

[zhuanlan.zhihu.com](https://zhuanlan.zhihu.com)

## Python可视化 | Seaborn5分钟入门(七)——pairplot

8-10 minutes

Seaborn是基于matplotlib的Python可视化库。它提供了一个高级界面来绘制有吸引力的统计图形。

Seaborn其实是在matplotlib的基础上进行了更高级的API封装，从而使得作图更加容易，不需要经过大量的调整就能使你的图变得精致。

今天给大家介绍的是**pairplot**，话不多说，直接进入正题。

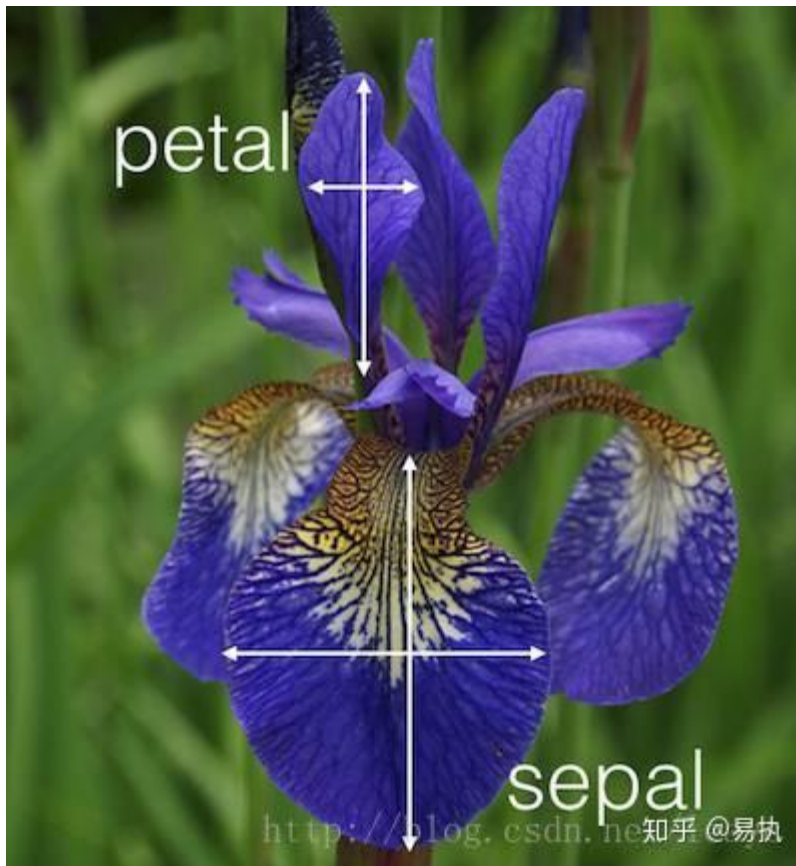
**注：所有代码均在IPython notebook中实现**

### pairplot

pairplot中pair是成对的意思，pairplot主要展现的是**变量两两之间的关系**（线性或非线性，有无较为明显的相关关系），照例来总览一下pairplot的API。

```
seaborn.pairplot(data, hue=None, hue_order=None, palette=None, vars=None, x_vars=None,
y_vars=None, kind='scatter', diag_kind='hist', markers=None, size=2.5, aspect=1, dropna=True,
plot_kws=None, diag_kws=None, grid_kws=None)
```

下面用鸢尾花数据集来介绍pairplot的用法。鸢尾花数据集已经用了不少次了，但大多数人（我也）可能还不知道鸢尾花长啥样，先上张鸢尾花的图。



先导入相关的包，并配置相关的参数

```
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline

sns.set_style('white',{'font.sans-serif':['simhei','Arial']})

#导入seaborn自带iris数据集
data=sns.load_dataset("iris")
#为了方便大家观看，把列名换成中文的
data.rename(columns={"sepal_length":"萼片长",
                    "sepal_width":"萼片宽",
                    "petal_length":"花瓣长",
                    "petal_width":"花瓣宽",
                    "species":"种类"},inplace=True)

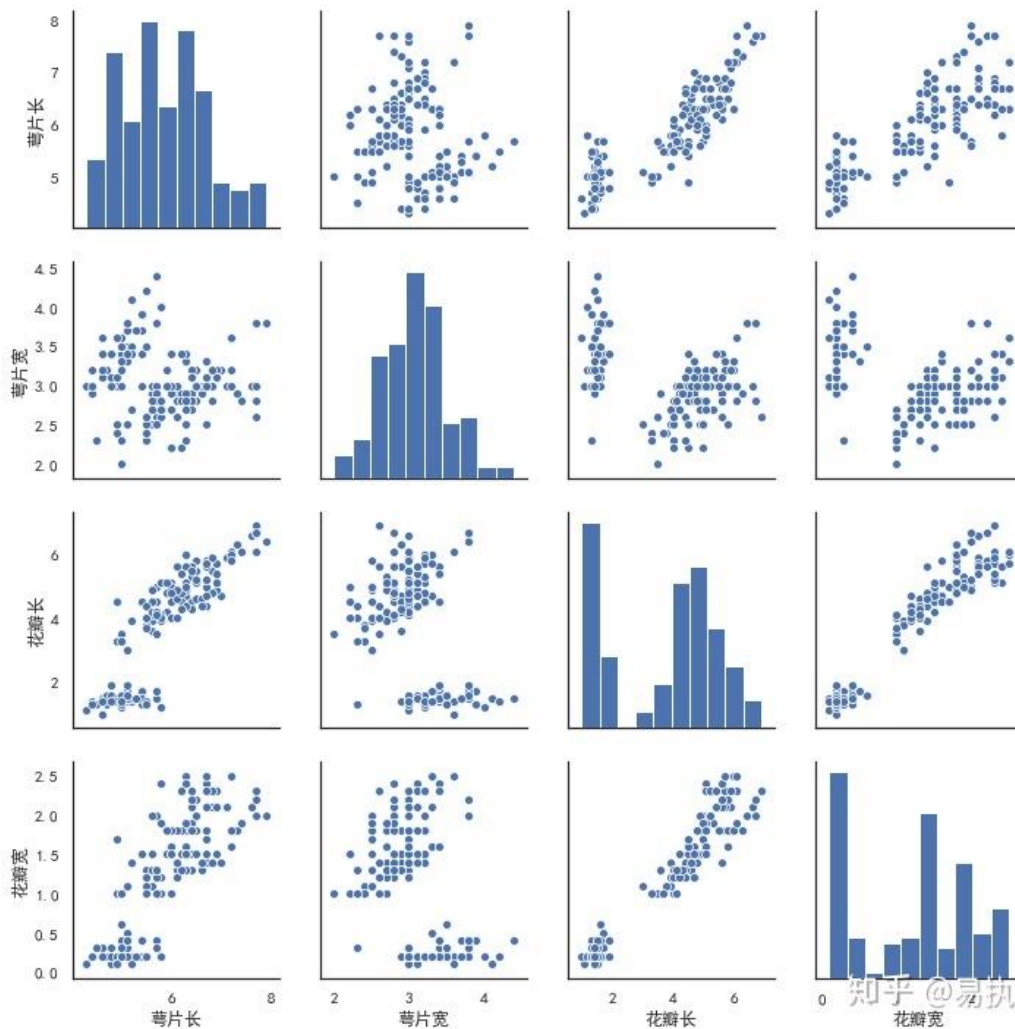
kind_dict = {
    "setosa":"山鸢尾",
    "versicolor":"杂色鸢尾",
    "virginica":"维吉尼亚鸢尾"
}
```

```
data["种类"] = data["种类"].map(kind_dict)
```

```
data.head() #数据集的内容如下
```

	萼片长	萼片宽	花瓣长	花瓣宽	种类
0	5.1	3.5	1.4	0.2	山鸢尾
1	4.9	3.0	1.4	0.2	山鸢尾
2	4.7	3.2	1.3	0.2	山鸢尾
3	4.6	3.1	1.5	0.2	山鸢尾
4	5.0	3.6	1.4	0.2	山鸢尾

先来直观看看pairplot长什么样



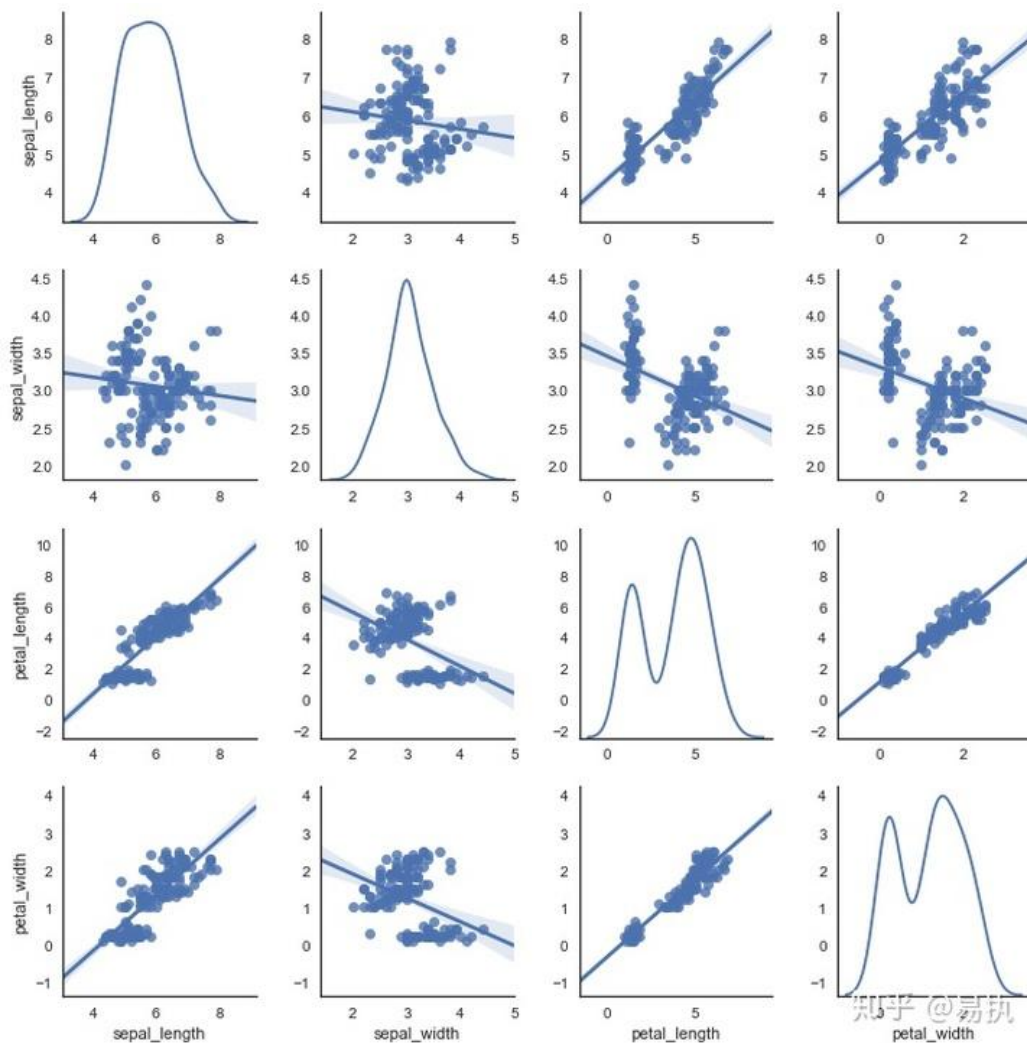
可以看到**对角线上**是各个属性的直方图（分布图），而非**对角线上**是两个不同属性之间的相关图，从图中我们发现，花瓣的长度和宽度之间以及萼片的长短和花瓣的长、宽之间具有比较明显的相关关系。

接下来介绍一下pairplot主要的参数及其用法

**kind**: 用于控制非对角线上的图的类型，可选"scatter"与"reg"

diag\_kind: 控制对角线上的图的类型, 可选"hist"与"kde"

```
sns.pairplot(data, kind="reg", diag_kind="kde")
```



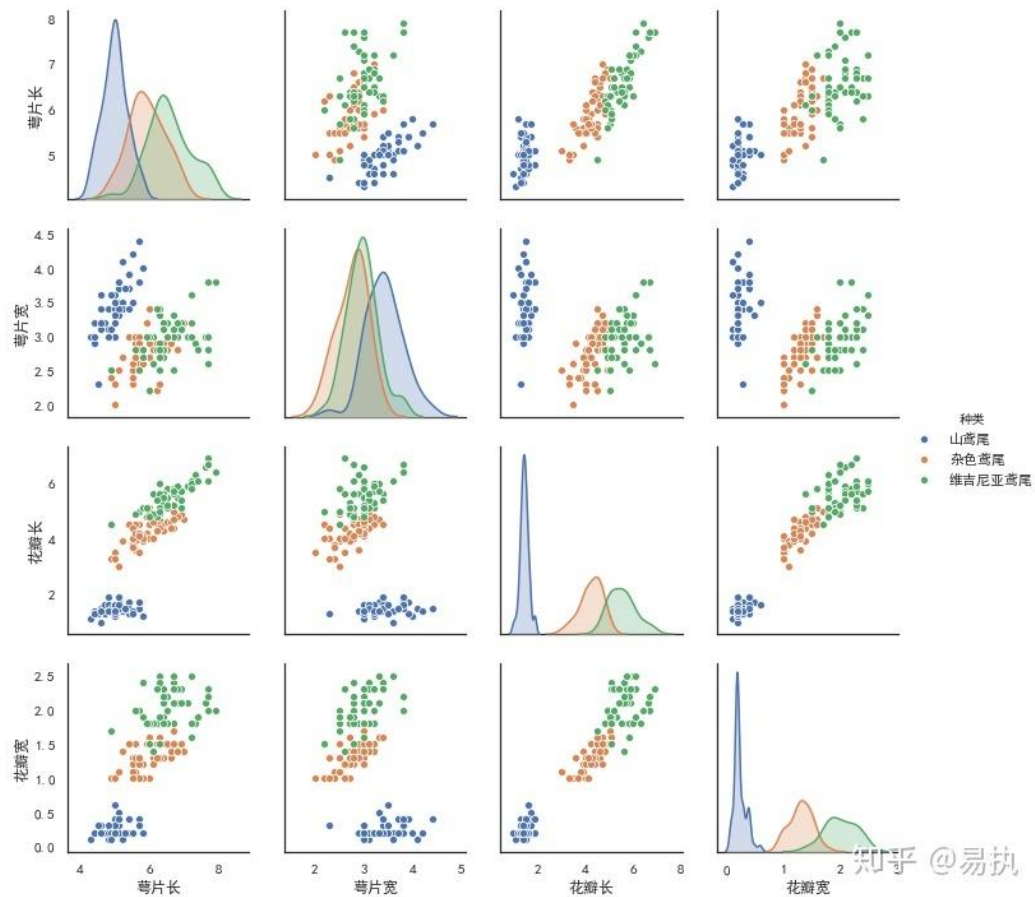
将 kind 参数设置为 "reg" 会为非对角线上的散点图拟合出一条回归直线, 更直观地显示变量之间的关系。

那对于不同种类的花, 其花萼和花瓣有什么比较鲜明的特征吗? 我们通过 hue 参数把不同种类的花区分开, 进行进一步分析。

hue: 针对某一字段进行分类

#不同类别的点会以不同的颜色显现出来

```
sns.pairplot(data, hue="种类")
```



我们可以从经过hue分类后的pairplot中发现，不论是从对角线上的分布图还是从分类后的散点图，都可以看出对于不同种类的花，其**萼片长**、**花瓣长**、**花瓣宽**的分布差异较大，换句话说，这些属性是可以帮助我们去**识别不同种类的花**的。

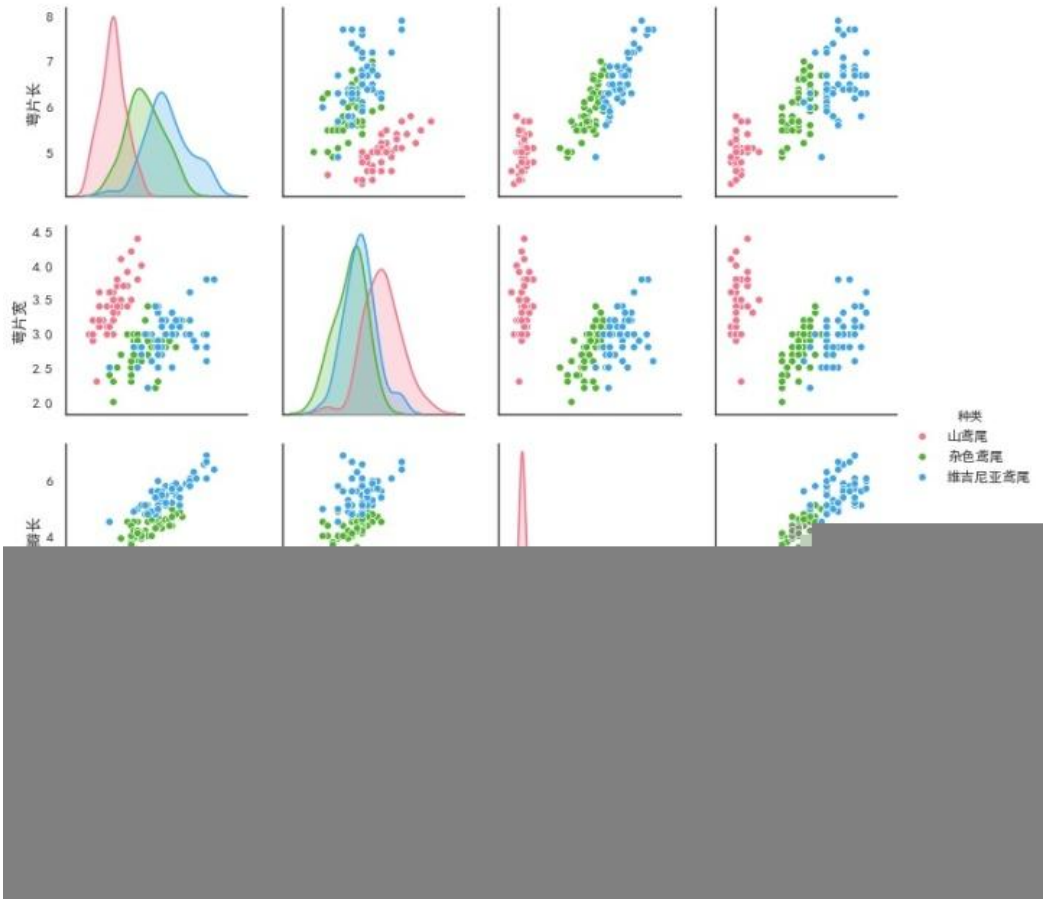
比如，对于萼片、花瓣长度较短，花瓣宽度较窄的花，那么它大概率是山鸢尾。

当然，可以通过 palette 参数来调出自己想要的颜色

palette: 控制色调

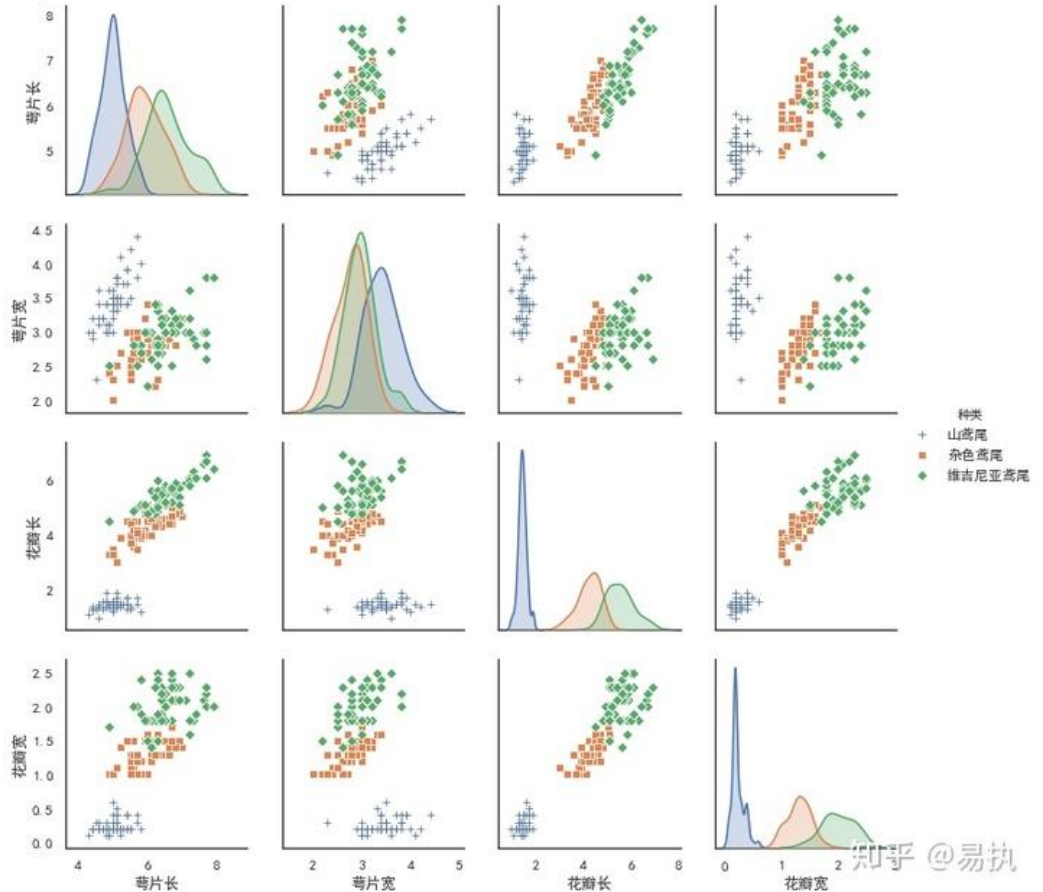
```
sns.pairplot(data,hue="种类",palette="husl")
```





markers：控制散点的样式

```
sns.pairplot(data,hue="种类",markers=["+", "s", "D"])
```

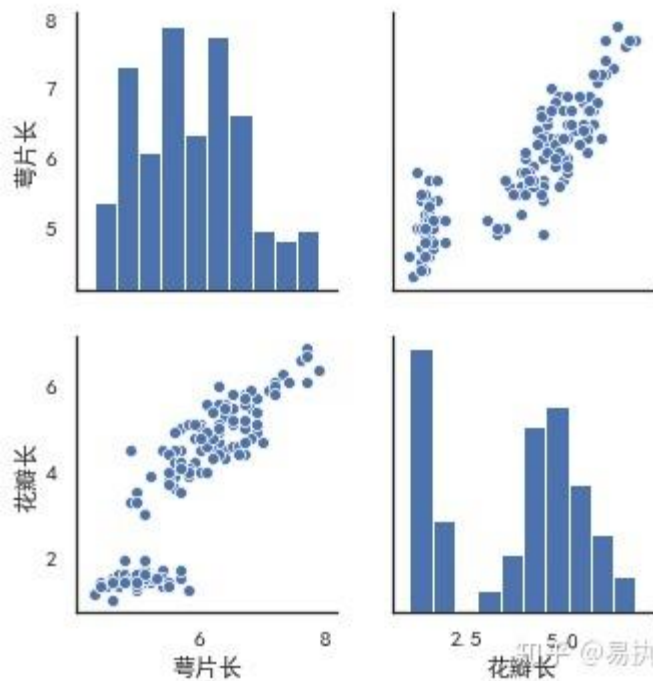


当我们想单独研究某两个（或多个）变量的关系时，我们只需要通过vars参数指定你想研究的变量

vars,x\_vars,y\_vars: 选择数据中的特定字段，以list形式传入

#单独用vars参数选择"萼片长"和"花瓣长"两种属性

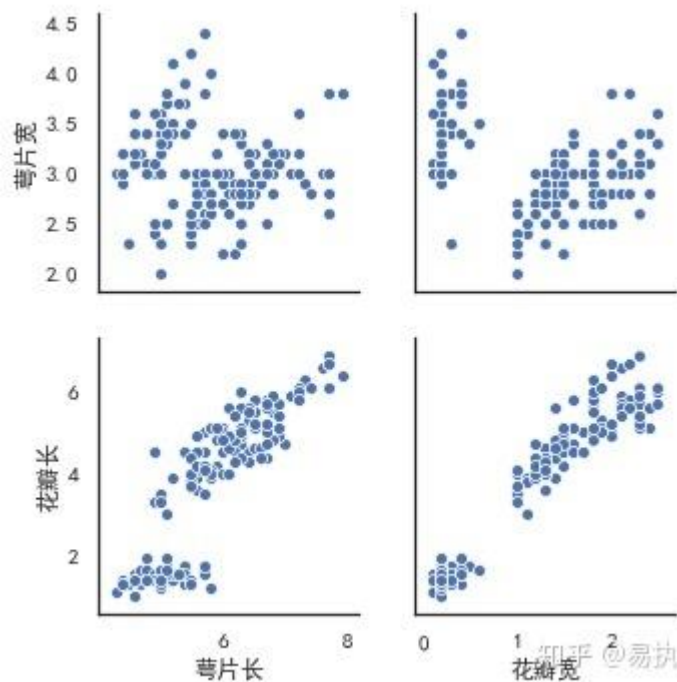
```
sns.pairplot(data,vars=["萼片长","花瓣长"])
```



# 用x\_vars和 y\_vars参数指定

# 需要注意的是，x\_vars和y\_vars要同时指定

```
sns.pairplot(data,x_vars=["萼片长","花瓣宽"],
              y_vars=["萼片宽","花瓣长"])
```

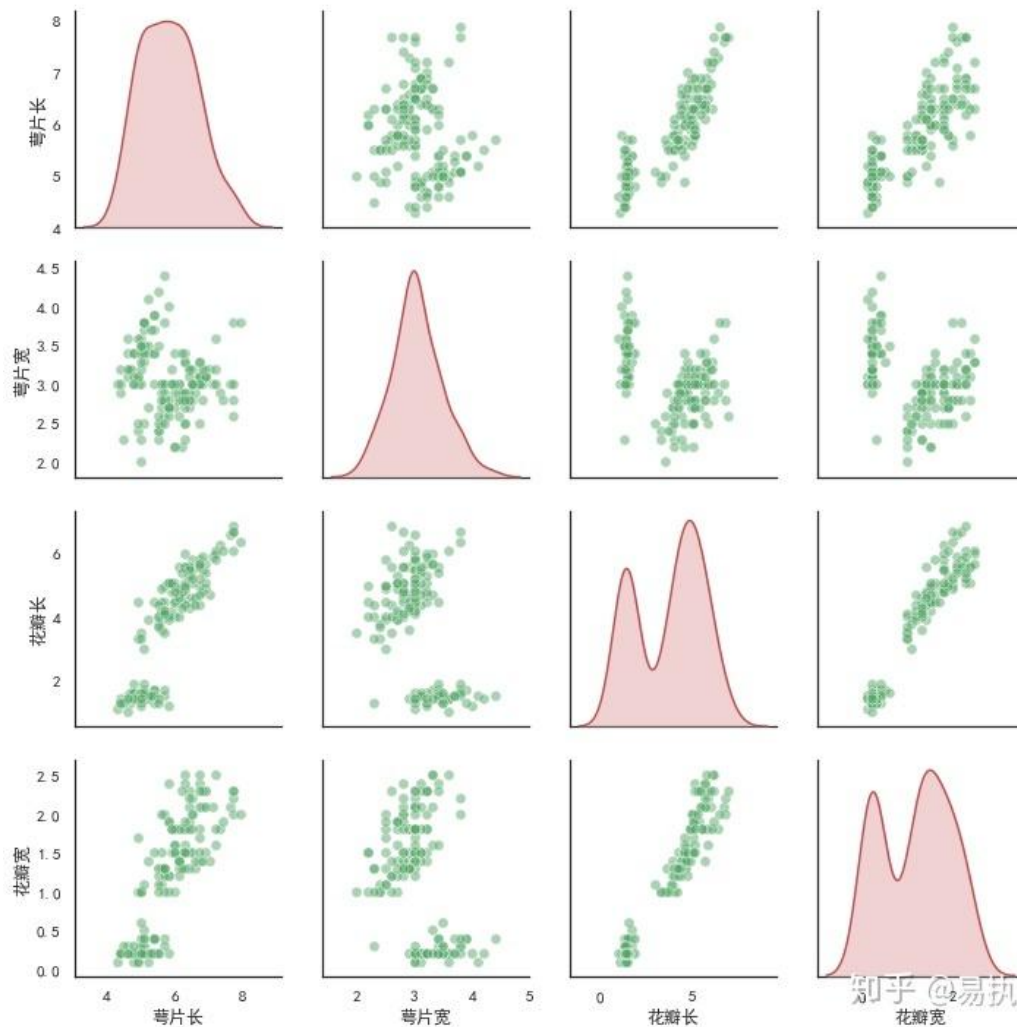


更多的样式可以通过plot\_kws和diag\_kws来进行控制

plot\_kws: 用于控制非对角线上的图的样式

diag\_kws: 用于控制对角线上图的样式

```
sns.pairplot(data,diag_kind="kde",  
              plot_kws=dict(s=50,edgecolor="w",color="g",alpha=.5),  
              diag_kws=dict(shade=True,color="r"))
```



原创不易，如果觉得有点用，希望可以点个赞，拜谢各位老铁！