

## Blot-1.0.0文档

### 图表示例

折线图 (line)  
散点图 (scatter、scatter3d)  
柱状图 (bar)  
箱形图 (box)  
直方图 (histogram)  
热力图 (heatmap)  
气泡图 (bubble、bubble3d)  
对比图 (spread、ratio)  
扇形图 (pie)

### Figure对象增强

保存静态图片: figure.save()

### 配置参数

#### 基础参数

kind  
data  
layout

#### 列名参数

x  
y  
y2  
z  
labels  
values  
text  
keys  
upper  
lower

#### 标题参数

title  
xtitle  
ytitle  
y2title  
ztitle

#### 属性参数

theme  
colors  
colorscale  
fill  
fillcolor  
width  
dash  
mode  
interpolation  
connectgaps

- symbol
- size
- bounds\_color
- anomaly\_color
- barmode
- bargap
- bargroupgap
- bins
- histnorm
- boxpoints
- 图例参数
  - orientation
  - annotations
  - vlines
  - hlines
- 图表参数
  - gridcolor
  - zerolinecolor
  - margin
  - dimensions
  - subplots
  - as\_figure
- 其他参数
  - 饼图参数
  - 热力图参数
  - 子图参数
  - shape参数
  - 轴参数
  - slider参数
  - 注解参数

## Blot-1.0.0文档

blot基于plotly和cufflinks，是一个支持多进程，比plotly更简约，比cufflinks更快的python动态图可视化库。更贴近日常使用场景，便捷的绘制基带、简单的静态图像转换存储...结合pandas的Series和DataFrame使用（monkey patching），仅需一行代码即可完成各类图表构建。

## 图表示例

### 折线图（line）

```
import blot # 后续代码均省略此行，但实际使用中不可省略！
pd.Series(np.random.randint(1,10,10)).blot()
```

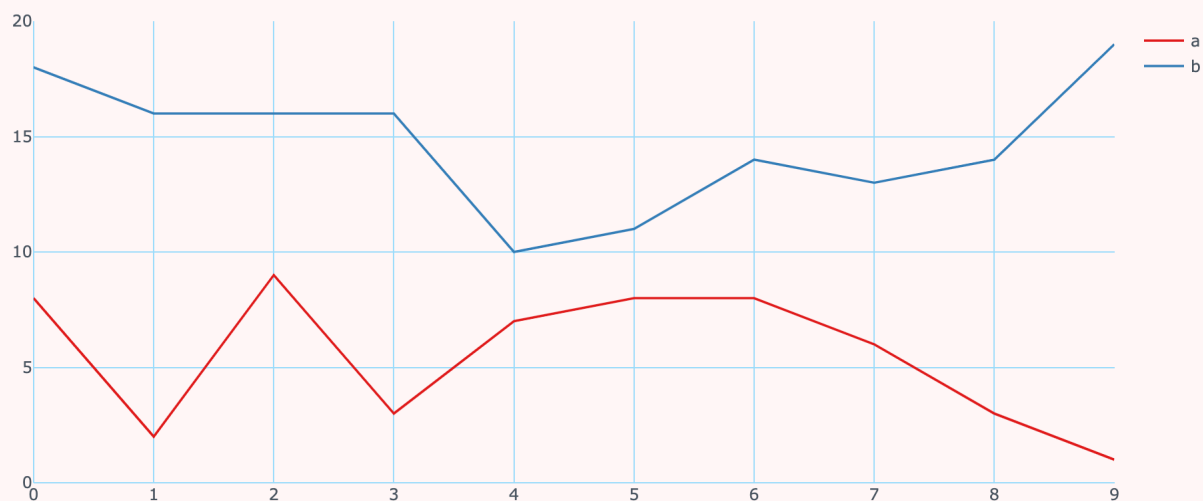
默认绘图类型即为「折线图」，在使用Series调用blot中，默认将index作为x轴，支持数值、字符串、datetime对象等，并进行显示优化



```
pd.DataFrame({
    'a':np.random.randint(1,10,10),
    'b':np.random.randint(10,20,10)
}).blot()
```

使用Dataframe调用时，默认会将所有可作为y轴的列视为一条曲线，并添加legend。也可手动指定x轴、y轴使用的数据。

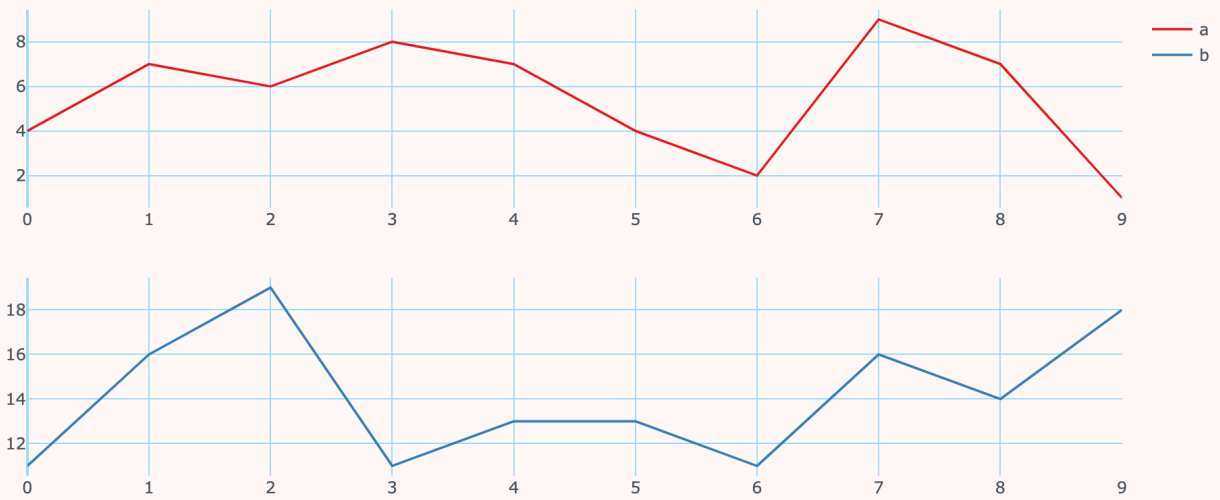
注：无特殊说明外，Series和DataFrame在使用blot中均无差别，后续示例只保留一个



```
pd.DataFrame({
    'a':np.random.randint(1,10,10),
    'b':np.random.randint(10,20,10)
}).blot(subplots=(2,1))
```

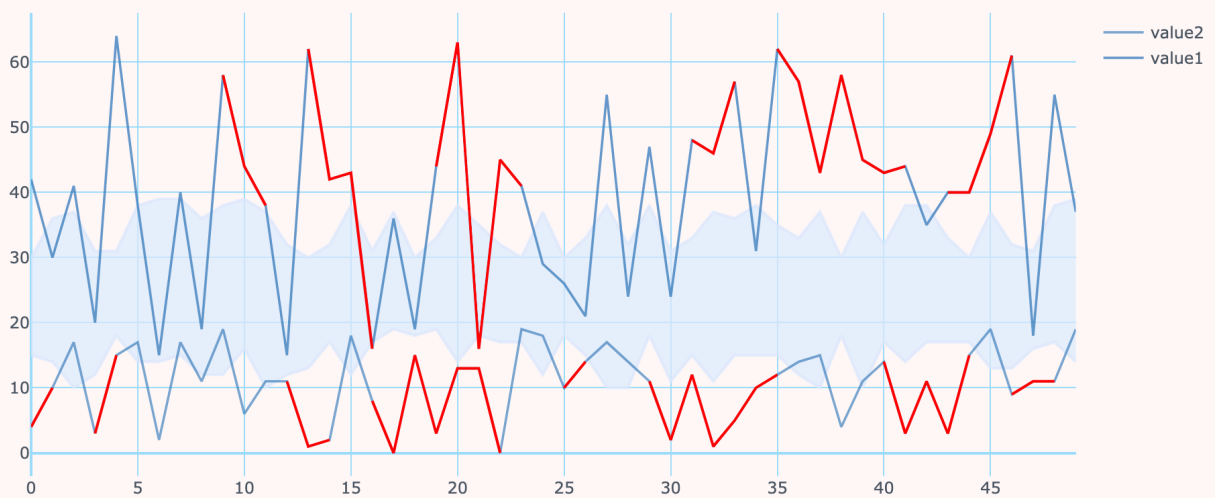
如果想要把多条曲线以子图的形式呈现，只需要加上subplots参数即可，如果给出(rows, cols)则会按照指定的形状布局子图。若简单指定subplots=True, 则默认按照两列的布局展示，即(n, 2)

注：其他图表设定子图类似，后续不再赘述



```
pd.DataFrame({
    'upper':np.random.randint(30,40,50),
    'lower':np.random.randint(10,20,50),
    'value1':np.random.randint(15,65,50),
    'value2':np.random.randint(0,20,50),
}).plot(upper='upper', lower='lower',color='blue')
```

通过upper、lower指定上下界基带对应的列名。如果只指定其一，则绘制单边基带。  
bounds\_color、anomaly\_color改变基带和异常点颜色

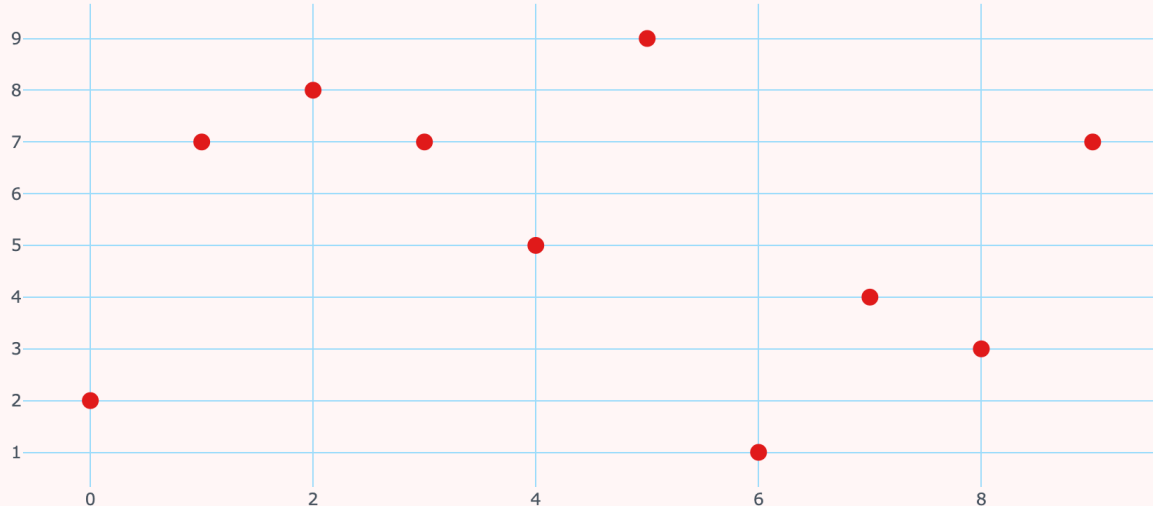


## 散点图 (scatter、scatter3d)

```
pd.Series(np.random.randint(1,10,10)).plot(kind='scatter')
```

散点图和「折线图」所需数据完全相同。并且可以相互转换。区别只是当kind='line'时，mode='lines'，当kind='scatter'时，mode='markers'。可以通过修改mode参数来进行切换。甚至指定mode='lines+markers'对两者进行组合。

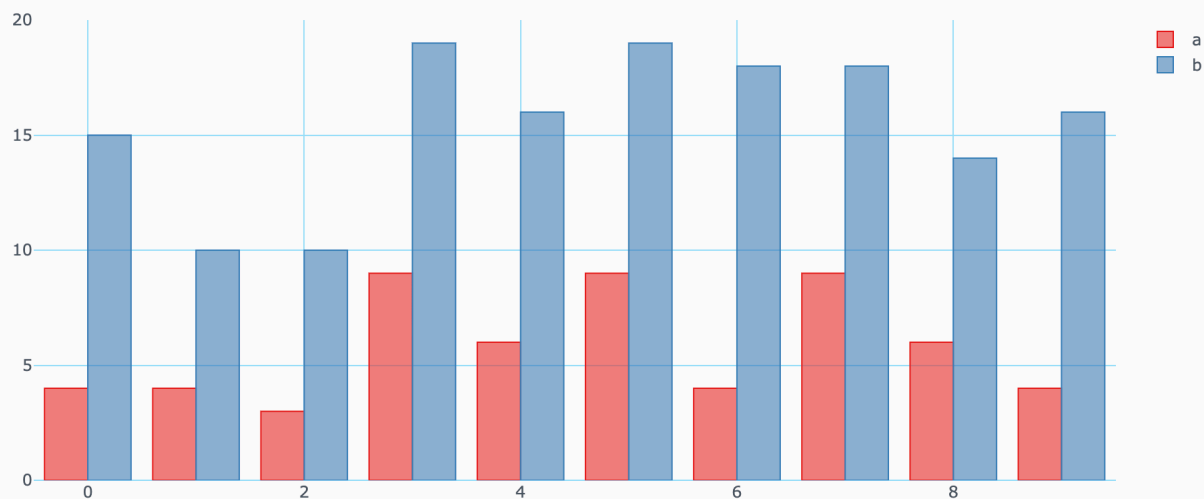
绘制3d图时，修改kind为scatter3d，且需要手动指定x, y, z轴对应的列名



## 柱状图 (bar)

```
pd.DataFrame({  
    'a':np.random.randint(1,10,10),  
    'b':np.random.randint(10,20,10)  
}).plot(kind='bar')
```

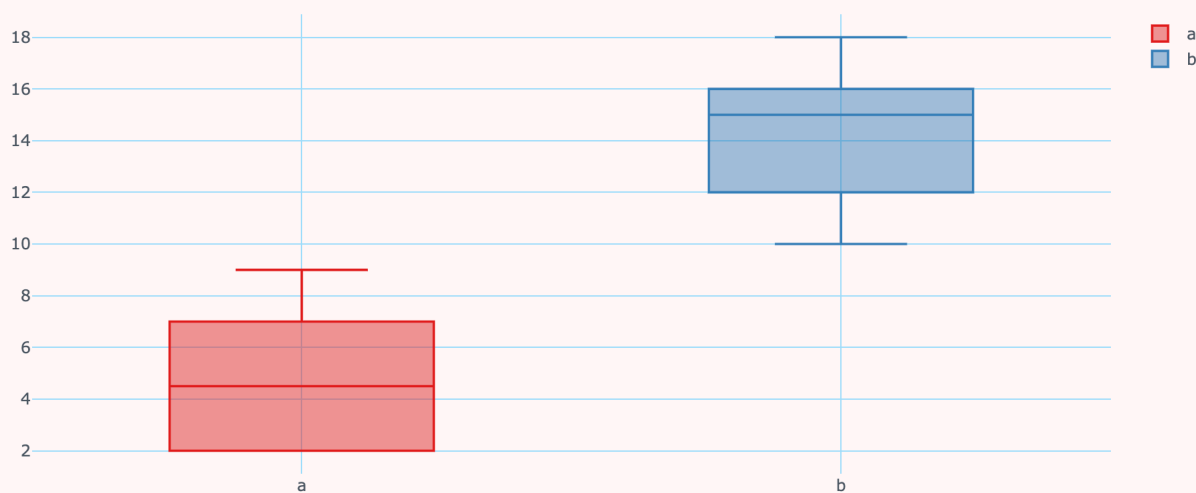
默认多维柱状图是通过「多头 (group)」展示，可以通过修改barmode参数切换展示方式 ['stack', 'overlay']



## 箱形图 (box)

```
pd.DataFrame({  
    'a': np.random.randint(1, 10, 10),  
    'b': np.random.randint(10, 20, 10)  
}).boxplot(kind='box')
```

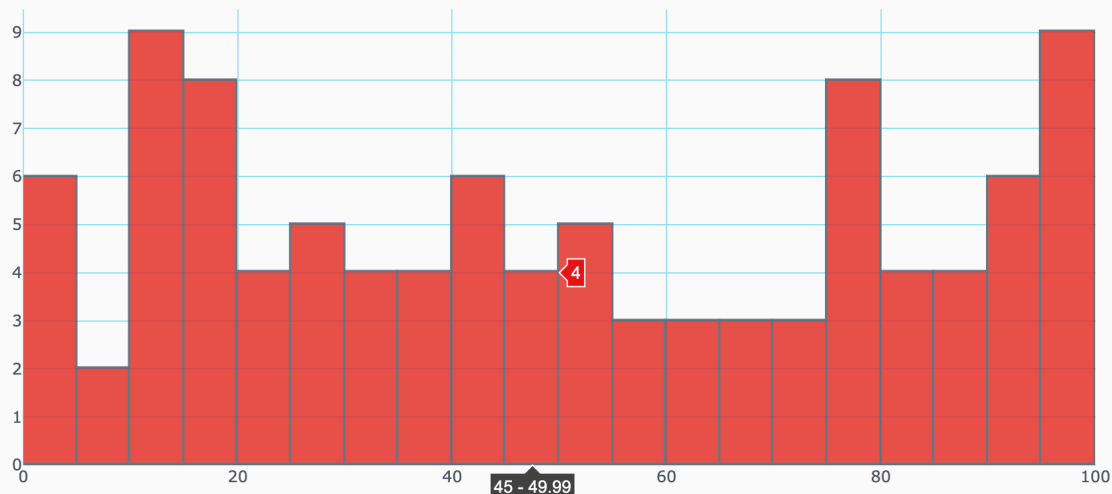
默认不对原始离散数据点做展示, 可通过boxpoints参数控制显示方式



## 直方图 (histogram)

```
pd.Series(np.random.rand(100)*100).plot(kind='histogram', bins=20)
```

分布直方图，通过bins控制分桶的数量。默认纵坐标为频数（frequency），可以通过histnorm参数控制纵坐标值类型（频数，百分数，频率，密度，频率密度）

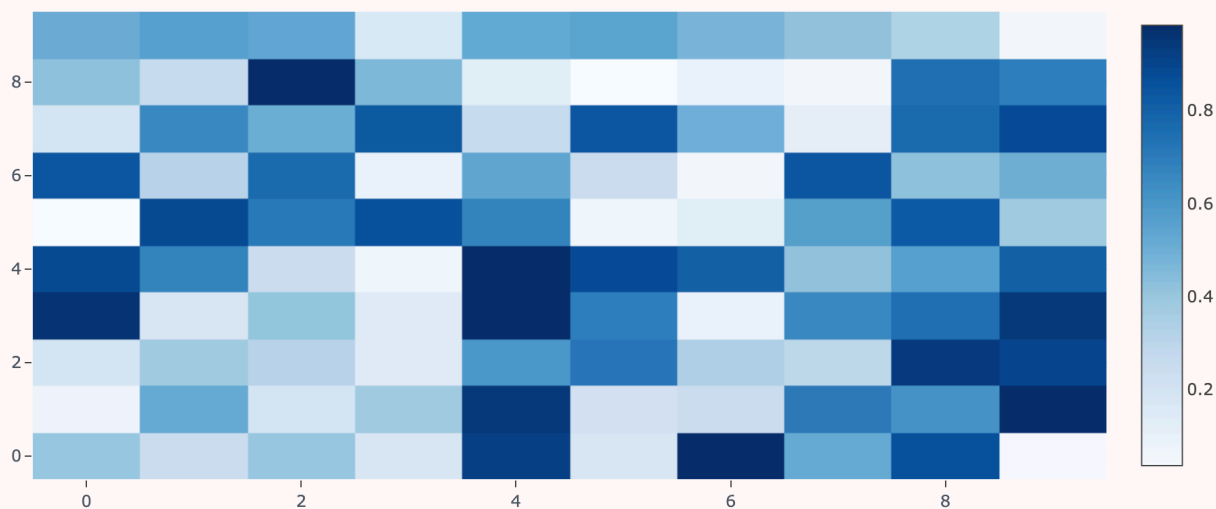


## 热力图 (heatmap)

```
pd.DataFrame(np.random.rand(10,10)).plot(kind='heatmap', colorscale='blues')
```

只有DataFrame支持，适合两个离散维度的关联分析。形状为m\*n的DataFrame对应的热力图也为m\*n，热力图的X轴默认取值为df.columns，Y轴取值默认为df.index。

若数据为columns=[key1, key2, value]的形式，可以用df.pivot(index='key1', columns='key2', values='value')先转换，也可以直接指定plot(x='key1', y='key2', z='value')



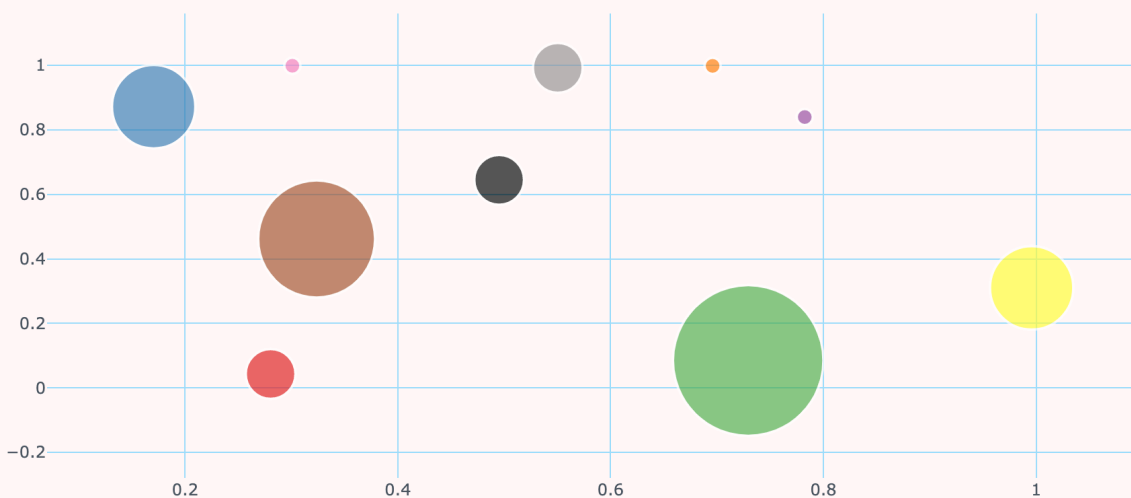
## 气泡图 (bubble、bubble3d)

```
pd.DataFrame({
    'a':np.random.rand(10),
    'b':np.random.rand(10),
    'size':np.random.randint(5,10,10)
}).plot(kind='bubble',x='a',y='b',size='size')
```

只有DataFrame支持，适合两个连续维度的二维分析。需手动指定「x, y, size」三个参数（参数和scatter很像，但是scatter的size只支持定值，而后者是变量）

默认无hover显示，指定text='value'即可将value列取值作为hover显示

绘制3d图时，修改kind为bubble3d，且需要手动指定x, y, z轴对应的列名



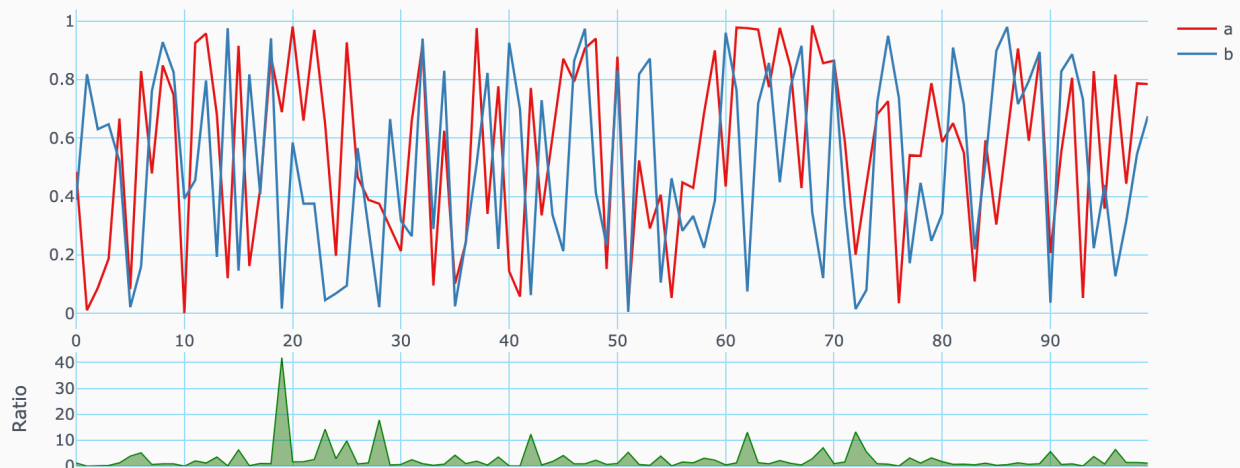
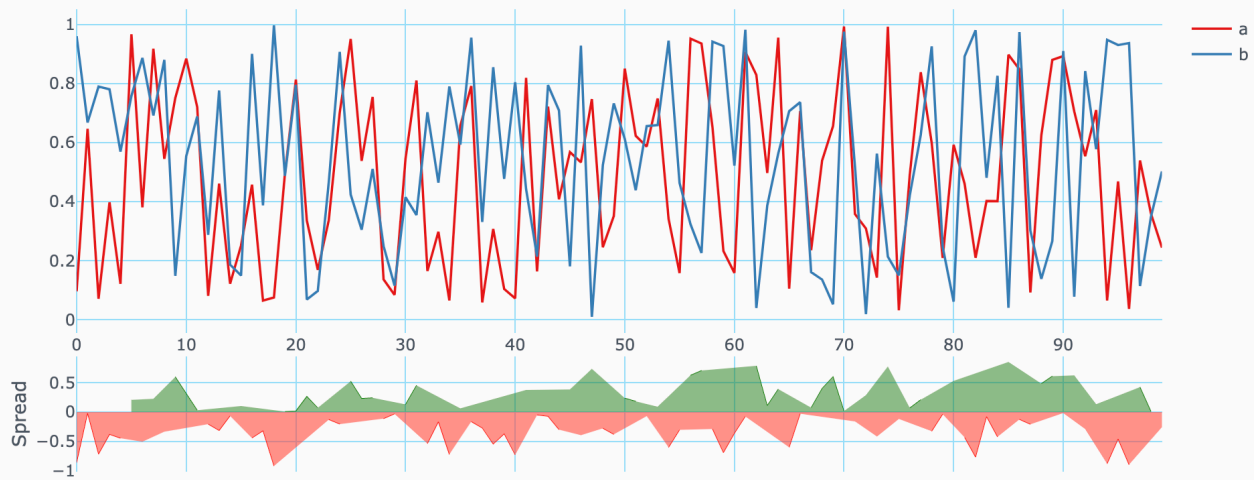


## 对比图 (spread、ratio)

```
pd.DataFrame({  
    'a':np.random.rand(100),  
    'b':np.random.rand(100),  
}).blot(kind='spread')
```

kind=spread时, 会增加一个子图表示a-b的取值情况

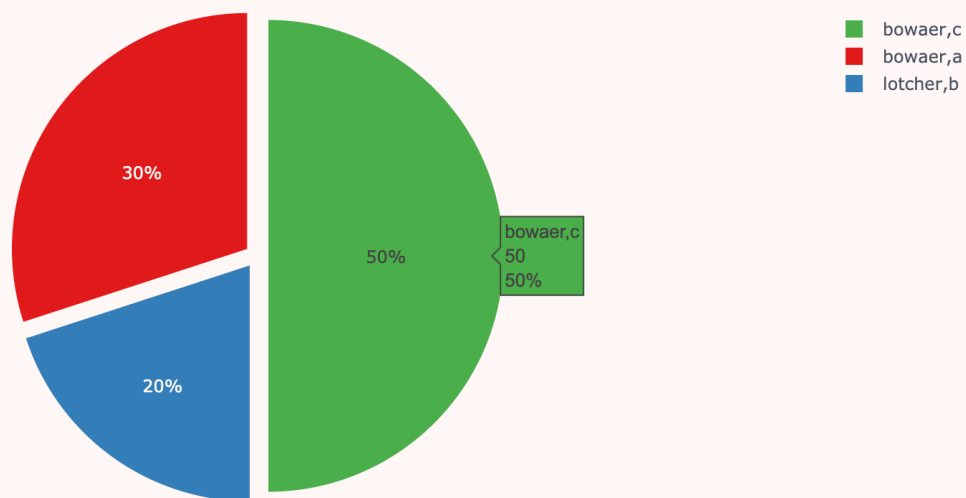
kind=ratio时, 会增加一个子图表示a/b的取值情况



## 扇形图 (pie)

```
pd.DataFrame({
    'name': ['bowaer', 'lotcher', 'bowaer'],
    'subname': ['a', 'b', 'c'],
    'value': [30, 20, 50]
}).blot(kind='pie', labels=['name', 'subname'], values='value', pull=0.05)
```

必须手动指定labels和values两个参数，默认legend会按照取值大小排序，如果不需要，指定sort=False即可



## Figure对象增强

### 保存静态图片：figure.save()

支持png、jpeg、webp、svg、pdf、eps等多种格式  
eg. `pd.DataFrame([1,2,3]).blot().save('test.png')`

## 配置参数

### 基础参数

## kind

【string】可选的绘制图表类型

```
line: 折线图
scatter: 散点图
scatter3d: 3d散点图
bar: 柱状图
histogram: 直方图
box: 箱形图
spread: 折线图对比, 自动增加fig1-fig2子图
ratio: 折线图对比, 自动增加fig1/fig2子图
heatmap: 热力图
surface: 曲面图
bubble: 气泡图
bubble3d: 3d气泡图
pie: 扇形图
```

## data

【Data】Plotly Data 对象.

一般无需显式指定, 可自动从DataFrame对象生成

## layout

【Layout】Plotly layout 对象

一般显式指定, 可自动从DataFrame对象生成

## 列名参数

### x

【str】代表x轴取值的列名

默认为df.index

### y

【str or list】代表x轴取值的列名

默认为df.columns

## y2

【str or list(str)】代表第二y轴取值的列名

用法同y相同，也可以和y同时使用

## z

【str】代表z轴取值的列名

仅在3d图及heatmap中生效

## labels

【str or list】Pie图中label对应的列名

```
str : 单列作为label  
list: 多列的组合作为label  
# 仅在kind='pie'时有效
```

## values

【str】Pie图中取值对应的列名

```
# 仅在kind='pie'时有效
```

## text

【string】包含text取值的列名

用于mode、hover等参数

## keys

【list】参与绘图的df列名

```
默认为df.columns  
也可以用作排序，作用类似于df[keys]
```

## upper

【str】基带上界的df列名

## lower

【list】基带下界的df列名

## 标题参数

### title

【str】 图像标题

### xtitle

【str】 x轴标题

### ytitle

【str】 y轴标题

### y2title

【str】 y2轴标题

### ztitle

【str】 z轴标题

## 属性参数

### theme

【str】 Laytou主题

```
bowaer  
solar  
pearl  
white  
.....
```

通过`blot.getThemes()` 查看所有内置可用主题，默认主题为bowaer

## colors

【str、list、dict】 trace颜色

```
str: 符合plotly规范的颜色字符串: hex(e.g. '#ff0000')、rgb/rgba(e.g.  
'rgba(255,0,0,0.8)')、css color name.....  
dict: {key:color} 为每一条trace单独指定颜色  
list: zip(colors, keys)来指定颜色
```

## colorscale

### 【str】色阶

内置的色阶名称  
若色阶名前带 '-', 则颜色反转  
仅当未设置colors参数时有效  
在`blot.colors.scales()` 查看有效色阶名称

## fill

### 【bool、str、list、dict】填充曲线的方式

填充曲线的方式，具体参数含义参照  
<https://plotly.com/python/reference/scatter/#scatter-fill>  
**bool**: `False`不填充。`True`: 'tozeroy' if kind=='area' else 'tonexty'  
**str**: 填充方式，取值列表: ( "none" | "tozeroy" | "tozeroyx" | "tonexty" | "tonextx" | "toself" | "tonext" )  
**list**: 单独为每条曲线指定填充方式  
**dict**: 同上, key of keys

## fillcolor

### 【str、list、dict】曲线填充的颜色控制

**str**: 填充颜色，和color方式一样。hex(e.g. '#ff0000')、rgb/rgba(e.g. 'rgba(255,0,0,0.8)')、css color name.....  
**list**: 单独为每条曲线指定填充颜色  
**dict**: 同上, key of keys

## width

### 【int、list、dict】线宽

如果不指定，则从theme读取，如果theme没有，则默认为2  
**int** : 应用于所有曲线  
**list** : 按照顺序应用在各条曲线  
**dict**: {column:value} 为每条曲线单独指定线宽

## dash

### 【str、list、dict】线型

solid 「默认」：实线 '——'  
dash：短线 '- - - - '  
dashdot：带点短线 '-.-.-.-.'  
dot：点连线 '.....'

**str**：应用于所有曲线  
**list**：按照顺序应用在各条曲线  
**dict**：{column:value} 为每条曲线单独指定线宽

## mode

### 【str、list、dict】散点图绘制模式

散点图下绘制模式

markers 「默认」  
lines：连线。和kind='line'效果相同  
lines+markers：用线将散点连接  
lines+text：线+注解。text由text参数指定  
markers+text：点+注解  
lines+markers+text：线+点+注解

**str**：应用于所有曲线  
**list**：按照顺序应用在各条曲线  
**dict**：{column:value} 为每条曲线单独指定模式

## interpolation

### 【str】连线方式

linear 「默认」：直线连接  
spline：曲线连接  
vhv：纵横纵的方式连接  
hvh：横纵横的方式连接  
vh：纵横的方式连接  
hv：横纵的方式连接

## connectgaps

### 【bool=False】折线图空值是否连接

## symbol

### 【str】散点形状

仅当mode的取值包含markers时生效

circle「默认」：实心圆点  
circle-dot：空心圆点  
diamond：菱形  
square：正方形  
.....

可参照<https://plotly.com/python/reference/scatter/#scatter-marker-symbol>查看更多有效值

## size

【str、int】散点大小

str：气泡图中气泡大小对应的列名  
int：散点图中散点的大小，定值

## bounds\_color

【str】基带颜色

格式同color。默认='rgba(222, 233, 252, 0.7)'

## anomaly\_color

【str】异常值颜色

格式同color。默认='red'

## barmode

【str】多维柱状图展示形式

# 仅当kind='bar'时有效  
group【默认】  
stack  
overlay

## bargap

【float[0,1)】柱状图组间间隔

# 仅当kind='bar'或者'histogram'时有效



## bargroupgap

【float[0,1)】柱状图组内间隔

```
# 仅当kind='bar'或者'histogram'时有效
```

## bins

【int、tuple】柱状图组内间隔

```
# 仅当kind='histogram'时有效
int: 分桶数量
tuple: (start, end, size)
    start : 横坐标起始截断值
    end: 横坐标结束截断值
    size: 分桶数量
```

## histnorm

【str】直方图y轴取值

```
# 仅当kind='histogram'时有效
'' (frequency) 「默认」: 频数
percent: 百分数 (0-100)
probability: 频率 (0-1)
density: 密度 = frequency/bins
probability density: 概率密度 = probability/bins
```

## boxpoints

【str】直方图y轴取值

```
箱形图 (box) 中散点的展示方式。默认不展示
None 「默认」: 不展示
outliers: 仅展示离群点
all: 展示所有点
suspectedoutliers: 仅展示疑似离群点
```

## 图例参数

### orientation

【str】图例方向

```
# 仅当kind='bar'、'histogram'、'box'时有效
v 「默认」：纵向
h：横向
```

## annotations

【str】图例标注

```
{x_value : text}: {横坐标取值, 标注内容}
```

## vlines

【list】竖直参考线

需要绘制竖直（平行于y轴）参考线的x取值列表

## hlines

【list】水平参考线

需要绘制水平（平行于x轴）参考线的y取值列表

## 图表参数

### gridcolor

【str】网格线颜色

取值同color

### zerolinecolor

【str】坐标轴颜色

取值同color

## margin

【dict、tuple】边界距离

图例和上下左右边界的距离

**tuple**: 按照 (left, right, bottom, top) 给出取值

**dict**: key of ('l', 'r', 'b', 't'), 例如{'l':30} 代表距离左边界30px

## dimensions

### 【tuple(int, int)】图像尺寸

图像的尺寸 (width,height)  
jupyter中默认会充满输出  
返回静态图或者plotly对象默认为 (800, 500)

## subplots

### 【bool、tuple(rows, cols)】子图展示控制

**bool**: 是否以子图的形式展示，默认为False。True则按照2列的形式展示子图  
**tuple**(rows, cols): 子图的栅栏分布

## as\_figure

### 【bool=True】是否返回plotly对象

## 其他参数

### 饼图参数

**sort** : **bool=True**  
legend中各label是否按照占比排序

**pull** : **float [0-1]** or **list(float)**  
**float**: 每块离中心的距离，默认为0  
**list**: 按顺序指定每个块离中心的距离  
饼图中块距离中心点的距离

**hole** : **float [0-1]**  
饼图中心孔的尺寸，默认为0

**textcolor** : **string**  
饼图中显示的文字颜色，默认自适应（深色背景浅色字体，浅色背景深色背景）

**textposition** : **str** or **list(str)**  
饼图中的文字显示的位置 ['inside', 'outside', 'auto', 'none']

**textinfo** : **str**  
饼图中文字显示的信息，默认为percent  
可以为['label', 'text', 'value', 'percent']中的一个  
或者多个用 '+' 连接的字符串，eg. 'label+value'

## 热力图参数

`center_value` : `float`

指定热力图scale的中心取值

如果不指定, 将自动从(`zmin`, `zmax`)中计算

`zmin` : `float`

指定计算colorscale最小的取值。eg. `zmin=0.5`, 则`z<=0.5`的颜色均为colorscale的起始颜色

会影响到整个colorscale `range`

`zmax` : `float`

指定计算colorscale最大的取值

会影响到整个colorscale `range`

## 子图参数

`horizontal_spacing` : `float` [`0`,`1`]

多个水平子图之间的距离, 列间距

`vertical_spacing` : `float` [`0`,`1`]

多个竖直子图之间的距离, 行间距

`subplot_titles` : `bool=False`

是否显示每个子图的title (默认使用legend信息)

`shared_xaxes`、`shared_xaxis` : `bool`

子图间是否共享x轴

`shared_yaxes` : `bool`

子图间是否共享y轴

## shape参数

`hpsans` : `tuple`(`y0`,`y1`) or `list`(`tuple`)

填充水平 (平行于y轴) 区域, `y0`,`y1`表示区域的起始和结束纵坐标。可以用`list`传入多个

`vspans` : `tuple`(`x0`,`x1`) or `list`(`tuple`)

填充竖直 (平行于x轴) 区域, `y0`,`y1`表示区域的起始和结束纵坐标。可以用`list`传入多个

`shapes` : `dict` or `list`(`dict`)

plotly中的shape属性字典. 具体参照

<https://plotly.com/python/shapes/#vertical-and-horizontal-lines-positioned-relative-to-the-axis-data>

eg. `{"type": "rect", "x0": 1, "y0": 1, "x1": 2, "y1": 3}`则绘制一个长方形框

## 轴参数

`xrange : [lower_bound,upper_bound]`

指定x轴取值范围

`yrange : [lower_bound,upper_bound]`

指定y轴取值范围

`zrange : [lower_bound,upper_bound]`

指定z轴取值范围

## slider参数

`rangeslider : bool or dict`

`bool`: 默认False, 如果为True, 将展示默认的range slider

`dict`: 具体参照plotly文档: <https://plotly.com/python/range-slider/>

eg. `{'bgcolor': ('blue', .3), 'autorange': True}`

展示range slider

## 注解参数

`fontcolor : str`

注解文字颜色

`fontsize : int`

注解文字大小

`textangle : int=0`

注解文字方向

完成注解相关配置参考: <https://plot.ly/python/reference/#layout-annotations>