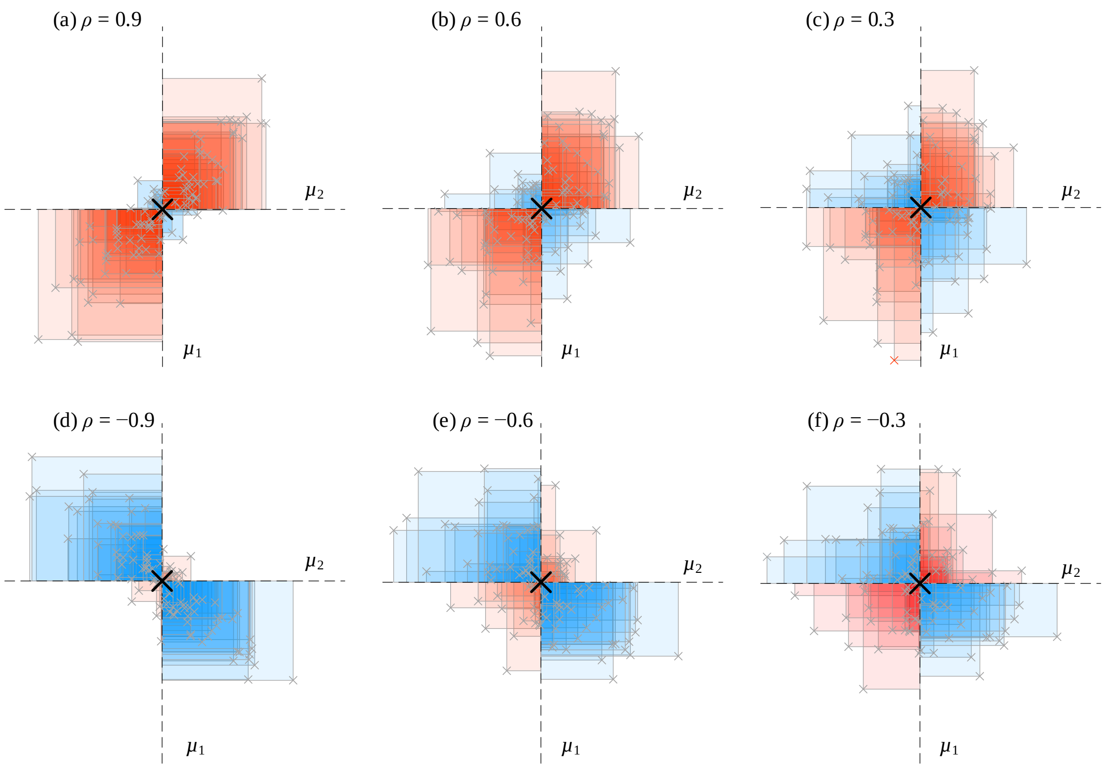


图 13. 几何视角看相关性系数

某个随机变量和自身求线性关系系数，结果为 1，比如下例：

$$\operatorname{corr}(X_1, X_1) = \frac{\operatorname{var}(X_1)}{\sigma_1 \sigma_1} = 1 \tag{14}$$

容易知道，下式成立：

$$\operatorname{corr}(X_i, X_j) = \operatorname{corr}(X_j, X_i) \tag{15}$$

相关性系数矩阵

类似上一节讲过的协方差矩阵，而相关性系数构成的矩阵叫做**相关性系数矩阵** (correlation matrix) $\boldsymbol{P}$；以鸢尾花四个特征为例，其相关性系数矩阵为 4 × 4：

$$\boldsymbol{P} = \begin{bmatrix} 1 & \rho_{1,2} & \rho_{1,3} & \rho_{1,4} \\ \rho_{2,1} & 1 & \rho_{2,3} & \rho_{2,4} \\ \rho_{3,1} & \rho_{3,2} & 1 & \rho_{3,4} \\ \rho_{4,1} & \rho_{4,2} & \rho_{4,3} & 1 \end{bmatrix} \tag{16}$$

线性相关性系数的主对角元素为 1，这是因为随机变量和自身的线性相关系数为 1；非对角线元素为成对相关性系数。

鸢尾花数据的相关性系数矩阵 $\boldsymbol{P}$ 具体为：

$$\boldsymbol{P} = \begin{array}{c} \begin{bmatrix} 1.000 & -0.118 & 0.872 & 0.818 \\ -0.118 & 1.000 & -0.428 & -0.366 \\ 0.872 & -0.428 & 1.000 & 0.963 \\ 0.818 & \underbrace{-0.366} & 0.963 & 1.000 \\ \text{Sepal length, } X_1 & \text{Sepal width, } X_2 & \text{Petal length, } X_3 & \text{Petal width, } X_4 \end{bmatrix} \end{array} \begin{array}{l} \leftarrow \text{Sepal length, } X_1 \\ \leftarrow \text{Sepal width, } X_2 \\ \leftarrow \text{Petal length, } X_3 \\ \leftarrow \text{Petal width, } X_4 \\ \end{array} \tag{17}$$

图 14 所示为 $\boldsymbol{P}$ 的热图。观察相关性系数矩阵 $\boldsymbol{P}$，可以发现花萼长度 X_1 和花萼宽度 X_2 线性负相关，花瓣长度 X_3 和花萼宽度 X_2 线性负相关，花瓣宽度 X_4 和花萼宽度 X_2 线性负相关。

当然，鸢尾花数据集样本数量有限，通过样本数据得出的结论远不足以推而广之。

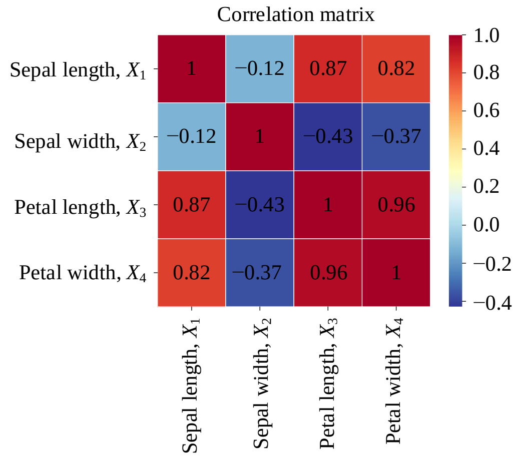


图 14. 鸢尾花数据相关性系数矩阵热图

考虑标签

图 15 为考虑分类标签条件下的协方差矩阵热图。

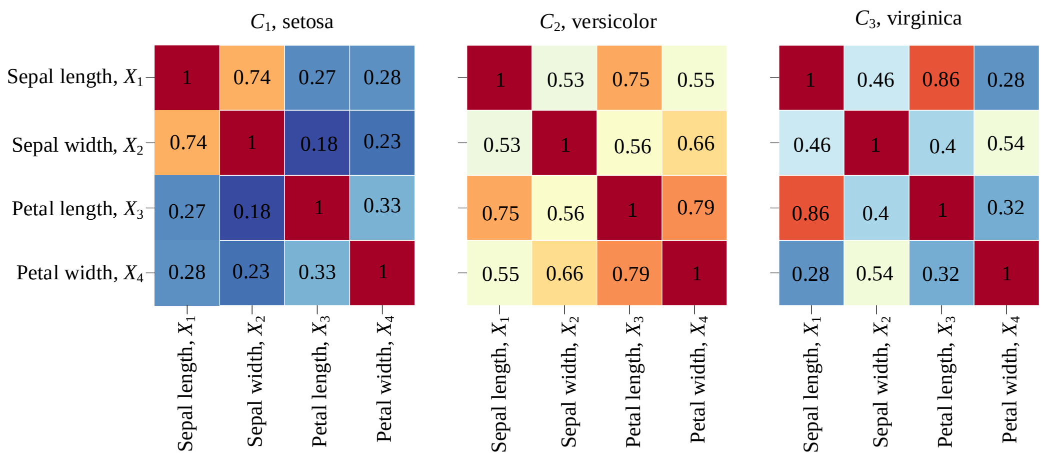


图 15. 相关性系数矩阵热图，考虑分类标签

代码文件 Bk3_Ch21_1.py 中 Bk3_Ch21_1_E 部分绘制本节热图。

```
# Bk3_Ch21_1_E

#%% correlation matrix

RHO = iris_sns.corr()

fig, axs = plt.subplots()
```

```python
h = sns.heatmap(RHO,cmap='RdBu_r', linewidths=.05, annot = True)
h.set_aspect("equal")
h.set_title('$\u03A1$')


#%% compare correlation matrices, with class labels

f,(ax1,ax2,ax3) = plt.subplots(1,3,sharey=True)

g1 = sns.heatmap(iris_sns.loc[iris_sns['species'] == 'setosa'].corr(),
                 cmap="RdYlBu_r",
                 annot=True,cbar=False,ax=ax1,square=True,
                 vmax = 1, vmin = 0.15)
ax1.set_title('Y = 0, setosa')

g2 = sns.heatmap(iris_sns.loc[iris_sns['species'] == 'versicolor'].corr(),
                 cmap="RdYlBu_r",
                 annot=True,cbar=False,ax=ax2,square=True,
                 vmax = 1, vmin = 0.15)
ax2.set_title('Y = 1, versicolor')

g3 = sns.heatmap(iris_sns.loc[iris_sns['species'] == 'virginica'].corr(),
                 cmap="RdYlBu_r",
                 annot=True,cbar=False,ax=ax3,square=True,
                 vmax = 1, vmin = 0.15)
ax3.set_title('Y = 2, virginica')
```

概率统计是数学中很大的一个版块，本书用两章的内容浮光掠影地介绍概率统计的入门知识，目的是让大家了解概率统计中重要概念，并建立它们和其他数学知识的联系。

概率统计，特别是多元概率统计，是数据科学和机器学习很多算法中重要的数学工具。本系列丛书将会在《概率统计》和大家系统探讨。