空气质量探索性分析报告

1. 业务背景

数据集是包含了北京 2013.12.02-2020.06.05 间每天的天气情况和空气质量等信息。
空气质量数据通常用于监测和评估城市或地区的空气质量，以确保环境保护和公共健康。 政府和相关机构依赖于这些数据来制定相关政策和法规，以减少污染和改善空气质量。 此外，空气质量直接关系到市民的健康，因此数据的透明性和可理解性对市民至关重要。

2. 分析目的

通过数据可视化来显示数据之间的关联，从而对数据进行处理。

1. **异常检测：** EDA可以帮助识别空气质量数据中的异常值，这些异常可能是设备故障、异常天气或污染事件的结果。
2. **模式识别：** EDA有助于识别季节性、周期性和趋势性模式，以了解不同时间段的空气质量变化。
3. **数据完整性：** 可以检查数据的缺失情况，以确定数据的完整性并采取适当的措施来填充缺失值。
4. **数据可视化：** 数据可视化可以将复杂的数据转化为易于理解的图形，有助于传达信息。

3. 分析过程与结果

1. 数据集介绍

属性	含义
日期	记录相关日期
AQI	AQI(Air Quality Index)，空气质量指数，描述了空气清洁或者污染的程度，以及对健康的影响
质量等级	根据 AQI 将空气质量等级划分为六个等级
PM2.5	直径小于或等于 2.5μm 的尘埃或飘尘在环境空气中的浓度 数值单位：$μg/m^3$
PM10	直径小于或等于 10.0μm 的尘埃或飘尘在环境空气中的浓度，数值单位：$μg/m^3$
SO2	二氧化硫，大气的主要污染物之一，数值单位：$μg/m^3$
CO	一氧化碳，大气的主要污染物之一，数值单位：mg/m^3
NO2	二氧化氮，大气的主要污染物之一，数值单位：$μg/m^3$
O3_8h	臭氧的 8 小时滑动平均值，数值单位：$μg/m^3$
天气状况	根据天气情况分为五种
气温	指在野外空气流通、不受太阳直射下测得的空气温度（一般在百叶箱内测定）
风力风向	风吹来的大小和方向

如下所示，导入所需库并加载数据集

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
# 以日期为索引
df = pd.read_csv('北京空气质量及天气情况缺失版.csv', index_col=0)
df.index = pd.to_datetime(df.index)
print(df)
```

	AQI	质量等级	PM2.5	PM10	SO2	CO	NO2	O3_8h	天气状况	气温	风力风向
日期											
2013-12-02	142	轻度污染	109	138	61	2.6	88	11	多云/多云	11℃/-1℃	无持续风向≤3级/无持续风向≤3级
2013-12-03	86	良	64	86	38	1.6	54	45	晴/晴	14℃/-1℃	无持续风向≤3级/无持续风向≤3级
2013-12-04	109	轻度污染	82	101	42	2.0	62	23	多云/多云	12℃/0℃	无持续风向≤3级/无持续风向≤3级
2013-12-05	56	良	39	56	30	1.2	38	52	晴/晴	12℃/-3℃	无持续风向≤3级/无持续风向≤3级

	AQI	质量等级	PM2.5	PM10	SO2	CO	NO2	O3_8h	天气状况	气温	风力风向
日期											
2013-12-06	169	中度污染	128	162	48	2.5	78	15	晴/霾	11℃/-2℃	无持续风向≤3级/无持续风向≤3级
...	...	...	...	...	...	...	...	...	...	...	...
2020-06-01	110	轻度污染	19	56	3	0.4	29	170	雷阵雨/多云	30℃/16℃	西南风4-5级/西南风4-5级
2020-06-02	99	良	23	74	2	0.3	25	158	晴/晴	32℃/21℃	南风3-4级/南风3-4级
2020-06-03	134	轻度污染	42	218	2	0.3	32	141	浮尘/晴	34℃/19℃	北风3-4级/北风3-4级
2020-06-04	68	良	8	42	3	0.3	20	121	多云/多云	30℃/17℃	东南风1-2级/东南风1-2级
2020-06-05	101	轻度污染	12	38	4	0.5	18	161	晴/晴	30℃/17℃	东南风1-2级/东南风1-2级

2376 rows × 11 columns

2. 查看缺失值

```
# 查看缺失值
k1 = df.isnull().sum()
k1.sort_values(ascending=False, inplace=True)
print(k1)
```

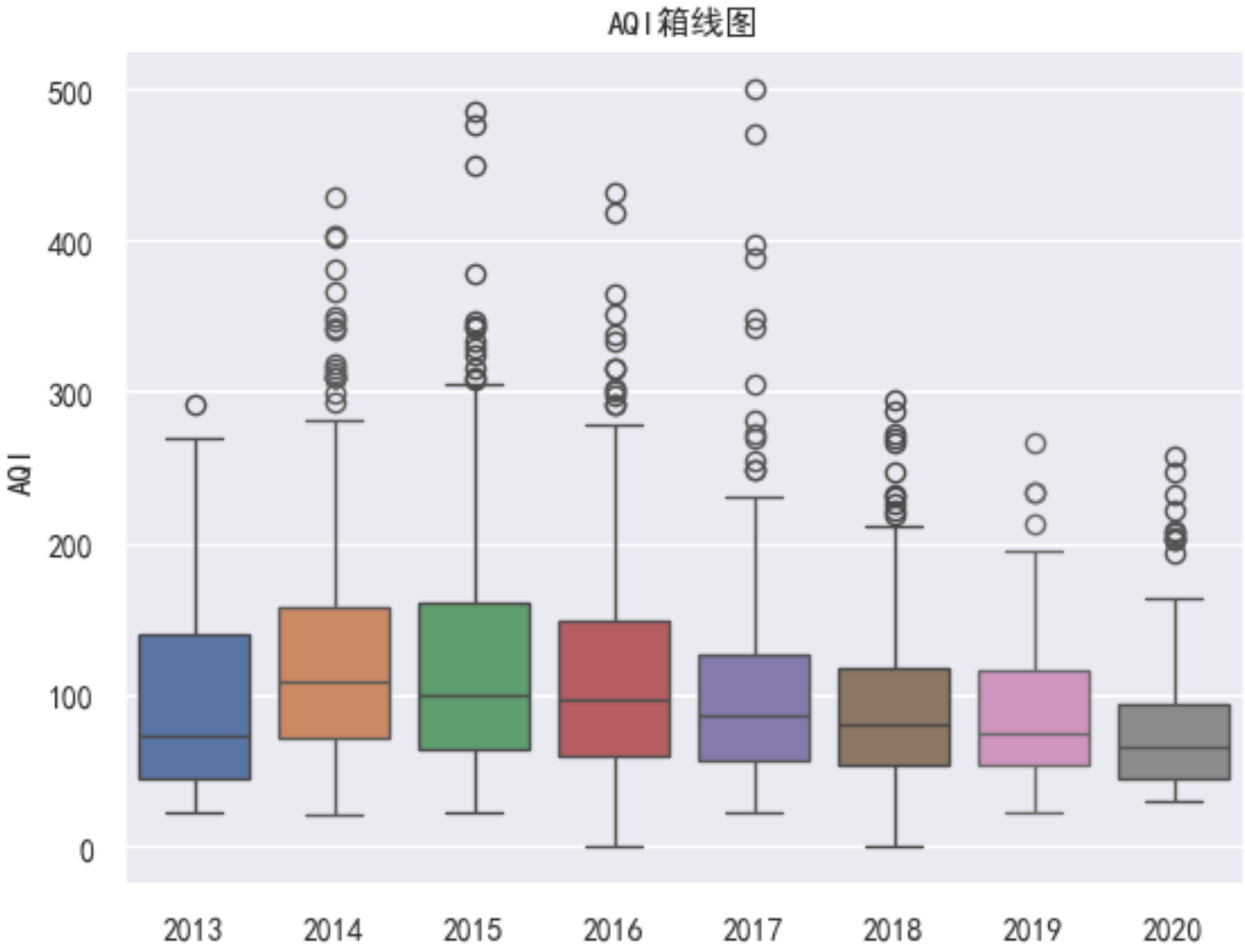
```
AQI       0
质量等级      0
PM2.5     0
PM10      0
SO2       0
CO        0
NO2       0
O3_8h     0
天气状况      0
气温        0
风力风向      0
dtype: int64
```

可以看到并没有缺失值，所有数据均完整。

3. 可视化分析

绘制AQI箱线图

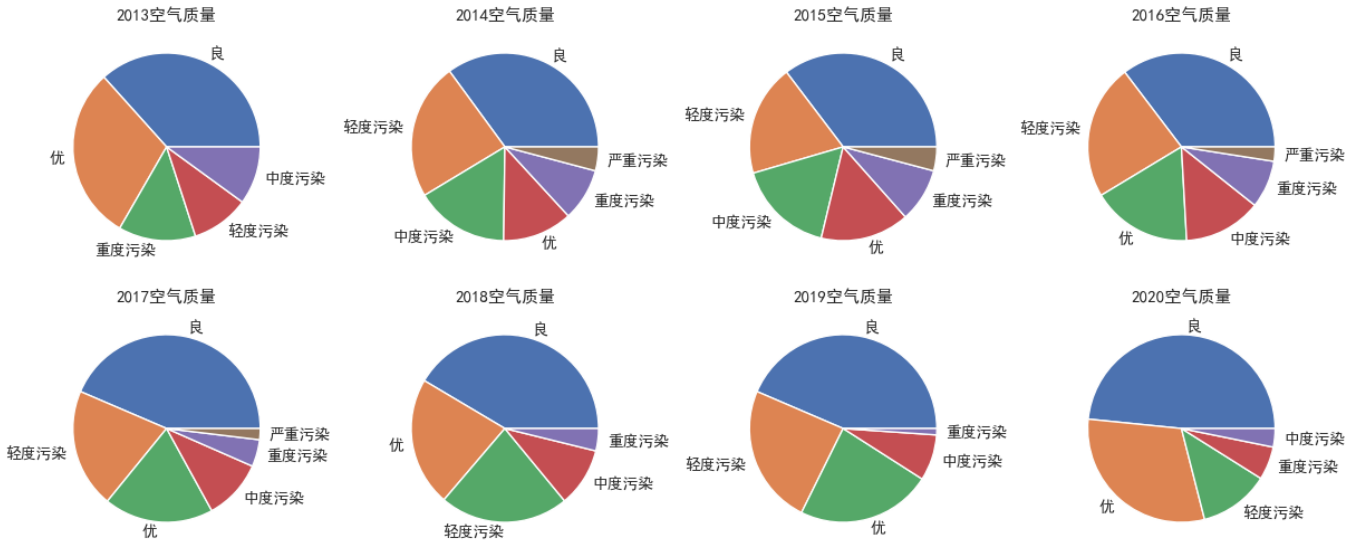
```
rc = {'font.sans-serif': 'SimHei',
      'axes.unicode_minus': False}
sns.set(font_scale=0.9,rc=rc)

plt.title('AQI箱线图')
sns.boxplot(x=2013, y=df['2013':'2013']['AQI'])
sns.boxplot(x=2014, y=df['2014':'2014']['AQI'])
sns.boxplot(x=2015, y=df['2015':'2015']['AQI'])
sns.boxplot(x=2016, y=df['2016':'2016']['AQI'])
sns.boxplot(x=2017, y=df['2017':'2017']['AQI'])
sns.boxplot(x=2018, y=df['2018':'2018']['AQI'])
sns.boxplot(x=2019, y=df['2019':'2019']['AQI'])
sns.boxplot(x=2020, y=df['2020':'2020']['AQI'])
plt.show()
```



可以发现平均AQI除2013-2014有所上升以外，其余年份均逐步下降。
说明环境保护工作卓有成效，空气质量年年提升。
由于极端天气，AQI指数极高的情况确实存在，因此无法判断箱线图中离群点是否为异常值。仅视为空气质量极差的天气。 可以看到，自2014年开始，空气质量极差的情况逐渐减少，到2018年以后，最高AQI不超过300.

```
# 空气质量等级分布图
plt.figure(figsize=(15, 6))
plt.subplot(241, title='2013空气质量')
plt.pie(df['2013':'2013']['质量等级'].value_counts(), labels=df['2013':'2013']['质
量等级'].value_counts().index)
plt.subplot(242, title='2014空气质量')
plt.pie(df['2014':'2014']['质量等级'].value_counts(), labels=df['2014':'2014']['质
量等级'].value_counts().index)
plt.subplot(243, title='2015空气质量')
plt.pie(df['2015':'2015']['质量等级'].value_counts(), labels=df['2015':'2015']['质
量等级'].value_counts().index)
plt.subplot(244, title='2016空气质量')
plt.pie(df['2016':'2016']['质量等级'].value_counts(), labels=df['2016':'2016']['质
量等级'].value_counts().index)
plt.subplot(245, title='2017空气质量')
plt.pie(df['2017':'2017']['质量等级'].value_counts(), labels=df['2017':'2017']['质
量等级'].value_counts().index)
plt.subplot(246, title='2018空气质量')
plt.pie(df['2018':'2018']['质量等级'].value_counts(), labels=df['2018':'2018']['质
```

```
量等级'].value_counts().index)
plt.subplot(247, title='2019空气质量')
plt.pie(df['2019':'2019']['质量等级'].value_counts(), labels=df['2019':'2019']['质
量等级'].value_counts().index)
plt.subplot(248, title='2020空气质量')
plt.pie(df['2020':'2020']['质量等级'].value_counts(), labels=df['2020':'2020']['质
量等级'].value_counts().index)
plt.show()
```



同样的，污染天气所占比例逐年减少，2018年以后已不存在重度污染，空气质量优良占比超过$\frac{2}{3}$。

4. 数据处理

对气温数据做处理，将原本字符型的气温数据拆分成最高和最低气温，替换掉原本气温数据。并且计算平均气温。

```
# 将气温拆分成最高气温和最低气温
df['最高气温'] = df['气温'].apply(lambda x :int(x.split('/')[0][0:-1]))
df['最低气温'] = df['气温'].apply(lambda x :int(x.split('/')[1][0:-1]))
df['平均气温'] = (df['最高气温'] + df['最低气温']) / 2
df.drop(columns=['气温'], inplace=True)
```

将空气质量等级数据量化

```
# 将空气质量等级量化
set(df['质量等级'].values)
{'严重污染', '中度污染', '优', '良', '轻度污染', '重度污染'}
```

可以发现共有六种类别，依次将其按照污染程度划分成1-5之间的数字

```
air_quality = {'严重污染': 5, '中度污染': 3, '优': 0, '良': 1, '轻度污染': 2, '重度污染': 4}
df['质量等级'] = df['质量等级'].map(air_quality)
```

将天气状况和风力风向量化，对每种类别随机分配一个数字。

```
weather = {elem:index for index,elem in enumerate(set(df['天气状况']))}
df['天气状况'] = df['天气状况'].map(weather)

wind = {elem:index for index,elem in enumerate(set(df['风力风向']))}
df['风力风向'] = df['风力风向'].map(wind)
```

得到量化之后的数据集：

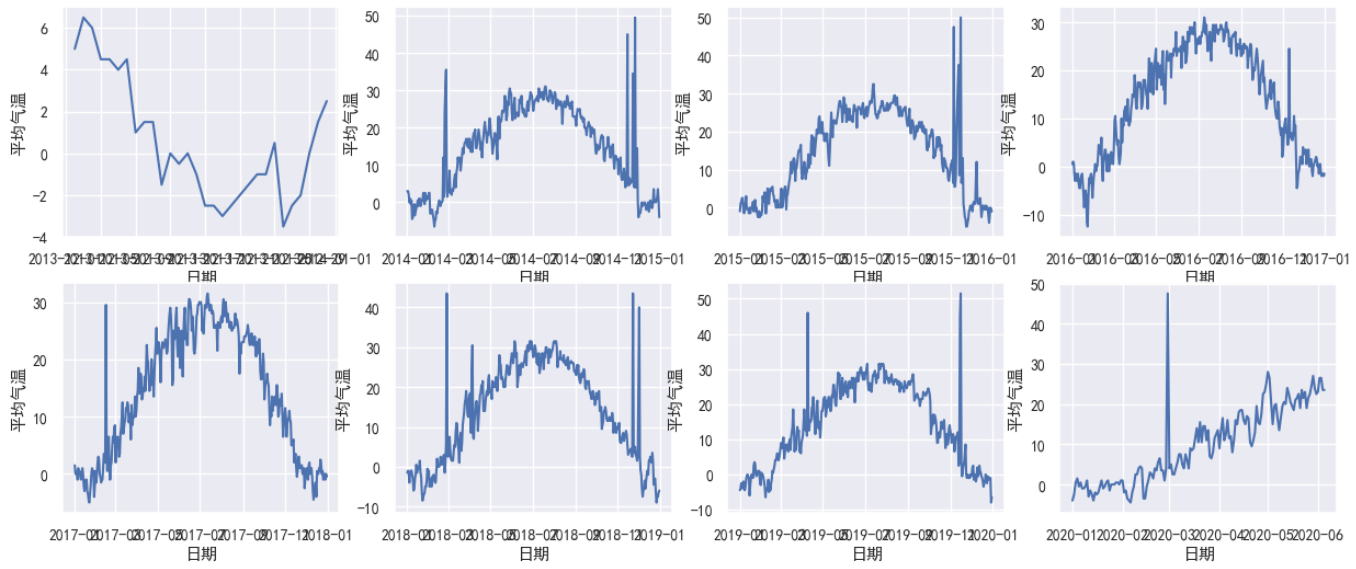
	AQI	质量等级	PM2.5	PM10	SO2	CO	NO2	O3_8h	天气状况	风力风向	最高气温	最低气温	平均气温
日期													
2013-12-02	142	2	109	138	61	2.6	88	11	1	120	11	-1	5.0
2013-12-03	86	1	64	86	38	1.6	54	45	0	120	14	-1	6.5
2013-12-04	109	2	82	101	42	2.0	62	23	1	120	12	0	6.0
2013-12-05	56	1	39	56	30	1.2	38	52	0	120	12	-3	4.5
2013-12-06	169	3	128	162	48	2.5	78	15	4	120	11	-2	4.5
...	...	...	...	...	...	...	...	...	...	...	...	...	...
2020-06-01	110	2	19	56	3	0.4	29	170	12	28	30	16	23.0
2020-06-02	99	1	23	74	2	0.3	25	158	0	17	32	21	26.5
2020-06-03	134	2	42	218	2	0.3	32	141	69	132	34	19	26.5
2020-06-04	68	1	8	42	3	0.3	20	121	1	144	30	17	23.5

	AQI	质量等级	PM2.5	PM10	SO2	CO	NO2	O3_8h	天气状况	风力风向	最高气温	最低气温	平均气温
日期													
2020-06-05	101	2	12	38	4	0.5	18	161	0	144	30	17	23.5

2376 rows × 13 columns

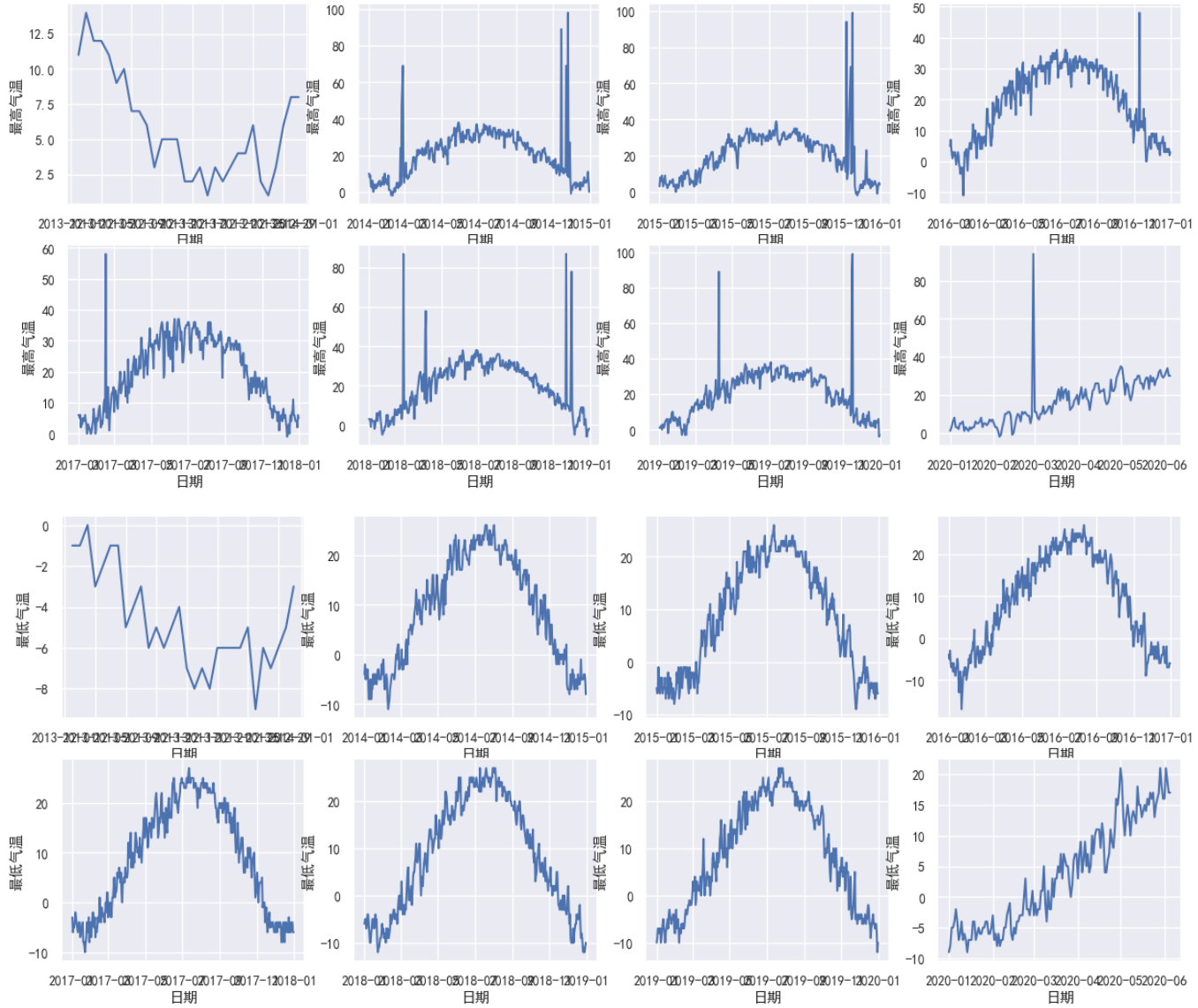
绘制各年份平均气温折线图

```
def draw_line_by_year(column='AQI'):
    plt.figure(figsize=(15, 6))
    plt.subplot(241)
    sns.lineplot(data=df['2013':'2013'][column])
    plt.subplot(242)
    sns.lineplot(data=df['2014':'2014'][column])
    plt.subplot(243)
    sns.lineplot(data=df['2015':'2015'][column])
    plt.subplot(244)
    sns.lineplot(data=df['2016':'2016'][column])
    plt.subplot(245)
    sns.lineplot(data=df['2017':'2017'][column])
    plt.subplot(246)
    sns.lineplot(data=df['2018':'2018'][column])
    plt.subplot(247)
    sns.lineplot(data=df['2019':'2019'][column])
    plt.subplot(248)
    sns.lineplot(data=df['2020':'2020'][column])
    plt.show()

draw_line_by_year('平均气温')
```



可以发现有明显的异常值，平均气温高达50摄氏度。
进一步绘制最高气温和最低气温折线图。

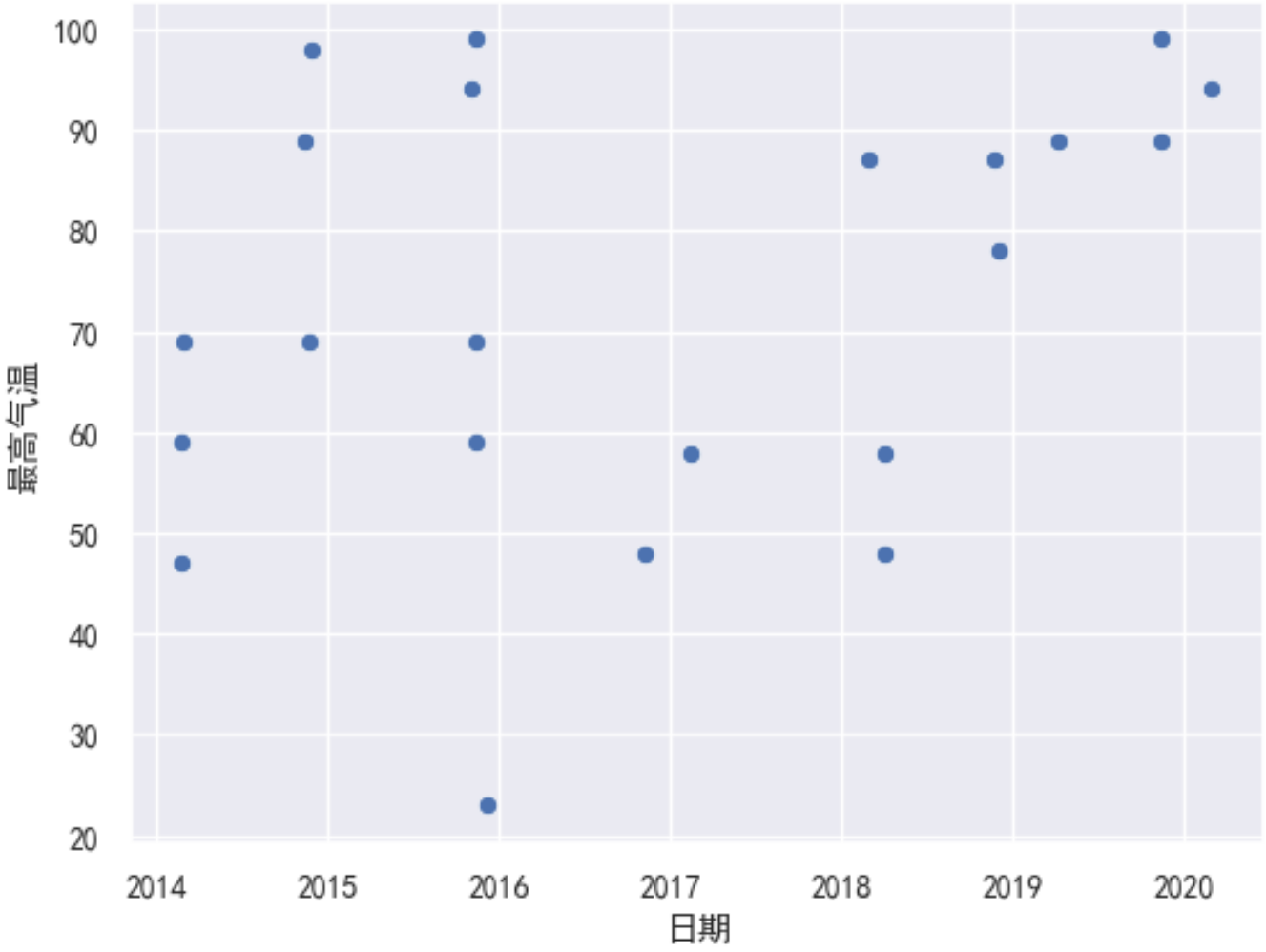
```
# 可以发现平均气温有明显异常值
# 绘制最低和最高气温进一步查看
# 最高气温折线图
draw_line_by_year('最高气温')
# 最低气温折线图
draw_line_by_year('最低气温')
```



发现最低气温正常，而最高气温有明显异常值。几乎每年的二月和十一月都有气温极高的异常出现。
寻找比附近15天平均气温高的日期，定为异常值，绘制散点图查看

```
# 绘图，可以发现最高气温有明显的的异常值，使用相邻日期气温替代

# window_size 是用于计算平均值的窗口大小
# 计算每个日期前后窗口大小内的平均值和中位数
rolling_average = df['最高气温'].rolling(window=15, min_periods=1,
center=True).mean()
```

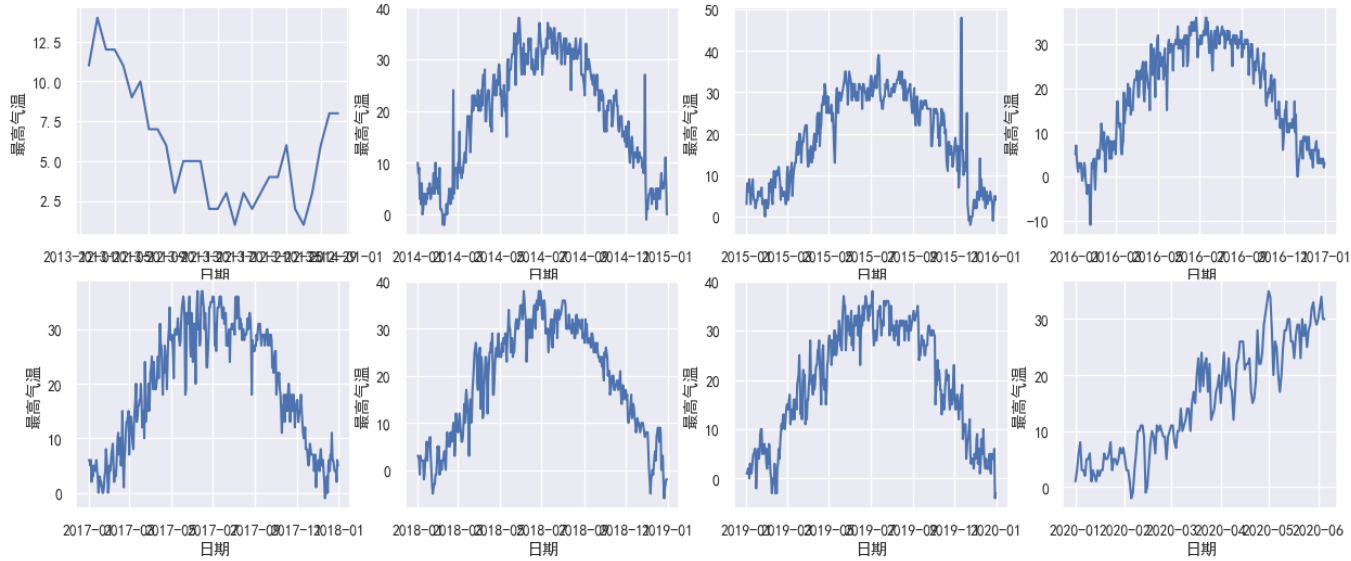
```
rolling_median = df['最高气温'].rolling(window=15, min_periods=1,
center=True).median()

# 找出异常值的索引
outliers = (df['最高气温'] - rolling_average) > 15

# 查看异常值，其中最低温度为2015年冬天，最高气温25度左右显然异常
sns.scatterplot(data=df.loc[outliers, '最高气温'])
plt.show()
```



发现其中最低温度为2015年冬天，最高气温25度左右显然异常，其余气温均超过40度，明显异常。

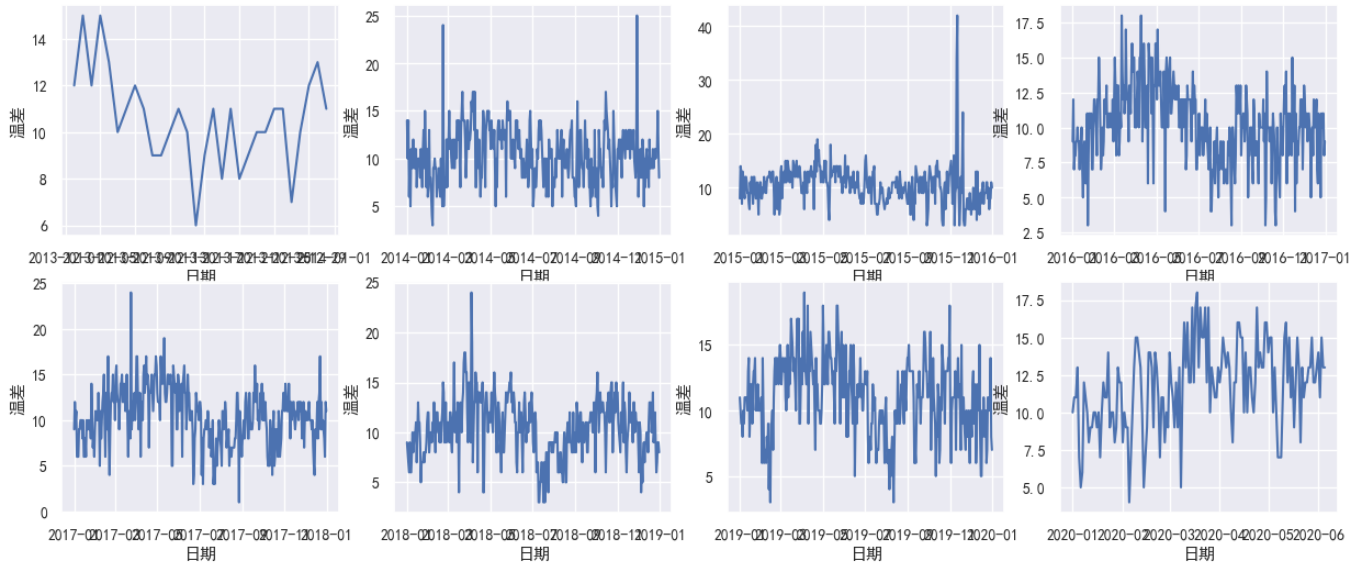
将异常值替换为相邻日期的中位数, 再次查看最高气温折线图

```
# 将异常值替换为相邻日期的中位数
df.loc[outliers, '最高气温'] = rolling_median[outliers].values
# 再次查看最高气温折线图
draw_line_by_year('最高气温')
```



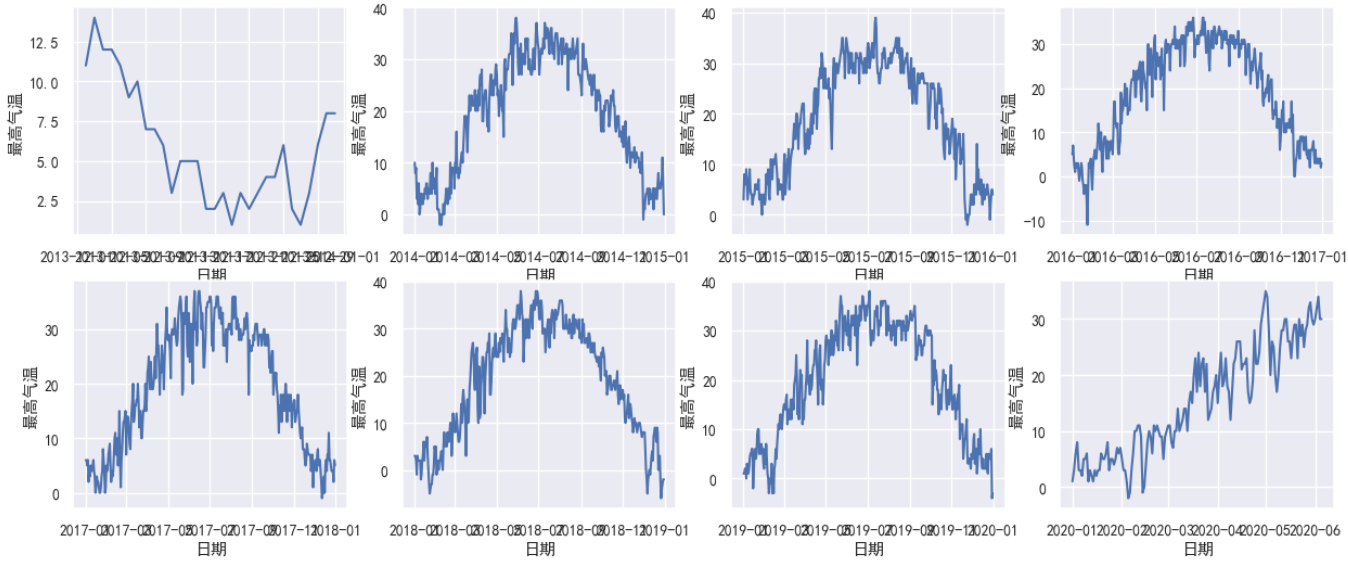
发现处理结果不太理想，对于2014及2015部分日期仍然有明显异常值,推测是很长时间（超过所设置15天窗口）的异常气温导致平均值和中位数都偏高, 替换过后的气温仍然属于异常范围。
改用温差检测异常值：

```
df['温差'] = df['最高气温'] - df['最低气温']
draw_line_by_year('温差')
```



可以看到，几乎全年的温差都在10度左右，超过20度的日期已经属于明显异常值。
将温差大于20的日期替换为最低气温+10，再次查看最高气温。

```
outliers = df['温差'] > 20
df.loc[outliers, '最高气温'] = df.loc[outliers, '最低气温'] + 10
draw_line_by_year('最高气温')
```

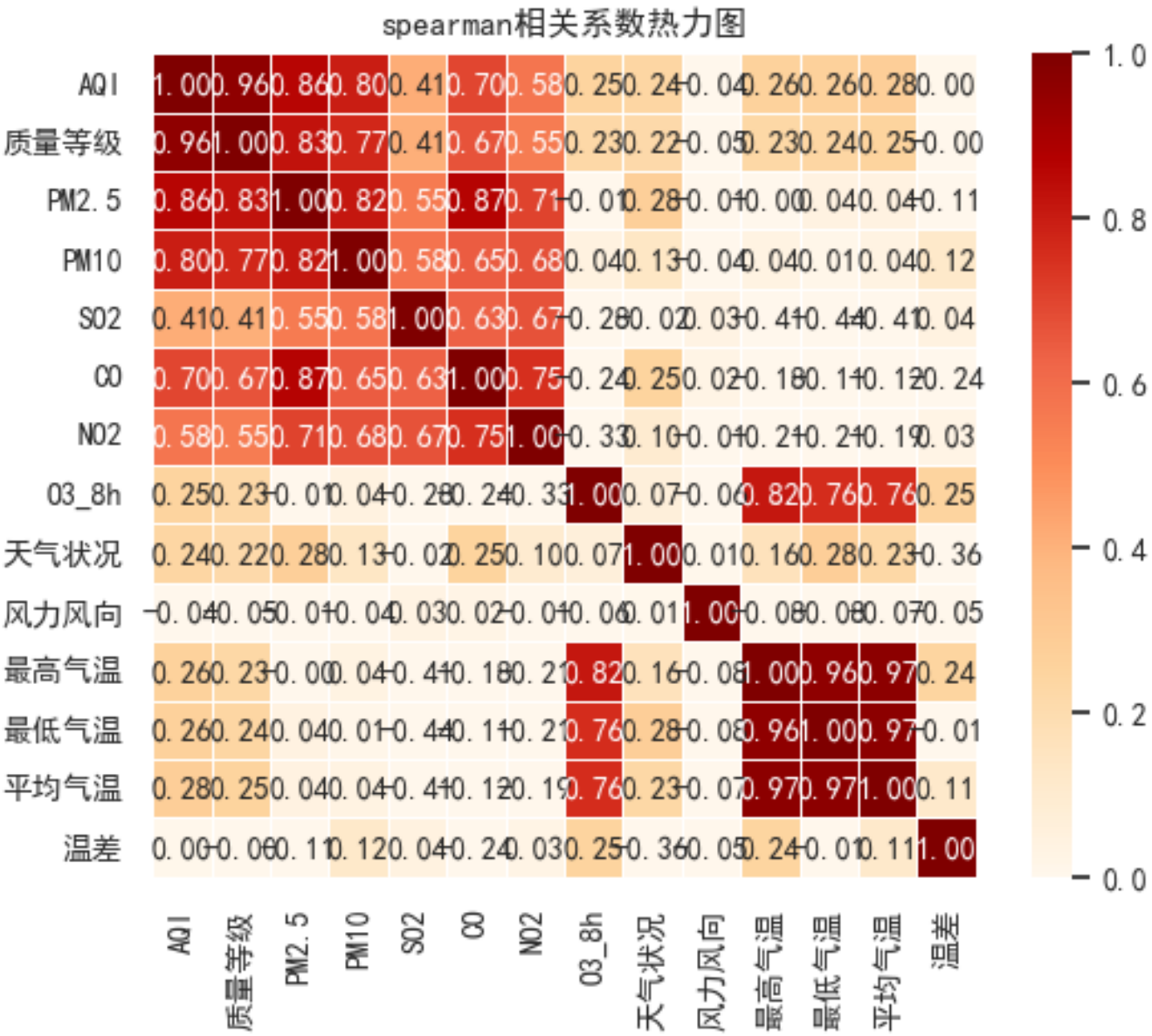


可以发现气温已经基本正常。

5. 相关性分析

- 绘制斯皮尔曼相关系数热力图

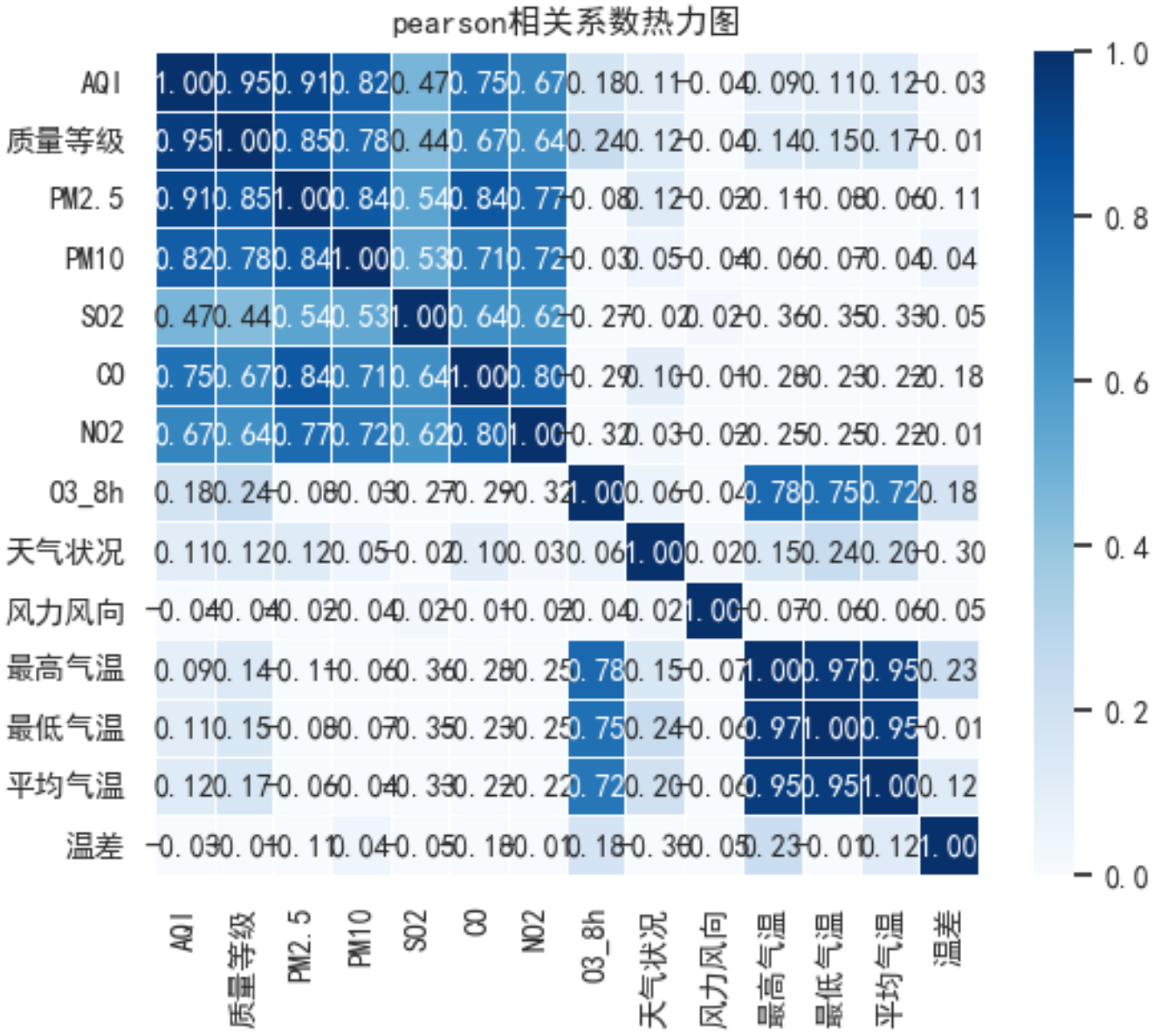
```
  # AQI与其他数据的spearman相关系数
sprm = df.corr(method='spearman')
# print(stmk.sort_values(ascending=False))

plt.title('spearman相关系数热力图')
sns.heatmap(sprm,
            annot=True,  # 显示相关系数的数据
            center=0.5,  # 居中
            fmt='.2f',  # 只显示两位小数
            vmin=0, vmax=1,  # 设置数值最小值和最大值
            xticklabels=True, yticklabels=True,  # 显示x轴和y轴
            square=True,  # 每个方格都是正方形
            cbar=True,  # 绘制颜色条
            linewidths=.5,
            cmap="OrRd",         #刻度颜色
            annot_kws={"size": 10}
            )
plt.show()
```



- 绘制皮尔逊相关系数热力图

```
# AQI与其他数据的pearson相关系数
prs = df.corr(method='pearson')
# print(stmk.sort_values(ascending=False))

plt.title('pearson相关系数热力图')
sns.heatmap(prs,
            annot=True,  # 显示相关系数的数据
            center=0.5,  # 居中
            fmt='.2f',  # 只显示两位小数
            vmin=0, vmax=1,  # 设置数值最小值和最大值
            xticklabels=True, yticklabels=True,  # 显示x轴和y轴
            square=True,  # 每个方格都是正方形
            cbar=True,  # 绘制颜色条
            linewidths=.5,
            cmap="Blues",         #刻度颜色
            annot_kws={"size": 10}
```

```
            )
plt.show()
```



观察两幅热力图，可以发现$(AQI, 质量等级, PM2.5, PM10, SO_2, CO, NO_2, O_38h)$，$(O_38h)$与$(最高气温, 最低气温, 平均气温)$之间相关系数较高。
实际上平均气温原本就是最高气温, 最低气温,的平均值，而最高气温和最低气温本就有很强的关联性。
而质量等级是根据AQI不同而划分等级得到的，而AQI又是通过$(PM2.5, PM10, SO_2, CO, NO_2, O_38h)$几个参数联合测定计算的。因此本该有极强的关联性。
相关系数显示臭氧的 8 小时滑动平均值与气温有很强的关联性，值得进一步分析探索。

4. 结论

分析过去几年内空气质量各项参数，可以发现北京市空气质量有明显改善，AQI值明显降低，质量等级明显提升。
虽然数据集是缺失版，但是实际发现并没有缺失值。AQI与空气中各种成分的浓度不好简单推断是否为异常值，需要进一步分析。而气温超过50摄氏度就可以确定为异常值，因此可以直观判断并且修正。
通过计算相关系数，发现臭氧与气温有很强的关联性，值得进一步挖掘分析。
此外天气状况与风力风向在数值化的过程中随机分配了数值。直观来看，天气状况为晴、风力较大的日期空气

应当较好，而天气状况为雾霾、浮尘的天气应当空气较差。因此根据天气状况的好坏，风力的大小顺序数值化天气状况与风力风向或许可以得到更强的关联性。