

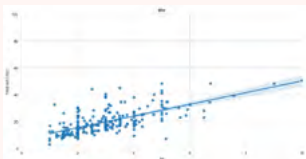


Seaborn 是一种基于 matplotlib 的 python 图形可视化工具库。

它是在 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, 能做出很具有吸引力的图, 更简单地对数据分析过程做出更美观的可视化图表结果呈现。

Seaborn 绘图基本步骤与示例

```
tips = sns.load_dataset("tips") # Step 1 准备数据
sns.set_style("whitegrid") # Step 2 设定画布外观
g = sns.lmplot(x="tip", y="total_bill", \
               data=tips, aspect=2) # Step 3 使用 Seaborn 绘图
g = (g.set_axis_labels("Tip", "Total bill(USD)")\
     .set(xlim=(0, 10), ylim=(0, 100)))
plt.title("title") # Step 4 自定义图形
plt.show(g) # Step 5 展示图形
```



1. 数据准备

可以是 numpy 数组和 Dataframe 等数据格式

```
import pandas as pd
import numpy as np
uniform_data = np.random.rand(10, 12)
```

```
data = pd.DataFrame({'x': np.arange(1, 101), \
                    'y': np.random.normal(0, 4, 100)})
```

Seaborn 提供了内置数据集:

```
titanic = sns.load_dataset("titanic")
iris = sns.load_dataset("iris")
```

2. 画布外观

2.1 Seaborn 样式

```
f, ax = plt.subplots(figsize=(5, 6)) # 创建画布与子图
sns.set() # 设置或重置 Seaborn 默认值
sns.set_style("whitegrid") # 设置 matplotlib 参数
sns.set_style("ticks", {"xtick.major.size": 8, "ytick.major.size": 8})
sns.axes_style("whitegrid") # 返回参数字典或用 with 设置临时样式
```

2.2 上下文函数

```
sns.set_context("talk") # 将上下文设置为 "talk"
# 上下文设为 "notebook", 缩放字体, 覆盖参数映射
sns.set_context("notebook", font_scale=1.5, rc={"lines.linewidth": 2.5})
```

2.3 调色板

```
sns.set_palette("husl", 3) # 定义调色版
sns.color_palette("husl") # 使用 with 临时设置调色板
flatui = ["#9b59b6", "#3498db", "#95a5a6", "#e74c3c", "#34495e", "#2ecc71"]
sns.set_palette(flatui) # 设置调色板
```

Show Me AI

4. 深度自定义

4.1 Axisgrid 对象

```
g.despine(left=True) # 移除左框

g.set_ylabels("Survived") # 设置 Y 轴标签
g.set_xticklabels(rotation=45) # 设置 X 轴刻度标签

g.set_axis_labels("Survived", "Sex") # 设置坐标轴标签

# 设置 X 与 Y 轴的幅度区间和刻度
h.set(xlim=(0, 5), ylim=(0, 5), xticks=[0, 2.5, 5], yticks=[0, 2.5, 5])
```

4.2 图形

```
plt.title("A Title") # 添加图形标题
```

```
plt.ylabel("Survived") # 调整 Y 轴标签
plt.xlabel("Sex") # 调整 X 轴标签
```

```
plt.ylim(0, 100) # 调整 Y 轴幅度区间
plt.xlim(0, 10) # 调整 X 轴幅度区间
```

```
plt.setp(ax, yticks=[0, 5]) # 调整图形属性
plt.tight_layout() # 调整子图参数
```

5. 显示或保存图形

show 与 savefig 函数

```
plt.show() # 显示图形
# 将画布保存为图形
plt.savefig("foo.png")
# 保存透明画布
plt.savefig("foo.png", transparent=True)
```

6. 关闭与清除

绘图关闭与清除操作

```
plt.cla() # 清除坐标轴
plt.clf() # 清除画布
plt.close() # 关闭窗口
```



Seaborn 速查表

获取最新版 | <http://www.showmeai.tech/>

作者 | 韩信子 @ShowMeAI

设计 | 南乔 @ShowMeAI

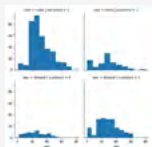
参考 | DataCamp Cheatsheet

扫码回复“数据科学”

下载最新全套速查表

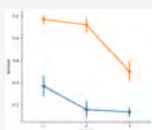
3. 使用 Seaborn 绘图

3.1 坐标轴栅格



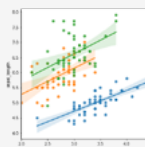
绘制条件关系的子图栅格

```
g = sns.FacetGrid(titanic, col="survived", row="sex")
g = g.map(plt.hist, "age")
```



在分面栅格上绘制分类图

```
sns.factorplot(x="pclass", y="survived", hue="sex", data=titanic)
```



绘制适配分面栅格的数据与回归模型

```
sns.lmplot(x="sepal_width", y="sepal_length", hue="species", data=iris)
```

```
h = sns.PairGrid(iris) # 绘制配对关系的子图栅格
```

```
h = h.map(plt.scatter) # 绘制配对的双变量分布
```

```
sns.pairplot(iris) # 绘制双变量图的边际单变量图栅格
```

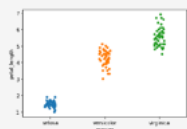
```
i = sns.JointGrid(x="x", y="y", data=data)
```

```
i = i.plot(sns.regplot, sns.distplot)
```

绘制双变量分布

```
sns.jointplot("sepal_length", "sepal_width", data=iris, kind='kde')
```

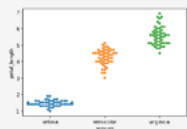
Show Me AI



抖动图(散点图)

含分类变量的抖动图

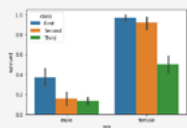
```
sns.stripplot(x="species", y="petal_length", data=iris)
```



蜂群图(散点图)

不重叠分类蜂群图

```
sns.swarmplot(x="species", y="petal_length", data=iris)
```



条形图

用散点图显示点估计值和置信区间

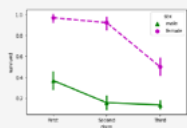
```
sns.barplot(x="sex", y="survived", hue="class", data=titanic)
```



计数图

显示观测数量

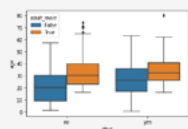
```
sns.countplot(x="deck", data=titanic, palette="Greens_d")
```



点图

显示点估计和置信区间

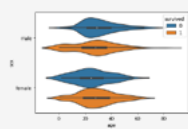
```
sns.pointplot(x="class", y="survived", hue="sex", data=titanic,
palette={"male": "g", "female": "m"},
markers=["^", "o"], linestyle=["-", "--"])
```



箱形图 / 宽表数据箱型图

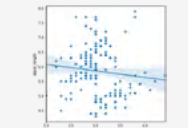
```
sns.boxplot(x="alive", y="age", hue="adult_male", data=titanic)
```

```
sns.boxplot(data=iris, orient="h") # 使用宽表数据的箱形图
```



小提琴图

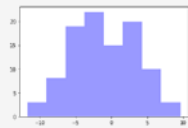
```
sns.violinplot(x="age", y="sex", hue="survived", data=titanic)
```



回归拟合图

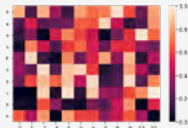
绘制与线性回归模型拟合的数据

```
sns.regplot(x="sepal_width", y="sepal_length", data=iris, ax=ax)
```



变量分布图

```
plot = sns.distplot(data.y, kde=False, color="b") # 绘制单变量分布
```



热力图

```
sns.heatmap(uniform_data, vmin=0, vmax=1) # 热力图
```

3.2 常用绘图类型

数据科学工具库速查表



NumPy 是 Python 数据科学计算的核心库，提供了高性能多维数组对象及处理数组的工具。使用以下语句导入 NumPy 库：

```
import numpy as np
```



SciPy 是基于 NumPy 创建的 Python 科学计算核心库，提供了众多数学算法与函数。



Pandas 是基于 NumPy 创建的 Python 库，为 Python 提供了易于使用的数据结构和数据分析工具。使用以下语句导入：

```
import pandas as pd
```



Matplotlib 是 Python 的二维绘图库，用于生成符合出版质量或跨平台交互环境的各类图形。

```
import matplotlib.pyplot as plt
```



Seaborn 是基于 matplotlib 开发的高阶 Python 数据可视图库，用于绘制优雅、美观的统计图形。使用下列别名导入该库：

```
import matplotlib.pyplot as plt
```

```
import seaborn as sns
```



Bokeh 是 Python 的交互式可视图库，用于生成在浏览器里显示的大规模数据集高性能可视图。Bokeh 的中间层通用 **bokeh.plotting** 界面主要为两个组件：数据与图示例。

```
from bokeh.plotting import figure
```

```
from bokeh.io import output_file, show
```

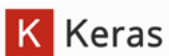


PySpark 是 Spark 的 Python API，允许 Python 调用 Spark 编程模型。Spark SQL 是 Apache Spark 处理结构化数据模块。

AI 垂直领域工具库速查表



Scikit-learn 是开源的 Python 库，通过统一的界面实现机器学习、预处理、交叉验证及可视化算法。



Keras 是强大、易用的深度学习库，基于 Theano 和 TensorFlow 提供了高阶神经网络 API，用于开发和评估深度学习模型。



“TensorFlow™ is an open source software library for numerical computation using data flow graphs.” **TensorFlow** 是 Google 公司开发的机器学习架构，兼顾灵活性和扩展性，既适合用于工业生产也适合用于科学研究。



PyTorch 是 Facebook 团队 2017 年初发布的深度学习框架，有利于研究人员、爱好者、小规模项目等快速搞出原型。**PyTorch** 也是 Python 程序员最容易上手的深度学习框架。



Hugging Face 以开源的 NLP 预训练模型库 **Transformers** 而广为人知，目前 GitHub Star 已超过 54000+。**Transformers** 提供 100+ 种语言的 32 种预训练语言模型，简单，强大，高性能，是新手入门的不二选择。



OpenCV 是一个跨平台计算机视觉库，由 C 函数 /C++ 类构成，提供了 Python、MATLAB 等语言的接口。**OpenCV** 实现了图像处理和计算机视觉领域的很多通用算法。

编程语言速查表



SQL 是管理关系数据库的结构化查询语言，包括数据的增删查改等。作为数据分析的必备技能、岗位 JD 的重要关键词，SQL 是技术及相关岗位同学一定要掌握的语言。



Python 编程语言简洁快速、入门简单且功能强大，拥有丰富的第三方库，已经成为大数据和人工智能领域的主流编程语言。

More...

AI 知识技能速查表



Jupyter Notebook 交互式计算环境，支持运行 40+ 种编程语言，可以用来编写漂亮的交互式文档。这个教程把常用的基础功能讲解得很清楚，对新手非常友好。



正则表达式 非常强大，能匹配很多规则的文本，常用于文本提取和爬虫处理。这也是一门令人难以捉摸的语言，字母、数字和符号堆在一起，像极了“火星文”。

More...



ShowMeAI 速查表 (©2021)

获取最新版 | <http://www.showmeai.tech/>

作者 | 韩信子 @ShowMeAI

设计 | 南乔 @ShowMeAI

数据科学工具库速查表

扫码回复“数据科学”

获取最新全套速查表

AI 垂直领域工具库速查表

扫码回复“工具库”

获取最新全套速查表

编程语言速查表

扫码回复“编程语言”

获取最新全套速查表

AI 知识技能速查表

扫码回复“知识技能”

获取最新全套速查表