

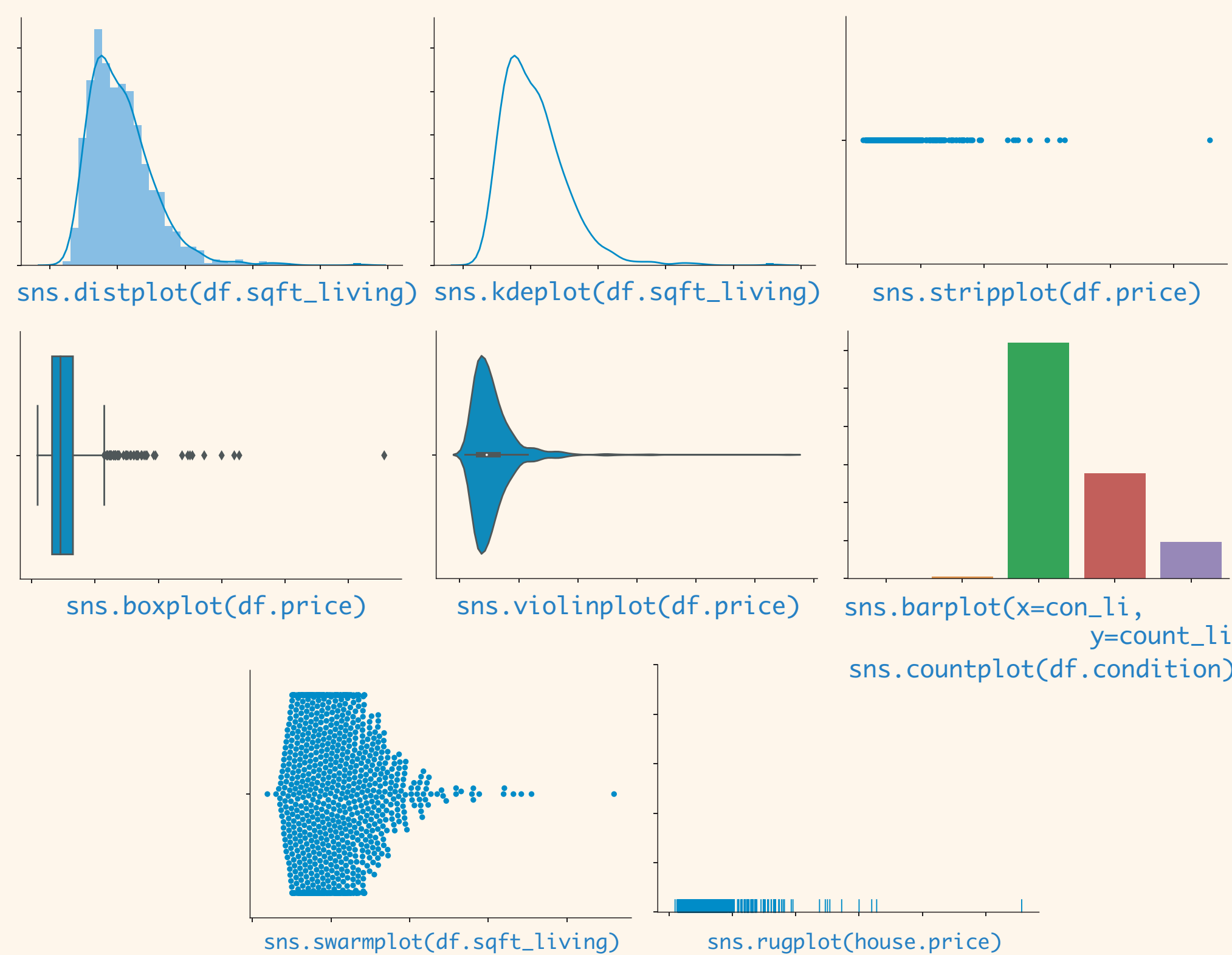
Seaborn

Seaborn 包基于 Matplotlib 设计，很多 Matplotlib 中的函数在 Seaborn 中同样适用，例如 `plt.title()` 等。Seaborn 还提供了对 Pandas、NumPy 等数据结构的完美支持并可以绘制出独具特色并含有统计信息的高品质图形。其核心功能包括优美的绘图模板、高级统计绘图能力以及绘制时间序列数据等。

导入包 (seaborn 0.8.1)
`import seaborn as sns`

可视化单一特征

效果展示



函数说明

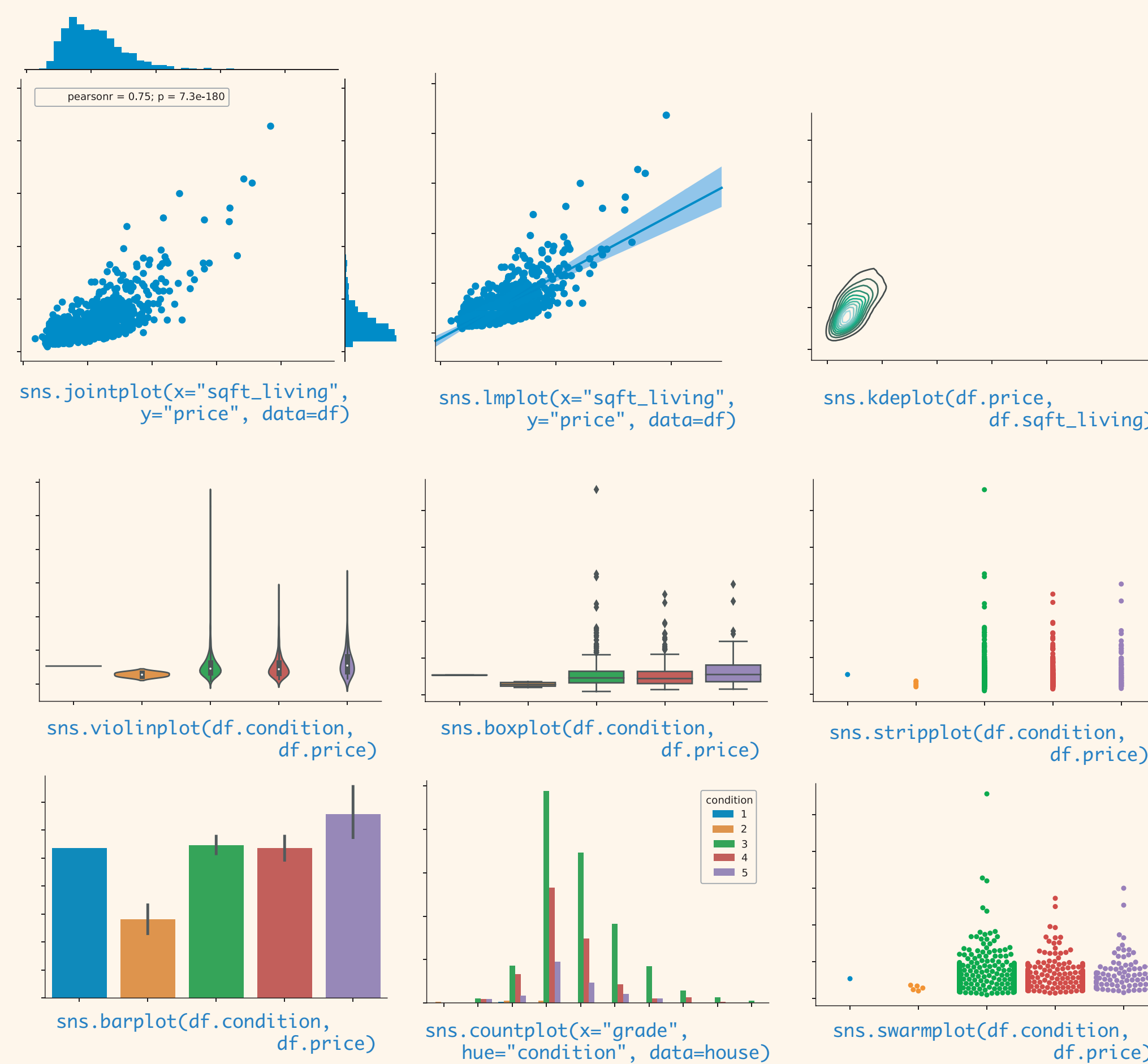
`sns.distplot()` | 连续型数据的直方分布。
`sns.kdeplot()` | 连续型数据的核密度分布。
`sns.stripplot()` | 绘制一维散点图，可通过其中的 `jitter=True` 参数对数据添加噪声。
`sns.boxplot()` | 绘制箱线图。
`sns.violinplot()` | 绘制小提琴图。
`sns.barplot()` | 绘制条形图，需要输入每个

矩形条的位置以及高度。

`sns.countplot()` | 绘制计数图，其可自动统计矩形条的高度，是 `barplot` 的简便版本。
`sns.swarmplot()` | 绘制集群图，即所有的数据点都会尽量围绕中线分布，避免重叠。
`sns.rugplot()` | 绘制密度条图，用来查看数据的一维分布，其效果类似于 `stripplot`。

可视化两个特征

效果展示



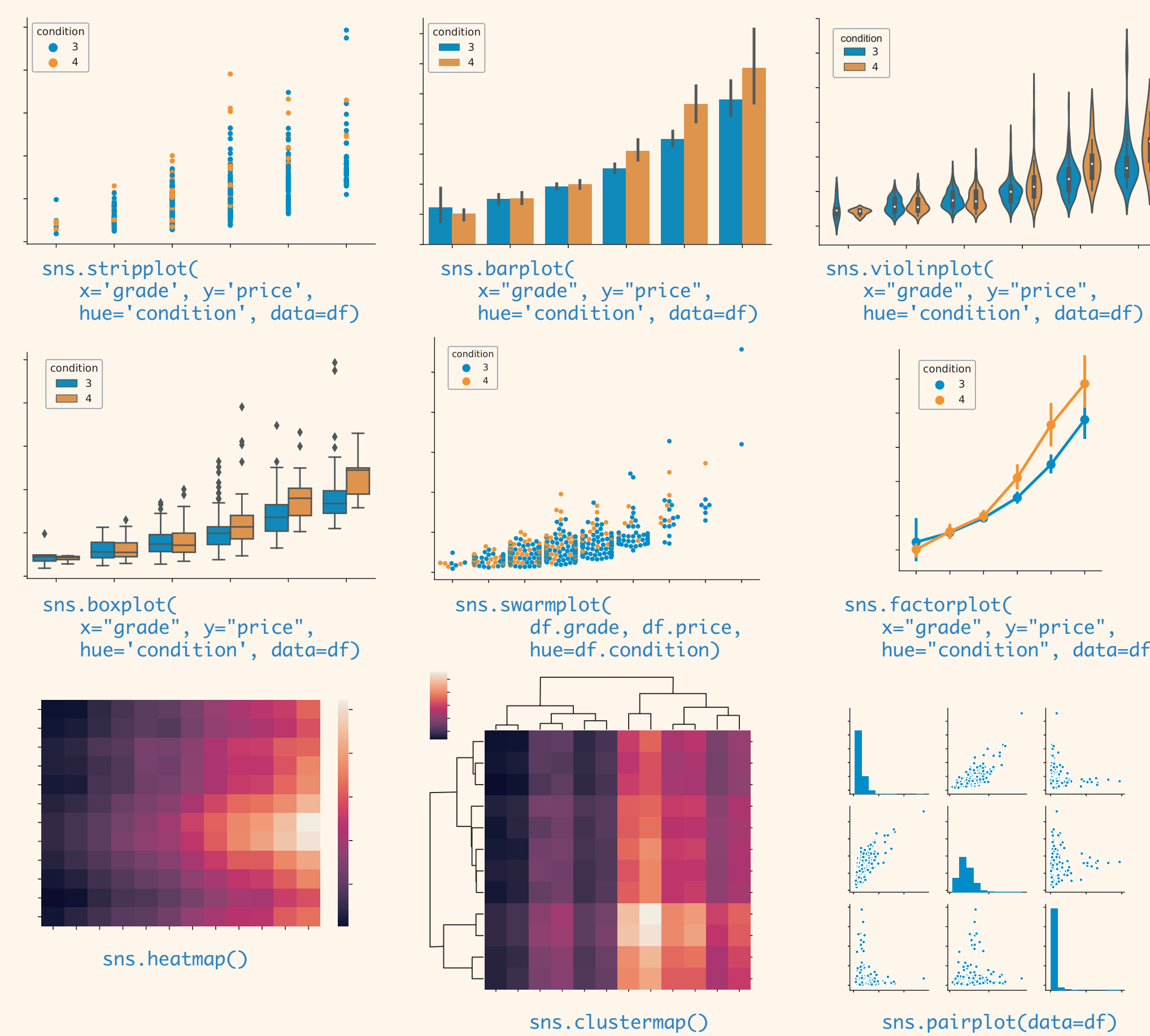
函数说明

`sns.jointplot()` | 绘制散点图（两个连续型特征），默认会添加两个特征的直方图。
`sns.lmplot()` | 绘制带有回归线的散点图（两个连续型特征）。
`sns.kdeplot()` | 绘制二维核密度图（两个连续型特征）。
`sns.boxplot()` | 绘制箱线图（一个为离散一个连续型特征）。
`sns.violinplot()` | 绘制小提琴图（一个离散一个连续型特征）。
`sns.barplot()` | 绘制条形图（一个离散一个连续型特征）并自定义添加每个矩形的误差条。

`sns.countplot()` | 绘制计数图（两个离散型），其可自动统计每个矩形条的高度。
`sns.stripplot()` | 绘制二维散点图（一个离散一个连续型），可通过其中的 `jitter=True` 参数对数据添加噪声。
`sns.swarmplot()` | 绘制集群图（一个离散一个连续型），即所有的数据点都会尽量围绕中线分布，避免重叠。

可视化多个特征

效果展示



函数说明

当添加第三个特征时可以对之前的函数添加新参数 `hue`，也可以使用新函数。如果使用参数 `hue`，则 `hue` 所对应的特征类别不宜过多。
`sns.stripplot` | 二维三特征散点图。
`sns.barplot()` | 二维三特征条形图。
`sns.violinplot()` | 二维三特征小提琴图。
`sns.boxplot()` | 二维三特征箱线图。
`sns.swarmplot()` | 二维三特征集群图。
`sns.factorplot()` | 二维三特征趋势图，默认添加误差条。可对其添加 `col` 参数绘制第四特征。
`sns.heatmap()` | 热力图，需处理数据集格式。

`sns.clustermap()` | 聚类热力图，需处理数据集格式。
`sns.pairplot()` | 绘制特征散点图矩阵，可用于快速检测数据集所有特征的分布情况。

绘制子图

绘制子图与 Matplotlib 方式一致，例如：
`fig, axes=plt.subplots(1,2)`
`sns.distplot(df.living, ax=axes[0])`
`sns.distplot(df.lot, ax=axes[1])`

调色盘 Palette

with `sns.color_palette()`:
`sns.barplot()`
用于临时修改绘图色彩，它的三个参数为：
`palette` | 色彩模板名称，可以是 `deep`、`muted`、`pastel`、`bright`、`dark` 和 `color-blind` 以及任何 Matplotlib 调色盘名称。
`n_colors` | 指定颜色的个数，默认为 6。
`desat` | 调整颜色饱和度（0 为完全灰度）。
该函数结合 `sns.palplot()` 函数使用可查看所使用调色盘的具体颜色，用法为：
`sns.palplot(sns.color_palette())`
`sns.set_palette()`:
用于修改全局调色盘样式，其参数用法与 `sns.color_palette()` 一致。

`sns.reset_defaults()`:
还原所有 RC 参数，即重置调色盘。

保存图形

将绘图函数赋值给变量 `a` 并查看该变量的类型，如果为 `AxesSubplot` 类型，则使用 `a.get_figure().savefig('fig.eps')`；如果为 `PairGrid` 类型，则使用 `a.savefig('fig.eps')`。

