

皮大大

489 篇文章

plotly-express-22-plotly使用技巧大全

学习

实践

活动

专区

工具

VIP

文章/答案/技巧/干货

与文单

提问

专栏首页

机器学习/数据可视化

plotly-express-22-plotly使用技巧大全

plotly-express-22-plotly使用技巧大全

发布于2021-03-01 16:04:26 阅读 1.1K

1

0

本文将前段时间写的 **plotly-express** 可视化库的相关技巧进行整理，方便后续快速实现调用

先整理之前写的亮点

后面肯定会补充内容

常见绘图参数

常见图形的[绘图参数](#)

修改x/y轴名称(px)

```
In [6]: x = np.linspace(0,10,100)
y = np.sin(x)
px.line(x=x,y=y labels={"x":"t","y":"sin(t)"})
```

修改x/y轴名称(go)

修改X/Y轴名称 (figure实现)

坐标轴刻度

作者介绍

多子图绘制-1

多子图绘制-2

冒泡点大小

冒泡点颜色

添加注释 (某个特点)

柱状图-信息显示在外面

柱状图-改变柱子颜色

绘制双坐标

柱状图-坐标轴排序

饼图-颜色 (自定义)

go实现

px实现

精选专题

饼图-文本隐藏

饼图-布局和属性设置

腾讯云原生专题

云原生技术干货地。

活动推荐

社区创作者年终回顾

参与活动，赢取限量周边礼品

邀请好友加入自媒体分...

邀请好友，同享奖励 30 / 100 / 180 元云服务器...

运营活动

腾讯云

移动推送

爆品特惠专场

限时4.5折起

立即抢购

关注

腾讯云[开发者](#)公众号

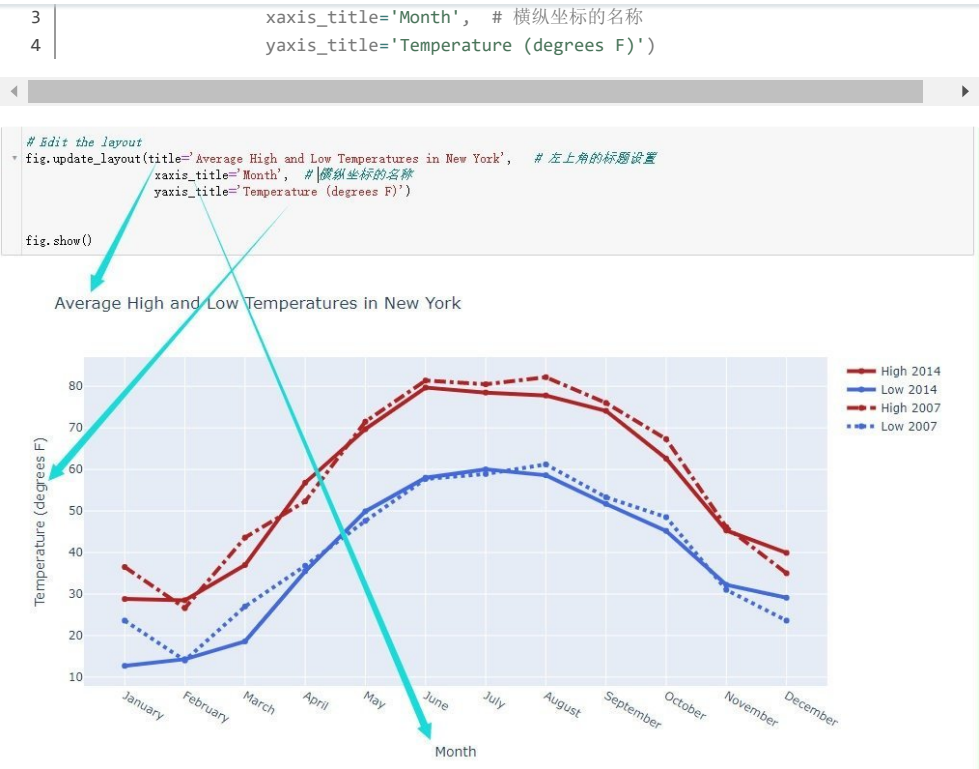
将获得

10元无门槛金券

洞察腾讯核心技术剖析业!

皮大大
489 篇文章

plotly-express-22-plotly使用技巧大全



修改X/Y轴名称 (figure实现)

```
1 # 布局设置
2 layout = go.Layout(
3     title = 'Prime genre', # 整个图的标题
4     margin = dict(
5         l = 100 # 左边距离
6     ),
7     xaxis = dict(
8         title = 'Type of app' # 2个轴的标题
9     ),
10    yaxis = dict(
11        title = 'Count of app'
12    ),
13    width = 900,
14    height = 500
15 )
16
17 fig = go.Figure(data=data, layout=layout)
18 fig.update_traces(textposition="outside")
```

坐标轴刻度

坐标轴的起始点和间距问题

多子图绘制-1

```
1 fig = go.Figure()
2
3 # add traces
4 fig.add_trace(go.Scatter(x=random_x,y=random_y0,
5                         mode="markers",name="markers"))
6 fig.add_trace(go.Scatter(x=random_x, y=random_y1,
7                         mode='lines+markers',name='lines+markers'))
8 fig.add_trace(go.Scatter(x=random_x, y=random_y2,
9                         mode='lines',name='lines'))
10 fig.add_trace(go.Scatter(x=random_x, y=random_y3,
11                         mode='markers',name='markers'))
12 fig.show()
```

- 修改x/y轴名称(go)
- 修改X/Y轴名称 (figure实现)
- 坐标轴刻度
- 多子图绘制-1
- 多子图绘制-2
- 冒泡点大小
- 冒泡点颜色
- 添加注释 (某个特殊点)
- 柱状图-信息显示在外面
- 柱状图-改变柱子颜色
- 绘制双坐标
- 柱状图-坐标轴排序
- 饼图-颜色 (自定义)
- go实现
- px实现
- 饼图-文本隐藏
- 饼图-布局和属性设置



皮大大

489 篇文章

plotly-express-22-plotly使用技巧大全

1

0



多子图绘制-2

子图的绘制知识点很多，主要包含：

- 每个子图名称
- 指定几行几列
- 子图属性设置
 - 第一个子图的起始位置
 - 每个子图的标题
 - 子图之间的间隔设置
 - 如何共享x轴
 - 每个子图中的文本信息设置及位置显示
 - 子图右边的图例名称
 - 子图的位置通过row/col实现
 - 单独设置xy轴的名称
- 共享轴
- 自定义子图位置（在哪行哪列）
- 子图类型

```
1 fig = make_subplots(
2     rows=2, cols=2, shared_yaxes=True, # 共享y轴
3     specs=[{"type": "xy"}, {"type": "polar"}], # 子图类型
4     [{"type": "domain"}, {"type": "scene"}]),
5 )
```

冒泡点大小

```
1 fig = go.Figure(go.Scatter(
2     x=np.linspace(0,50,10),
3     y=np.random.randint(0,50,10),
4     mode="markers",
5     marker=dict(size=np.random.randint(0,50,10), # 通过字典的形式来实现
6               color=np.random.randint(50,100,10))
7 ))
```

冒泡点颜色

```
1 fig = go.Figure(data=go.Scatter(
2     x=x,
3     y=y,
4     mode="markers",
5     marker=dict( # marker是字典的形式
6         size=20,
7         color=np.random.randint(0,100,500), # 指定颜色区间
8         colorscale="Viridis", # 选择哪种颜色
9         showscale=True # 右边的颜色尺度尺是否显示
10    )
11 ))
```

添加注释（某个特殊点）

修改x/y轴名称(go)

修改X/Y轴名称（figure实现）

坐标轴刻度

多子图绘制-1

多子图绘制-2

冒泡点大小

冒泡点颜色

添加注释（某个特殊点）

柱状图-信息显示在外面

柱状图-改变柱子颜色

绘制双坐标

柱状图-坐标轴排序

饼图-颜色（自定义）

- go实现
- px实现

饼图-文本隐藏

饼图-布局和属性设置



https://cloud.tencent.com/developer/article/1794886

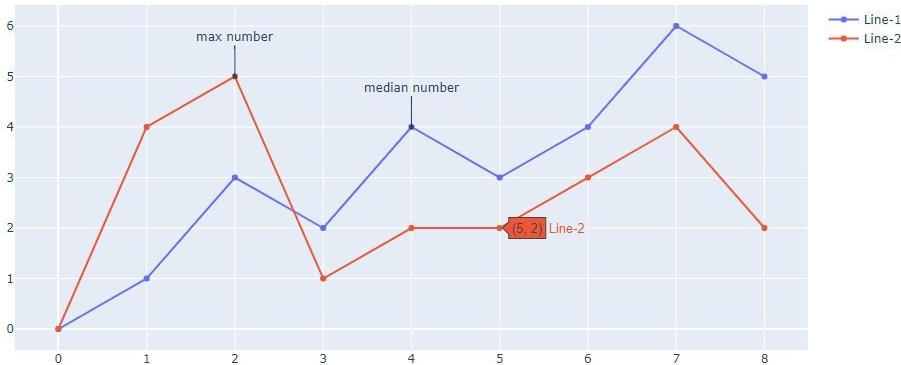
3/13

皮大大

489 篇文章

plotly-express-22-plotly使用技巧大全

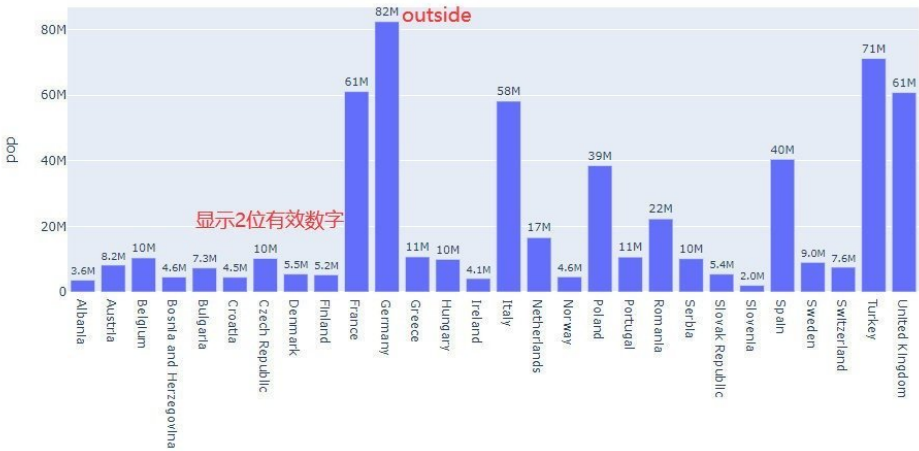
```
3         y=5,
4         text="max number")
5
6     fig.add_annotation(
7         x=4,
8         y=4,
9         text="median number")
10
11     fig.update_annotations(dict(
12         xref="x",
13         yref="y",
14         showarrow=True,
15         arrowhead=7,
16         ax=0,
17         ay=-40
18     ))
```



- 修改x/y轴名称(go)
- 修改X/Y轴名称 (figure实现)
- 坐标轴刻度
- 多子图绘制-1
- 多子图绘制-2
- 冒泡点大小
- 冒泡点颜色
- 添加注释 (某个特殊点)
- 柱状图-信息显示在外面
- 柱状图-改变柱子颜色
- 绘制双坐标
- 柱状图-坐标轴排序
- 饼图-颜色 (自定义)
- go实现
- px实现
- 饼图-文本隐藏
- 饼图-布局和属性设置

柱状图-信息显示在外面

```
1 df = px.data.gapminder().query("continent == 'Europe' and year == 2007 and pop > 2.e6")
2 fig = px.bar(df, x="country", y="pop", text="pop")
3
4 fig.update_traces(texttemplate="%{text:.2s}", # 数字保留2位有效数字
5                   textposition="outside") # 数字显示在外面
6 fig.update_layout(uniformtext_minsize=10, uniformtext_mode="hide")
7
8 fig.show()
```



柱状图-改变柱子颜色

结合颜色随机生成方法

```
1 # 生成颜色的函数
2 def random_color_generator(number_of_colors):
3     color = ["#" + ''.join([random.choice('0123456789ABCDEF') for j in range(6)])
4
```



```
# 改变每个柱状图的颜色
trace1 = Bar(x=[2,3,4,5],
             y=[10,12,16,20],
             marker=dict(color=random_color_generator(4)), # 使用随机函数生成4个颜色，改变柱状图的颜色
             opacity=0.7
            )

data = [trace0,trace1]
```

绘制双坐标

```
In [15]: import pandas as pd
import numpy as np

df = pd.DataFrame({"地区":["华南","华东","华北","东南","东北","西南"],
                  "产量":[3000,4000,2000,1000,2500,3500],
                  "财政收入":[28000,30000,15000,20000,12000,23000]
                 })

df
```

Out[15]:

	地区	产量	财政收入
0	华南	3000	28000
1	华东	4000	30000
2	华北	2000	15000
3	东南	1000	20000
4	东北	2500	12000
5	西南	3500	23000

```
In [16]: mean_chanliang = df["产量"].mean()
mean_chanliang
```

Out[16]: 2666.6666666666665

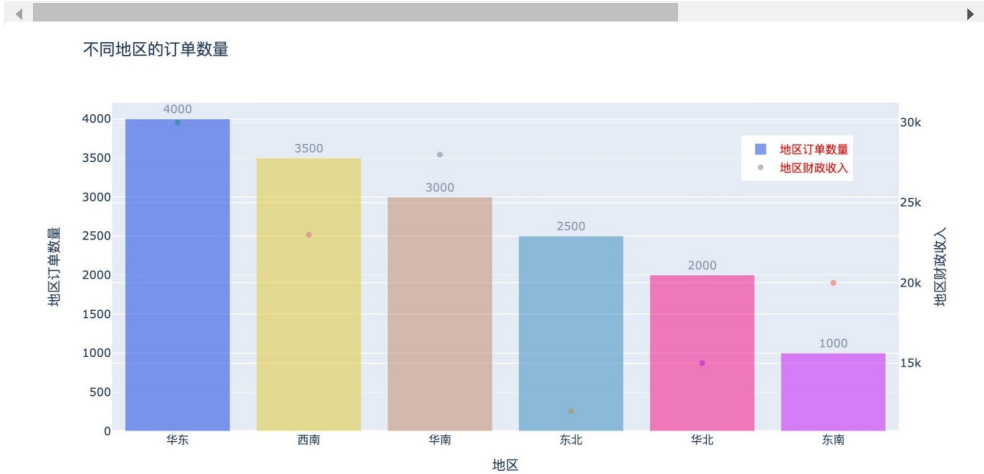
```
In [17]: y = np.tile(mean_chanliang,6) # 将平均数复制6份
y
```

Out[17]: array([2666.66666667, 2666.66666667, 2666.66666667, 2666.66666667, 2666.66666667, 2666.66666667])

```
1 # df.sort_values("产量",ascending=False): 将产量进行降序排列，图形更直观
2 # 排序函数: sort_values("col",ascending=True) 根据某个字段，默认是升序
3
4 trace1 = go.Bar(
5     x=df.sort_values("产量",ascending=False)["地区"].values.tolist(),
6     y=df.sort_values("产量",ascending=False)["产量"].values.tolist(),
7     marker=dict(color=random_color_generator(6)),
8     # 指定显示的文本，类似y值
9     text=df.sort_values("产量",ascending=False)["产量"].values.tolist(),
10    textposition="outside", # 文本的位置
11    opacity=0.5,
12    name="地区订单数量"
13 )
14
15 trace2 = go.Scatter(
16     x=df["地区"].values.tolist(),
17     y=df["财政收入"].values.tolist(),
18     mode="markers",
19     marker=dict(color=random_color_generator(6)),
20     opacity=0.5,
21     name="地区财政收入",
22     yaxis="y2" # 这是第二条y轴
23 )
24
25 data = [trace1,trace2] # 添加图形的轨迹数据
26
27 layout = go.Layout(title="不同地区的订单数量",
28                    xaxis=dict(title="地区"), # 共用x轴
29                    yaxis=dict(title="地区订单数量"), # 第一条y轴的名字
30                    yaxis2=dict(title="地区财政收入",overlaping="y", side="right"), # 第二
31                    legend=dict(x=0.8,y=0.9,font=dict(size=12,color="red")) # 图例的位置 (
32                    )
33
```

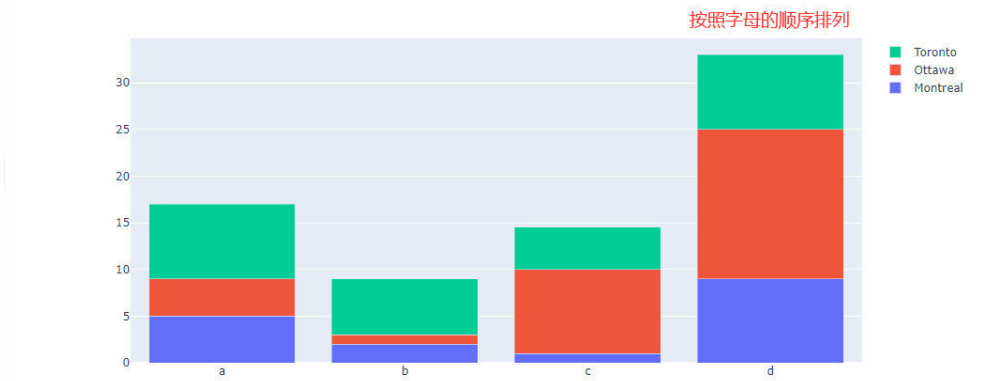
- 修改x/y轴名称(go)
- 修改X/Y轴名称 (figure实现)
- 坐标轴刻度
- 多子图绘制-1
- 多子图绘制-2
- 冒泡点大小
- 冒泡点颜色
- 添加注释 (某个特殊点)
- 柱状图-信息显示在外面
- 柱状图-改变柱子颜色
- 绘制双坐标
- 柱状图-坐标轴排序
- 饼图-颜色 (自定义)
 - go实现
 - px实现
- 饼图-文本隐藏
- 饼图-布局和属性设置



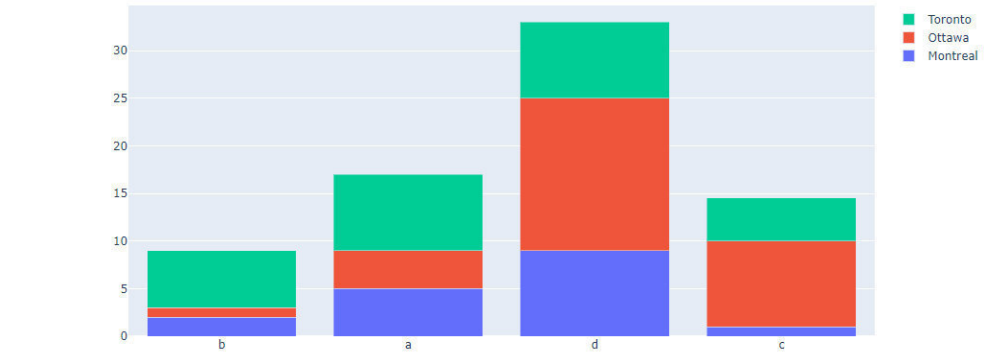


柱状图-坐标轴排序

```
[30]: x=['b', 'a', 'c', 'd']
fig = go.Figure(go.Bar(x=x, y=[2,5,1,9], name='Montreal'))
fig.add_trace(go.Bar(x=x, y=[1, 4, 9, 16], name='Ottawa'))
fig.add_trace(go.Bar(x=x, y=[6, 8, 4.5, 8], name='Toronto'))
fig.update_layout(barmode='stack', xaxis=[{'categoryorder': 'category ascending'}])
fig.show()
```



```
[38]: x=['b', 'a', 'c', 'd']
fig = go.Figure(go.Bar(x=x, y=[2,5,1,9], name='Montreal'))
fig.add_trace(go.Bar(x=x, y=[1, 4, 9, 16], name='Ottawa'))
fig.add_trace(go.Bar(x=x, y=[6, 8, 4.5, 8], name='Toronto'))
fig.update_layout(barmode='stack', xaxis=[{'categoryorder': 'array', 'categoryarray': ['b', 'a', 'd', 'c']}]
fig.show()
```



饼图-颜色（自定义）

go实现

- 修改x/y轴名称(go)
- 修改X/Y轴名称（figure实现）
- 坐标轴刻度
- 多子图绘制-1
- 多子图绘制-2
- 冒泡点大小
- 冒泡点颜色
- 添加注释（某个特殊点）
- 柱状图-信息显示在外面
- 柱状图-改变柱子颜色
- 绘制双坐标
- 柱状图-坐标轴排序
- 饼图-颜色（自定义）
 - go实现
 - px实现
- 饼图-文本隐藏
- 饼图-布局和属性设置

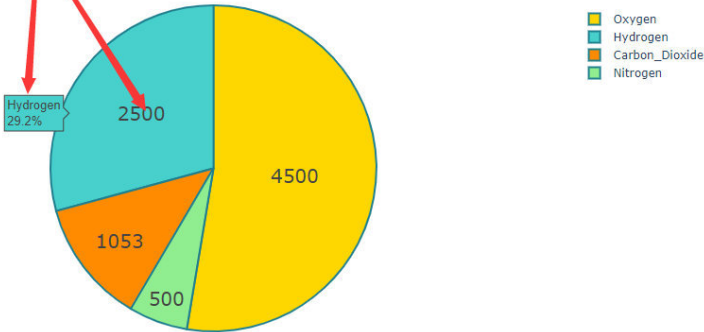


皮大大
489 篇文章

plotly-express-22-plotly使用技巧大全

```
values=[4500,2500,1053,500]))
# 更新饼图配置
fig.update_traces(hoverinfo='label+percent', # 悬停信息
                  textinfo='value', # 饼图中显示的信息
                  textfont_size=20,
                  marker=dict(color=colors, line=dict(color="#208090", width=2)))
fig.show()
```

线条的颜色和宽度设置



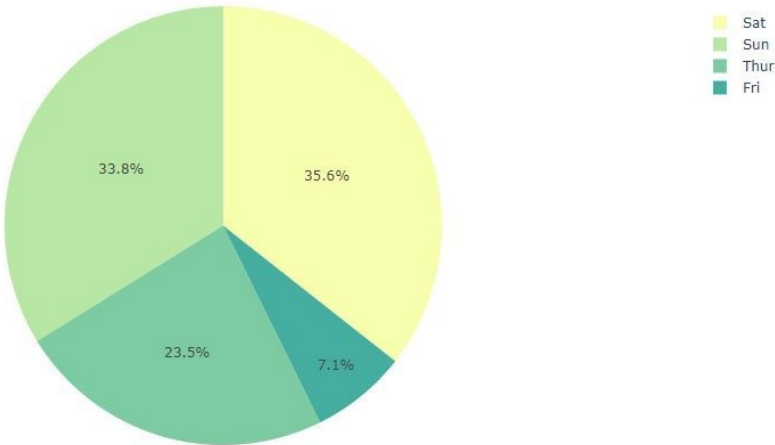
px实现
字典形式

```
1 fig = px.pie(df, values="tip",names="day",color="day",
2             color_discrete_map={'Thur':'lightcyan',
3                                 'Fri':'cyan',
4                                 'Sat':'royalblue',
5                                 'Sun':'darkblue'})
```

通过序列形式

通过 color_discrete_sequence=px.colors.sequential.Bluy1 来实现

```
1 df = px.data.tips()
2 # 设置饼图的颜色: px.colors.sequential.Bluy1
3 fig = px.pie(df, values="tip", names="day",color_discrete_sequence=px.colors.sequential.
4 fig.show()
```



饼图-文本隐藏

```
1 df = px.data.gapminder().query("continent == 'Asia'")
2 fig = px.pie(df, values='pop', names='country')
3
4 fig.update_traces(textposition='inside') # 文字信息在里面
5 fig.update_layout(uniformtext_minsize=15, # 文本信息的最小值
6                  uniformtext_mode='hide' # 小于最小值则被隐藏
7                  )
8 fig.show()
```

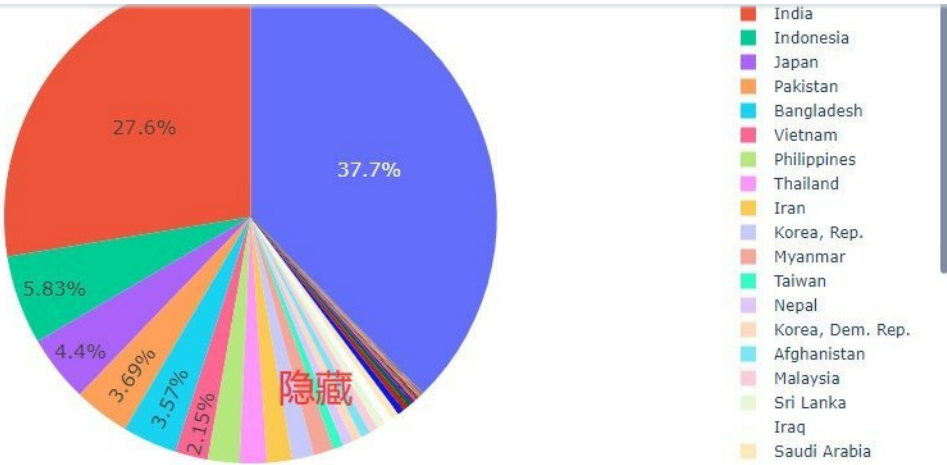
- 修改x/y轴名称(go)
- 修改X/Y轴名称 (figure实现)
- 坐标轴刻度
- 多子图绘制-1
- 多子图绘制-2
- 冒泡点大小
- 冒泡点颜色
- 添加注释 (某个特殊点)
- 柱状图-信息显示在外面
- 柱状图-改变柱子颜色
- 绘制双坐标
- 柱状图-坐标轴排序
- 饼图-颜色 (自定义)
 - go实现
 - px实现
- 饼图-文本隐藏
- 饼图-布局和属性设置





皮大大
489 篇文章

plotly-express-22-plotly使用技巧大全

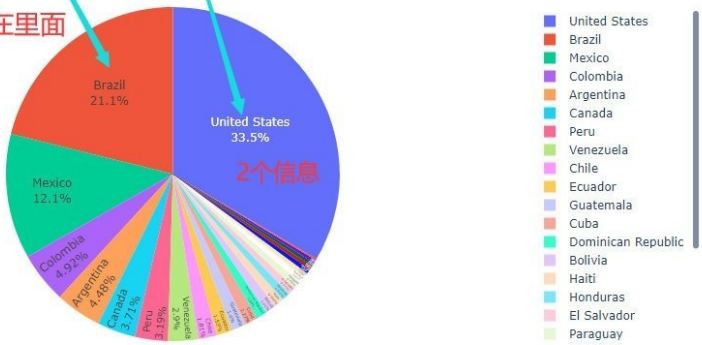


饼图-布局和属性设置

```
fig.update_traces(textposition='inside', textinfo='percent+label') # 将文本的信息放在里面
fig.show()
```

Population of American continent

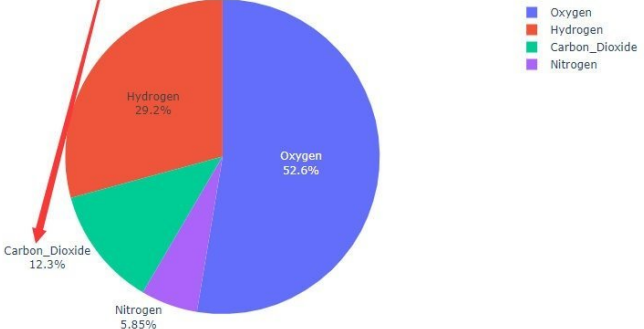
信息在里面



2个信息

饼图-文本位置 (3种)

```
In [21]: labels = ['Oxygen', 'Hydrogen', 'Carbon_Dioxide', 'Nitrogen']
         values = [4500, 2500, 1053, 500]
         fig = go.Figure(data=[go.Pie(labels=labels, values=values, textinfo='label+percent',
         insidetextorientation='horizontal' # "horizontal", "radial", "tangential"
         ))
         fig.show()
```



百分比实现

将各个类别的数量变成百分比

- 修改x/y轴名称(go)
- 修改X/Y轴名称 (figure实现)
- 坐标轴刻度
- 多子图绘制-1
- 多子图绘制-2
- 冒泡点大小
- 冒泡点颜色
- 添加注释 (某个特殊点)
- 柱状图-信息显示在外面
- 柱状图-改变柱子颜色
- 绘制双坐标
- 柱状图-坐标轴排序
- 饼图-颜色 (自定义)
- go实现
- px实现
- 饼图-文本隐藏
- 饼图-布局和属性设置



皮大大

489 篇文章

plotly-express-22-plotly使用技巧大全

```
Out[19]: ['53.66%',
          '7.43%',
          '6.29%',
          '4.85%',
          '3.45%',
          '2.50%',
          '2.47%',
          '2.32%',
          '2.00%',
          '1.92%',
          '1.70%',
          '1.58%',
          '1.56%',
          '1.45%',
          '1.13%',
          '1.04%',
          '1.00%',
          '0.89%',
          '0.88%',
          '0.79%',
          '0.64%',
          '0.32%',
          '0.14%']
```

颜色随机生成（优秀）

这个方法很巧妙，能够用在任何绘制的图形中，只要有多个颜色出现：只需要在color参数中调用函数即可实现

```
1 # 颜色的随机生成: #123456 # 加上6位数字构成
2 def random_color_generator(number_of_colors):
3     color = ["#" + ''.join([random.choice('0123456789ABCDEF') for j in range(6)])
4               for i in range(number_of_colors)]
5     return color
6
7 trace = go.Bar(
8     x = cnt_srs.index,
9     y = cnt_srs.values,
10    text = text,
11    marker = dict(
12        color = random_color_generator(100),
13        line = dict(color='rgb(8, 48, 107)',
14                    width = 1.5)
15    ),
16    opacity = 0.7
17 )
```

坐标轴倾斜

将坐标轴的文字信息倾斜显示

```
1 # 图表标题
2 fig.update_layout(title_text='January 2013 Sales Report',xaxis_tickangle=-45) # + x轴标签
```



Plotly实现表格

如何使用Plotly实现[表格](#)

- 修改x/y轴名称(go)
- 修改X/Y轴名称（figure实现）
- 坐标轴刻度
- 多子图绘制-1
- 多子图绘制-2
- 冒泡点大小
- 冒泡点颜色
- 添加注释（某个特殊点）
- [柱状图-信息显示在外面](#)
- 柱状图-改变柱子颜色
- 绘制双坐标
- 柱状图-坐标轴排序
- 饼图-颜色（自定义）
 - go实现
 - px实现
- 饼图-文本隐藏
- 饼图-布局和属性设置



皮大大

489 篇文章

plotly-express-22-plotly使用技巧大全

1

0

```
                y=[2,4,6,8])

        trace1 = Bar(x=[2,3,4,5],
                    y=[10,12,16,20])

        data = [trace0,trace1]

In [3]: # 改变目录, os模块

import os
os.chdir("/Users/piqianchao/Desktop/project/px") # 保存到指定目录下

In [4]: # plot是生成HTML格式的文件
py.offline.plot(data,filename="test2.html")

Out[4]: 'test2.html'

In [5]: # iplot函数是用于在notebook中直接生成图片
# 不要filename属性
py.offline.iplot(data) # 如果不改变路径, 此时文件是生成在当前的目录下, 同级目录
```

kaleido保存图片

```
1 from kaleido.scopes.plotly import PlotlyScope
2 import plotly.graph_objects as go
3 scope = PlotlyScope()
4
5 fig = go.Figure(data=[go.Scatter(y=[1, 3, 2])])
6 with open("figure.png", "wb") as f: # 在本地目录下面就会生成文件
7     f.write(scope.transform(fig, format="png"))
```

图例设置

对于图例设置的技巧, 主要包含:

整体基本设置

修改图例名称

隐藏图例入口 (第一个图例)

图例位置显示

自定义优美图例

图例散点大小设置

组图例设置

标题设置

本文参与 腾讯云自媒体分享计划, 欢迎热爱写作的你一起参与!

本文分享自作者个人站点/博客: <http://www.renpeter.cn> 复制

如有侵权, 请联系 cloudcommunity@tencent.com 删除。

Go

举报

点赞 1

分享

登录后参与评论

修改x/y轴名称(go)

修改X/Y轴名称 (figure实现)

坐标轴刻度

多子图绘制-1

多子图绘制-2

冒泡点大小

冒泡点颜色

添加注释 (某个特殊点)

柱状图-信息显示在外面

柱状图-改变柱子颜色

绘制双坐标

柱状图-坐标轴排序

饼图-颜色 (自定义)

go实现

px实现

饼图-文本隐藏

饼图-布局和属性设置

https://cloud.tencent.com/developer/article/1794886

10/13

1

0

皮大大
489 篇文章

plotly-express-22-plotly使用技巧大全

相关文章

Idea使用技巧大全

idea 2017破解的方法在上面的网页中 很方便

MonroeCode

Jupyter notebook使用技巧大全

Jupyter Notebook是一款开源的web应用，它允许使用者创建和分享包含代码，公式，可视化图表和纯文本的文档，并支持多种编程语言的交互式计算，对于py...

超哥的杂货铺

IntelliJ IDEA 使用技巧大全，速度收藏！

如果说IntelliJ IDEA是一款现代化智能开发工具的话，Eclipse则称得上是石器时代的东西了。其实笔者也是一枚从Eclipse转IDEA的探索者，随着...

用户5224393

总结IntelliJ IDEA 使用技巧大全，速度收藏！

如果说IntelliJ IDEA是一款现代化智能开发工具的话，Eclipse则称得上是石器时代的东西了。其实笔者也是一枚从Eclipse转IDEA的探索者，随着...

用户5224393

SAP 绝密小技巧大全

1.A:在公司代码分配折旧表时报错? 在公司代码分配折旧表时报错，提示是“3000 的公司代码分录不完全 - 参见长文本” 希望各位大侠帮我看看。 ...

matinal

办公 | word实用操作技巧大全可收藏

哈喽各位小伙伴，大家好！我是橙c一个爱喝橙汁的橙c。今天一早啃着馒头去上班有点落魄把我冷的瑟瑟发抖。

IT小侠公社

DataGridView控件使用大全

DataGridView控件使用大全：http://www.haogongju.net/art/847949

跟着阿笨一起玩NET

Android Studio使用大全

开发工具：Android Studio 2.1.2（mac版本） 开发环境：OS X El Capitan 版本 10.11.5

专注APP开发

- 修改x/y轴名称(go)
- 修改X/Y轴名称（figure实现）
- 坐标轴刻度
- 多子图绘制-1
- 多子图绘制-2
- 冒泡点大小
- 冒泡点颜色
- 添加注释（某个特殊点）
- 柱状图-信息显示在外面
- 柱状图-改变柱子颜色
- 绘制双坐标
- 柱状图-坐标轴排序
- 饼图-颜色（自定义）
 - go实现
 - px实现
- 饼图-文本隐藏
- 饼图-布局和属性设置





皮大大
489 篇文章

plotly-express-22-plotly使用技巧大全

git reset --mixed: 此为默认方式，不带任何参数的git reset，即时这种方式，它回退到某个版本，只保留源码，回退commit和index信息 ...

MickyInvQ

Find 命令使用大全

root@localhost ~# find /etc/ -name passwd

用户8418197

.htaccess文件使用大全

这里收集的是各种实用的 .htaccess 代码片段，你能想到的用法几乎全在这里。

星哥玩云

【Docker】命令使用大全

容器的出现是为了解决环境配置的问题，虚拟机是一种解决方案，但是也有一些缺点，再后来出现了 Linux 容器，Docker 属于 Linux 容器的一种封装，提供...

小白学视觉

jupyter notebook使用大全

使用python的人不可能不会接触jupyter notebook，因为它太实用了，尤其是边写代码边调试的时候。

锦小年

Nginx基本使用大全

输出Nginx的基本配置信息,其中/usr/local/etc/nginx/nginx.conf就是Nginx 的配置文件

Coxhuang

【Docker】命令使用大全

容器的出现是为了解决环境配置的问题，虚拟机是一种解决方案，但是也有一些缺点，再后来出现了 Linux 容器，Docker 属于 Linux 容器的一种封装，提供...

机器视觉CV

Xmind使用技巧(xmind使用技巧)

之前讲到思维导图笔记法的时候，给大家推荐了一款免费的思维导图软件—xmind软件，然后好多小伙伴都加我问关于思维导图的使用方法。

全栈程序员站长

linux Dig命令使用大全

Dig是一个在类Unix命令行模式下查询DNS包括NS记录，A记录，MX记录等相关信息的工具。由于一直缺失Dig man page文档，本文就权当一个dig使用...

碰漏

修改x/y轴名称(go)

修改X/Y轴名称（figure实现）

坐标轴刻度

多子图绘制-1

多子图绘制-2

冒泡点大小

冒泡点颜色

添加注释（某个特殊点）

柱状图-信息显示在外面

柱状图-改变柱子颜色

绘制双坐标

柱状图-坐标轴排序

饼图-颜色（自定义）

go实现

px实现

饼图-文本隐藏

饼图-布局和属性设置



皮大大

489 篇文章

plotly-express-22-plotly使用技巧大全

本期先来学习Button的两个子控件，无论是单选还是复选，在实际开发中都是使用的较多的控件，相信通过本期的学习即可轻松掌握。 一、CheckBox ...

分享达人秀

更多文章

1

0

专栏文章

阅读清单

互动问答

技术沙龙

技术视频

团队主页

腾讯云TI平台

活动

自媒体分享计划

邀请作者入驻

自荐上首页

技术竞赛

资源

技术周刊

社区标签

开发者手册

开发者实验室

关于

视频介绍

社区规范

免责声明

联系我们

友情链接

修改x/y轴名称(go)

修改X/Y轴名称 (figure实现)

坐标轴刻度

多子图绘制-1

多子图绘制-2

冒泡点大小

冒泡点颜色

添加注释 (某个特殊点)

柱状图-信息显示在外面

柱状图-改变柱子颜色

绘制双坐标

柱状图-坐标轴排序

饼图-颜色 (自定义)

go实现

px实现

饼图-文本隐藏

饼图-布局和属性设置

热门产品	域名注册	云服务器	区块链服务	消息队列	网络加速	云数据库	域名解析	云存储
热门推荐	人脸识别	腾讯会议	企业云	CDN 加速	视频通话	图像分析	MySQL 数据库	SSL 证书
更多推荐	数据安全	负载均衡	短信	文字识别	云点播	商标注册	小程序开发	网站监控

Copyright © 2013 - 2023 Tencent Cloud. All Rights Reserved. 腾讯云 版权所有 京公网安备 11010802017518 粤B2-20090059-1

https://cloud.tencent.com/developer/article/1794886

13/13