数据可视化

数据没有经过处理，那么他就只是一堆数据。

如果可能够将数据进行可视化操作，那数据它就可以很轻松的说明问题啦。

0. 准备工作

绘图工具：

基于`Python`

`pyecharts`，这里主要使用`pyecharts`去一个简单的介绍。

`matplotlib`，底层，学习需要一定成本

`seaborn`，对`matplotlib`的一个封装。

---

pyecharts官方文档：http://gallery.pyecharts.org/#/

0.1 模块安装

> pip install `pyecharts`
>
> pip install jupyter notebook
>
> pip install jupyter notebook==6.1.0 版本控制
>
> 运行采用：jupyter notebook

0.2 数据获取

- 八仙过海，各显神通
- 视频中示例的数据我会提供给到你。

1. 数据预处理（数据清洗

主要使用`pandas`模块，

1.导入模块：pip install pandas

pip install openpyxl

2.路径写入：jupyter notebook

3.新建一个文件

4.编写代码

清理空值

去除重复项

将数据处理一致等，

以下两篇文章是我在`CSDN`写的博文，对于简单的数据清洗，不妨一看。

遇到“脏乱差”的Excel数据怎么办？？利用Python规范Excel表格数据（数据清洗）

【数据分析】Python分析淘宝4200款Bra，发现最好卖的款式居然是。。。

导入模块：

```
import pandas as pd

# 打开文档
df = pd.read_excel('taobao_goods.xlsx')
```

删除重复的行：

```
# 删除行完全一样的值
df.drop_duplicates(inplace=True)
# 删除列重复的值
df.drop_duplicates(subset=['列名','列名'])
```

对地理位置进行处理：

```
location_list = []
for location in df['location']:
    location = location.split(' ')[0]
    location_list.append(location)
df['location'] = location_list
```

对销售量进行处理：

```
sales_list = []
for sale in df['sales']:
    sale = sale[:-3].replace('+', '')
    if '万' in sale:
        sale = int(float(sale.replace('万', '')) * 10000)
    sales_list.append(sale)

df['sales'] = sales_list
```

保存为新的表格：

```
df.to_excel('new_taobao_goods.xlsx',index=None)
```

2. 制作图表

导入模块

```
import jieba
import pandas as pd

from pyecharts import options as opts
from pyecharts.globals import ThemeType
from pyecharts.globals import SymbolType
from pyecharts.charts import Pie, Bar, Map, WordCloud, Page
```

备注：

安装：pip install jieba

2.1 词云

两种方法：

1. `pyecharts` 自带的生成词云
2. `wordcloud` 模块生成词云（推荐

方法一：

```
stop_words_txt = 'stop_words.txt'
# 载入停用词,即过滤词
jieba.analyse.set_stop_words(stop_words_txt)
# TextRank 关键词抽取，只获取固定词性
# topK为返回权重最大的关键词，默认值为20
# withWeight为返回权重值，默认为False
keywords_count_list = jieba.analyse.textrank(' '.join(df1.comment), topK=100,
withWeight=True)
print(keywords_count_list)
```

```
word_cloud = (
    WordCloud()
        .add("", keywords_count_list, word_size_range=[5, 50],
             shape=SymbolType.TRIANGLE,
            )
        .set_global_opts(title_opts=opts.TitleOpts(title="这里输入标题"))
)
# 这句话是渲染成一个html文件到当前文件夹下面
#     word_cloud.render('WordCloud.html')
```

方法二： （推荐，可自定义

> pip install `wordcloud`

```
import jieba
import numpy as np
import matplotlib.pyplot as plt

from PIL import Image
from wordcloud import WordCloud


# 打开文本
# text = open('1.txt',encoding='utf-8').read()

# 中文分词
text = ' '.join(jieba.cut(text))

# 生成对象
mask = np.array(Image.open("input_picture"))
wc = WordCloud(mask=mask,font_path='C:\Windows\Fonts\SimHei.ttf',mode='RGBA').generate(text)

# 显示词云
# plt.imshow(wc, interpolation='bilinear')
# plt.axis("off")
# plt.show()

# 保存到文件
wc.to_file('output_picture')
```

2.2 柱状图

一般柱状图：

```
bar = (
    Bar()
    .add_xaxis(Faker.days_attrs)
    .add_yaxis("商家A", Faker.days_values)
    .set_global_opts(
        title_opts=opts.TitleOpts(title="Bar-DataZoom（slider+inside）"),
    )
#      .render("bar_datazoom_both.html")
)
```

横向柱状图：

```
.reversal_axis()
.set_series_opts(label_opts=opts.LabelOpts(position="right"))
```

滑块柱状图：

```
datazoom_opts=[opts.DataZoomOpts()]
```

2.3 饼图

数据来自：`standard_goods_comments.xlsx`

这里用cup做展示

```
[('B', 1909), ('C', 810), ('A', 696), ('D', 259)]
```

多图显示cup：

```
from pyecharts import options as opts
from pyecharts.charts import Pie
from pyecharts.commons.utils import JsCode


fn = """
    function(params) {
        if(params.name == 'other')
            return '\\n\\n\\n' + params.name + ' : ' + params.value + '%';
        return params.name + ' : ' + params.value + '%';
    }
    """


def new_label_opts():
    return opts.LabelOpts(formatter=JsCode(fn), position="center")


pie = (
    Pie()
    .add(
        "",
        [['A_cup', round(696/total_cup, 2)*100],['other',round(1 -
696/total_cup, 2)*100]],
        center=["20%", "30%"],
        radius=[60, 80],
        label_opts=new_label_opts(),
    )
    .add(
        "",
        [['B_cup', round(1909/total_cup, 2)*100],['other',round(1 -
1909/total_cup, 2)*100]],
        center=["55%", "30%"],
        radius=[60, 80],
        label_opts=new_label_opts(),
    )
    .add(
        "",
        [['C_cup', round(810/total_cup, 2)*100],['other',round(1 -
810/total_cup, 2)*100]],
        center=["20%", "70%"],
        radius=[60, 80],
        label_opts=new_label_opts(),
    )
    .add(
        "",
        [['D_cup', round(259/total_cup * 100, 1)],['other',round(1 -
259/total_cup, 2)*100]],
        center=["55%", "70%"],
        radius=[60, 80],
        label_opts=new_label_opts(),
    )
    .set_global_opts(
        title_opts=opts.TitleOpts(title="Cup-多饼图"),
        legend_opts=opts.LegendOpts(
            type_="scroll", pos_top="20%", pos_left="80%", orient="vertical"
        ),
    )
#     .render("mutiple_pie.html")
)
```

疫情展示:

```
from pyecharts import options as opts
from pyecharts.charts import Pie
from pyecharts.faker import Faker


v = Faker.choose()
pie = (
    Pie()
    .add(
        "",
        [list(z) for z in zip(v, list(range(10,80,10)))],
        radius=["30%", "75%"],
        center=["25%", "50%"],
        rosetype="radius",
        label_opts=opts.LabelOpts(is_show=False),
    )
    .add(
        "",
        [list(z) for z in zip(v,list(range(10,80,10))[::-1])],
        radius=["30%", "75%"],
        center=["75%", "50%"],
        rosetype="area",
    )
    .set_global_opts(title_opts=opts.TitleOpts(title="Pie-玫瑰图示例"))
)
```

2.4 地图

```
from pyecharts import options as opts
from pyecharts.charts import Map
from pyecharts.faker import Faker

map = (
    Map()
    .add("店铺数量",[['广东',100],['广西',100],['湖南',19,]], "china")
    .set_global_opts(
        title_opts=opts.TitleOpts(title="商家店铺地址分布图"),
        visualmap_opts=opts.VisualMapOpts(max_=200),
    )
)
```

2.5 水球图

天气:

```
from pyecharts import options as opts
from pyecharts.charts import Liquid

liquid = (
    Liquid()
    .add("lq", [0.45,0.5])
    # 第一个值为显示的值，第二个值为水的分量
    .set_global_opts(title_opts=opts.TitleOpts(title="今日湿度"))
#     .render("liquid_base.html")
)
```

3. 整合图表

[多图表整合]()

```
Page.save_resize_html('page_draggable_layout.html',cfg_file=
'chart_config.json')
```