针对北京空气质量及天气情况的探索性分析报告
刘东洋 1120212831

1．业务背景分析

北京作为中国的首都及经济、政治和文化中心,空气质量一直是影响公众健康的重要问题。为监测和改进北京空气质量,北京环境监测中心从2008年开始在北京多个站点每天持续收集空气中的主要污染物浓度数据,这其中就包括我们分析的这一数据集。

本数据集中收集的指标直接反映空气质量的现状,如PM2.5、PM10和各气体污染物浓度,这些数据长期以来被广泛应用于评估空气质量、警报发布和政策评价等多个领域。同时,这些指标也与公众的生命和健康息息相关。监测数据及时公布,有利于政府采取相应应对措施,也便于公众掌握空气质量信息,在一定程度上保障公共健康。

与此同时,环境部门还通过将不同污染物浓度等级划分,制定了空气质量指数AQI,为大众提供一种更直观简单的判断空气质量优劣的度量值。这对促进空气治理意识的增长也有很重要的作用。

数据集由教师提供, 包含了北京 2013. 12.02-2020.06.05 间每天的天气情况和空气质量 。

以此数据为基础利用numpy,pandas,matplotlib 等工具进行可视化探索分析操作。

数据集共 2376 条记录，包含空气质量和天气情况两部分 12 个属性，如下表 2-所示：

表 2-数据属性信息

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

数据示例如下图 2-所示：

日期	AQI	质量等级	PM2.5	PM10	SO2	CO	NO2	O3_8h	天气状况	气温	风力风向
2013/12/2	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
2013/12/3	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
2013/12/4	109	轻度污染	82	101	42	2	62	23	多云/多云	12℃/0℃	无持续风向≤3级/无持续风向≤3级
2013/12/5	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
2013/12/6	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
2013/12/7	291	重度污染	241	285	64	4.2	98	6	霾/霾	9℃/-1℃	无持续风向≤3级/无持续风向≤3级
2013/12/8	223	重度污染	173	189	47	2.9	60	41	霾/晴	10℃/-1℃	北风4-5级/北风4-5级
2013/12/9	26	优	11	16	10	0.6	22	51	晴/晴	7℃/-5℃	北风3-4级/无持续风向≤3级
##########	45	优	21	45	14	1	29	52	多云/晴	7℃/-4℃	北风4-5级/北风4-5级
##########	30	优	19	30	15	0.7	30	45	晴/晴	6℃/-3℃	无持续风向≤3级/北风3-4级
##########	29	优	16	29	11	0.8	25	56	晴/晴	3℃/-6℃	北风4-5级/无持续风向≤3级
##########	66	良	48	63	29	1.3	45	29	晴/晴	5℃/-5℃	无持续风向≤3级/无持续风向≤3级
##########	56	良	40	48	29	1.2	41	46	晴/晴	5℃/-6℃	无持续风向≤3级/无持续风向≤3级
##########	64	良	46	55	31	1.5	49	31	晴/晴	5℃/-5℃	无持续风向≤3级/无持续风向≤3级
##########	134	轻度污染	102	126	59	2.5	70	10	多云/小雪	2℃/-4℃	无持续风向≤3级/无持续风向≤3级
##########	80	良	59	41	35	1.4	39	42	多云/晴	2℃/-7℃	无持续风向≤3级/无持续风向≤3级
##########	45	优	29	45	22	0.9	32	43	晴/晴	3℃/-8℃	无持续风向≤3级/无持续风向≤3级
##########	63	良	45	60	30	1.2	50	35	晴/晴	1℃/-7℃	无持续风向≤3级/无持续风向≤3级
##########	45	优	30	45	28	1.1	46	47	晴/晴	3℃/-8℃	无持续风向≤3级/无持续风向≤3级
##########	82	良	61	81	43	1.6	69	30	晴/多云	2℃/-6℃	无持续风向≤3级/无持续风向≤3级
##########	179	中度污染	135	178	67	2.8	95	14	晴/晴	3℃/-6℃	无持续风向≤3级/无持续风向≤3级
##########	166	中度污染	126	166	62	2.9	90	30	晴/晴	4℃/-6℃	无持续风向≤3级/无持续风向≤3级

图 2-：天气数据示例

2. 分析目的

本次对北京空气质量数据进行探索性分析,主要目的是为后续建立预测或分类模型做准备工作。

具体来说,我们期望通过探索性分析可以得到以下效益:

1.了解数据集的整体情况,如记录条数、属性类型和数量等基本特征。这有助于选择合适的数据处理方法。

2.检测和分析数据中的缺失情况。找出缺失程度高的属性,针对缺失进行补充或特征选择。

3.探索属性间的相关性。识别出高度相关的属性,为去除冗余特征奠定基础。

4.分析单变量和多变量之间的分布规律。 Revel异常值和异常样本,识别易受异常值影响的属性。

5.总结不同空气质量指标与气象因素之间的关联。为构建功能强的预测模型提供依据。

通过对数据集进行全面且系统的统计描述、可视化分析,我们期望找到其内在规律和特征,包括属性类型、缺失程度、分布情况、相关性等。这将为后续对数据的预处理(如插值、标准化等)及模型构建(如选择输入变量和算法)提供重要参考。从而能够建立一个性能更优的空气质量预测模型。

3. 可视化探索分析

（1）基本统计
导入数据并查看记录数和属性数量:

```
import pandas as pd
# 导入数据
df = pd.read_csv('air_quality_data.csv')
# 记录数和属性数量
print(df.shape)
print(df.columns)
```

输出结果：(2376, 12)
Index(['日期', 'AQI', '质量等级', 'PM2.5', 'PM10', 'SO2', 'CO', 'NO2', 'O3_8h',
'天气状况', '气温', '风力风向'],
dtype='object')

查看各属性的数据类型:

```
# 查看属性类型
print(df.info())
```

输出结果如下：

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2376 entries, 0 to 2375
Data columns (total 12 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   日期     2376 non-null   object
```

```
 1   AQI     2376 non-null   int64
 2   质量等级 2376 non-null  object
 3   PM2.5   2376 non-null   int64
 4   PM10    2376 non-null   int64
 5   SO2     2376 non-null   int64
 6   CO      2376 non-null   float64
 7   NO2     2376 non-null   int64
 8   O3_8h   2376 non-null   int64
 9   天气状况    2376 non-null   object
 10  气温      2376 non-null   object
 11  风力风向    2376 non-null   object
dtypes: float64(1), int64(6), object(5)
memory usage: 222.9+ KB
None
```

（2）缺失值分析

导入需要的库和计算缺失值比例:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns


# 计算每个属性的缺失值比例
missing_data = df.isnull().sum() / len(df)
```

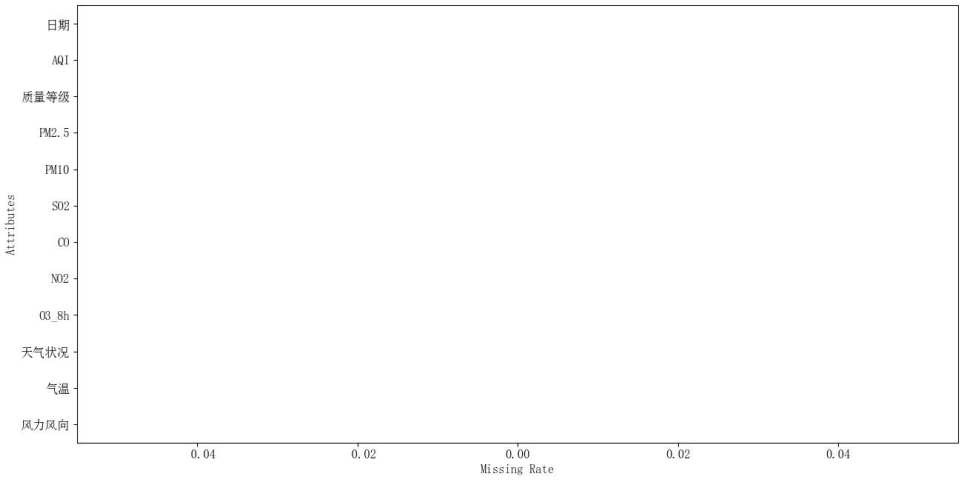
可视化各属性缺失值比例:

```
# 绘制缺失值比例条形图
sns.barplot(x=missing_data, y=missing_data.index)
plt.xlabel('Missing Rate')
plt.ylabel('Features')
plt.savefig('missing_rate.png')
```

结果：

```
日期        0.0
AQI       0.0
质量等级      0.0
PM2.5     0.0
PM10      0.0
SO2       0.0
CO        0.0
NO2       0.0
O3_8h     0.0
天气状况      0.0
气温        0.0
风力风向      0.0
dtype: float64
```

并未发现缺失值

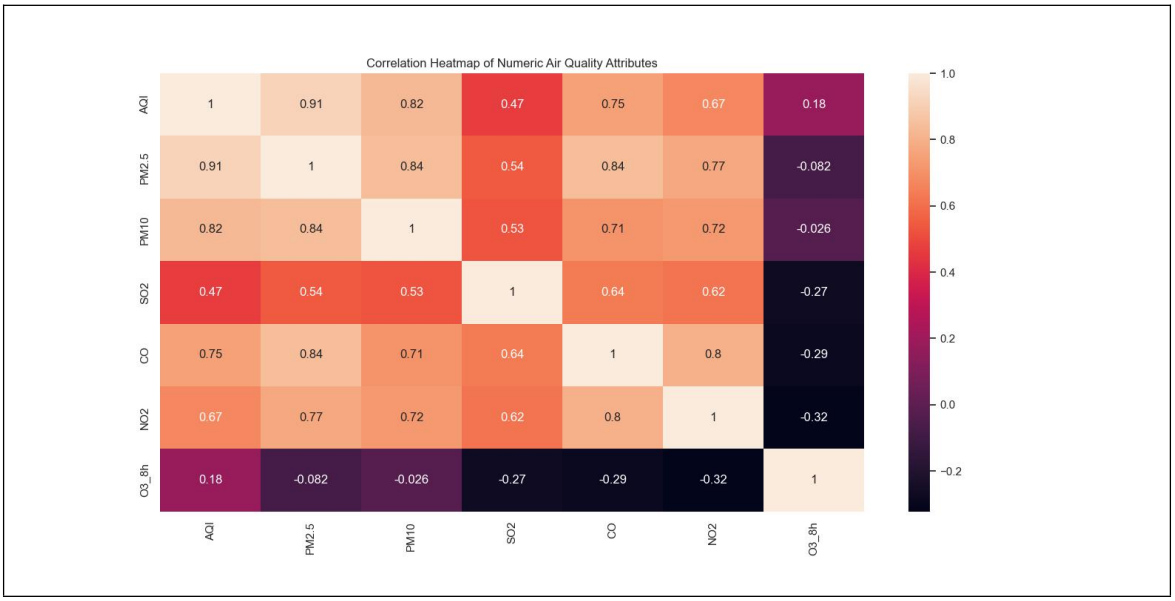


（3）相关性分析

数值空气质量属性间相关系数热图

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

#加载数据
df = pd.read_csv('air_quality_data.csv', encoding='utf-8')

#查看数据属性
print(df.info())

#选择需要分析的数值型属性
num_cols = ['AQI', 'PM2.5', 'PM10', 'SO2', 'CO', 'NO2', 'O3_8h']

#计算数值属性之间的相关系数
corr = df[num_cols].corr()

#绘制热图可视化相关系数
sns.set(font_scale=1)
ax = sns.heatmap(corr,annot=True,
                 xticklabels=corr.columns,
                 yticklabels=corr.columns)

#设置标题和标签
plt.title('Correlation Heatmap of Numeric Air Quality Attributes')
plt.xticks(rotation=90)

#显示热图
plt.show()
```

可视化效果如图 2-21 所示。



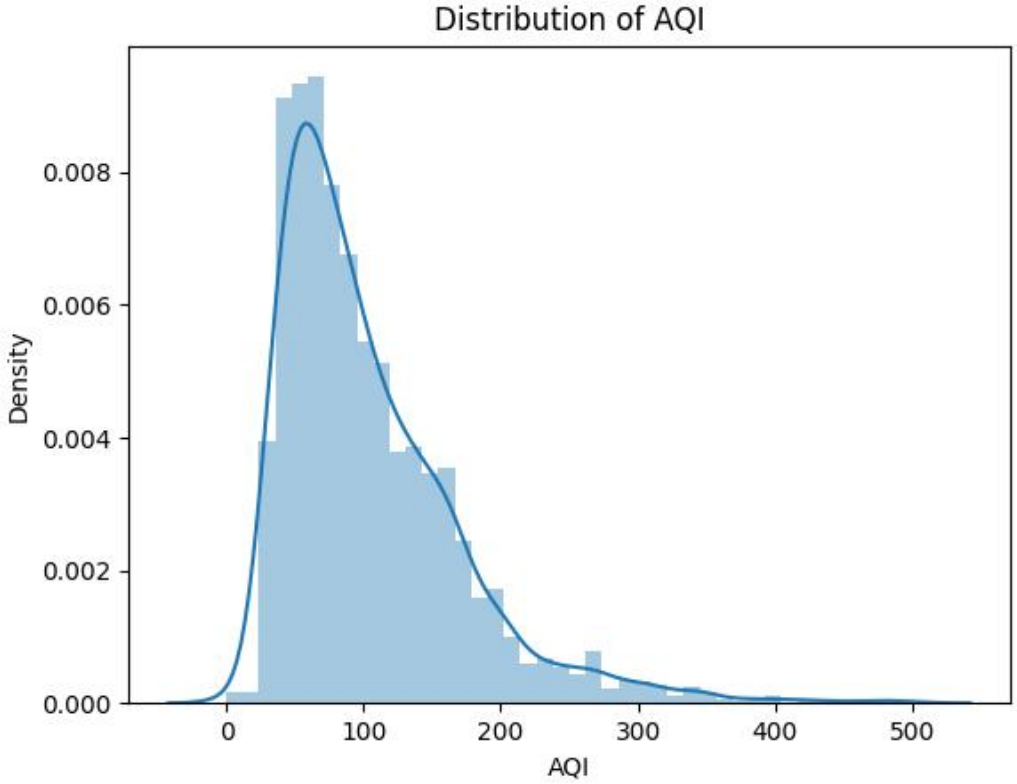
（4）单变量分布分析

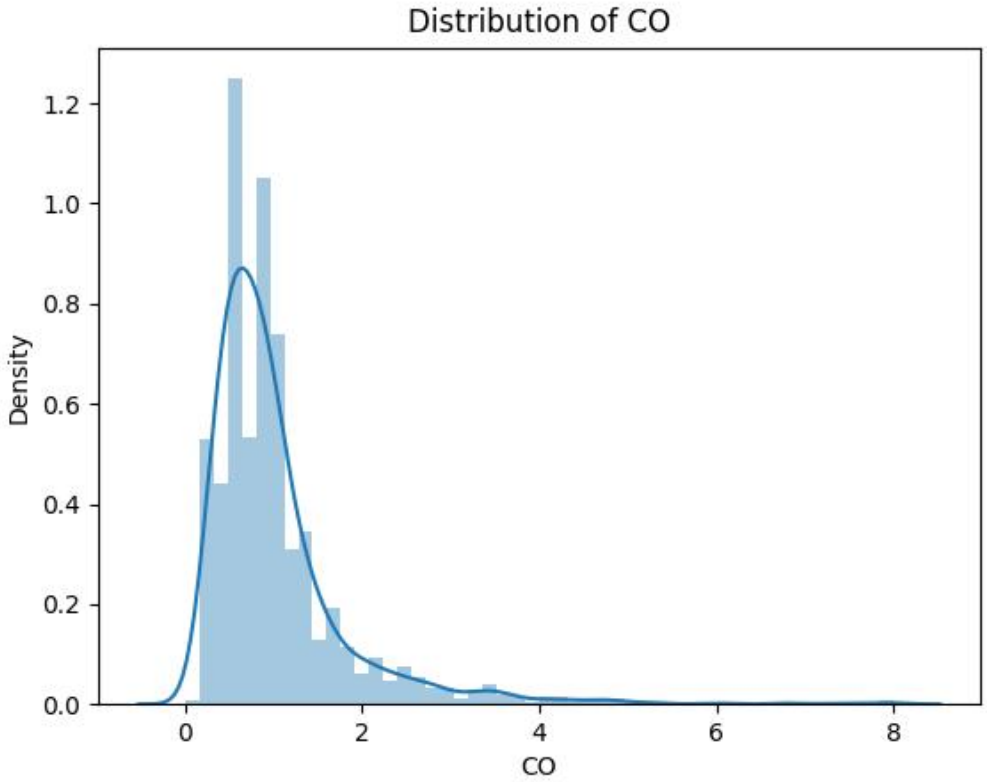
数值空气质量属性:

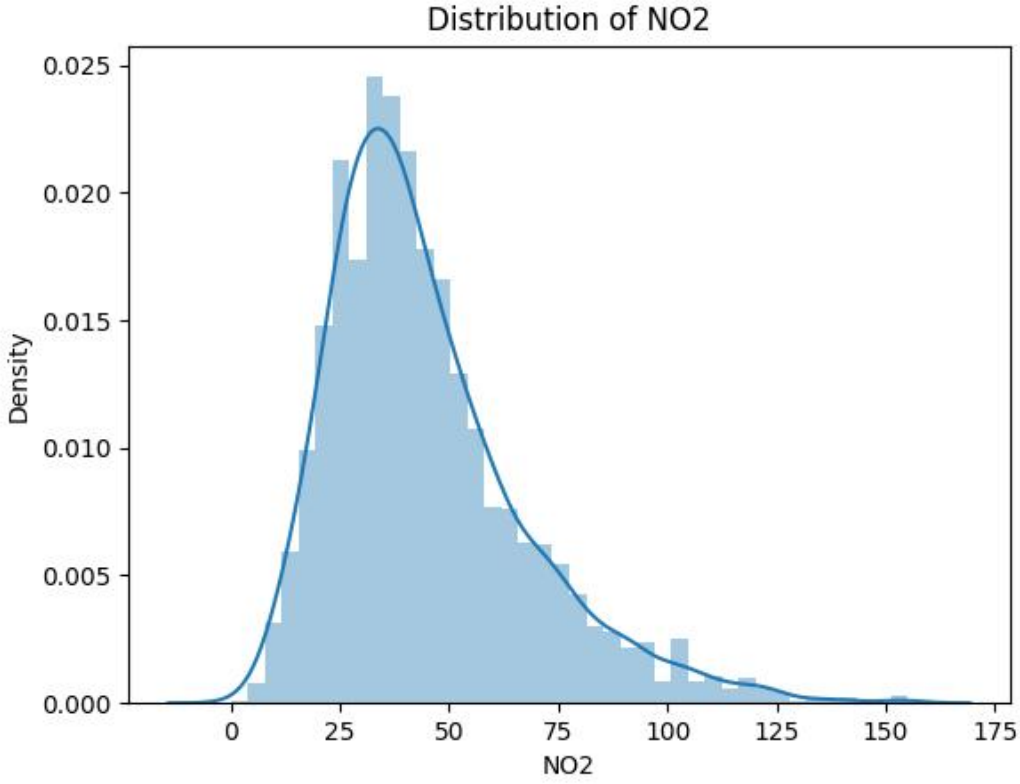
定义需要分析的数值型属性列表,使用循环绘制每个属性的分布直方图,设置每个图表标题代表对应的属性.使用Seaborn的distplot函数绘制每个属性的分布情况,每次绘制一个属性后展示图表。

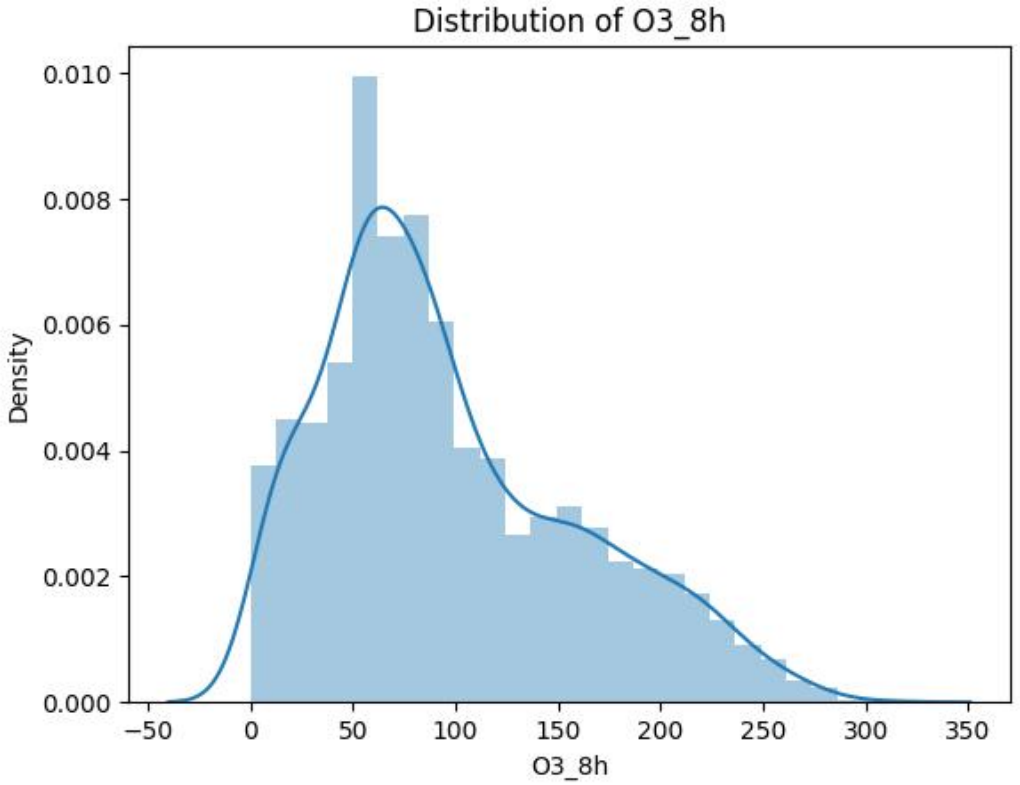
```
# 导入相关模块
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# 加载数据
df = pd.read_csv('北京空气质量及天气情况缺失版.csv')

# 选择数值型属性分析
num_cols = ['AQI', 'PM2.5', 'PM10', 'SO2', 'CO', 'NO2', 'O3_8h']

# 循环绘制每个属性的直方图
for col in num_cols:
    # 设置图表标题
    plt.title('Distribution of ' + col)

    # 绘制属性分布直方图
    sns.distplot(df[col])


    # 展示当前图表
    plt.show()
```

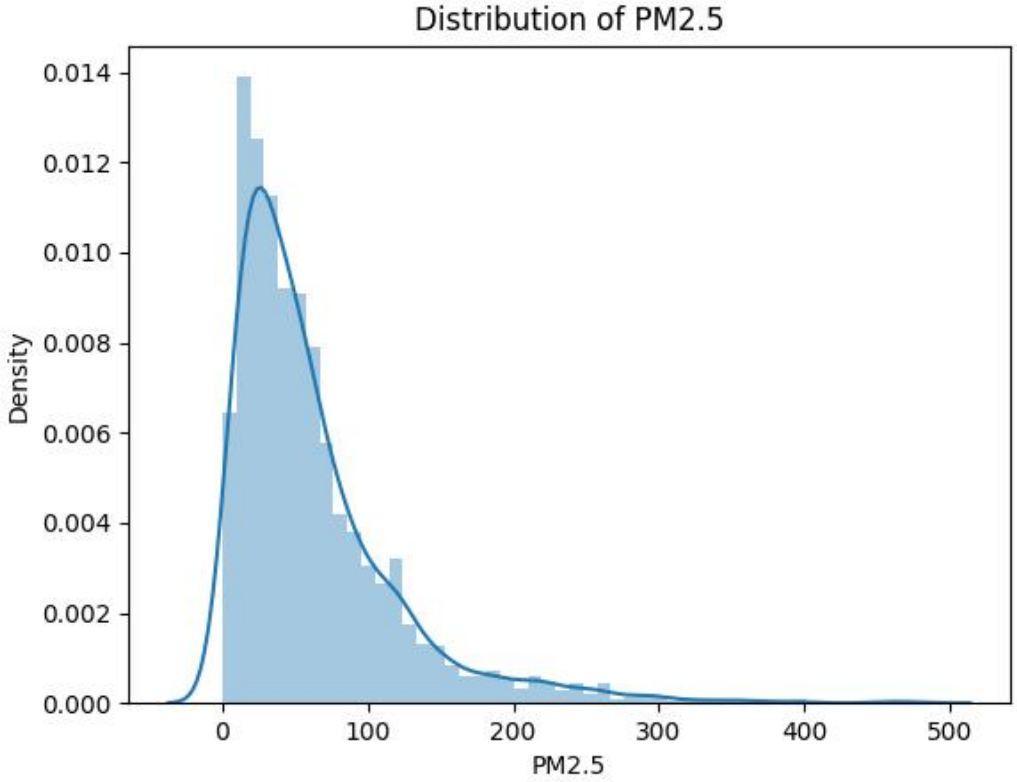
结果如下：

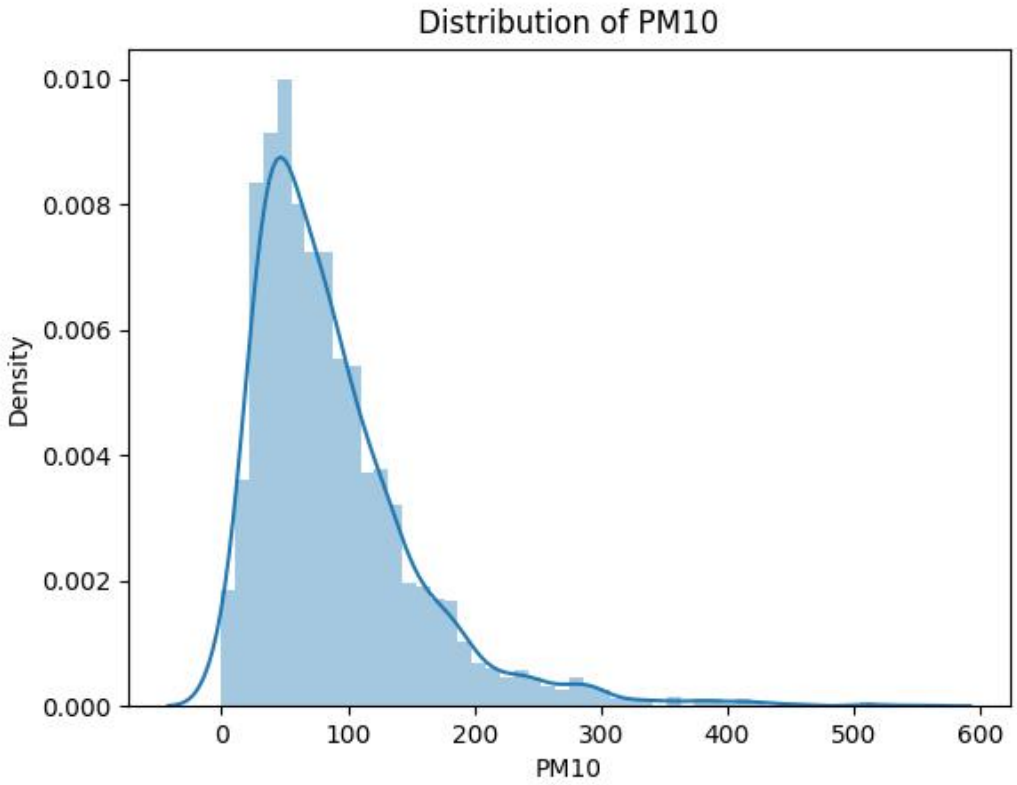

Distribution of AQI
Density
0.008
0.006
0.004
0.002
0.000
0
100
200
300
400
500
AQI

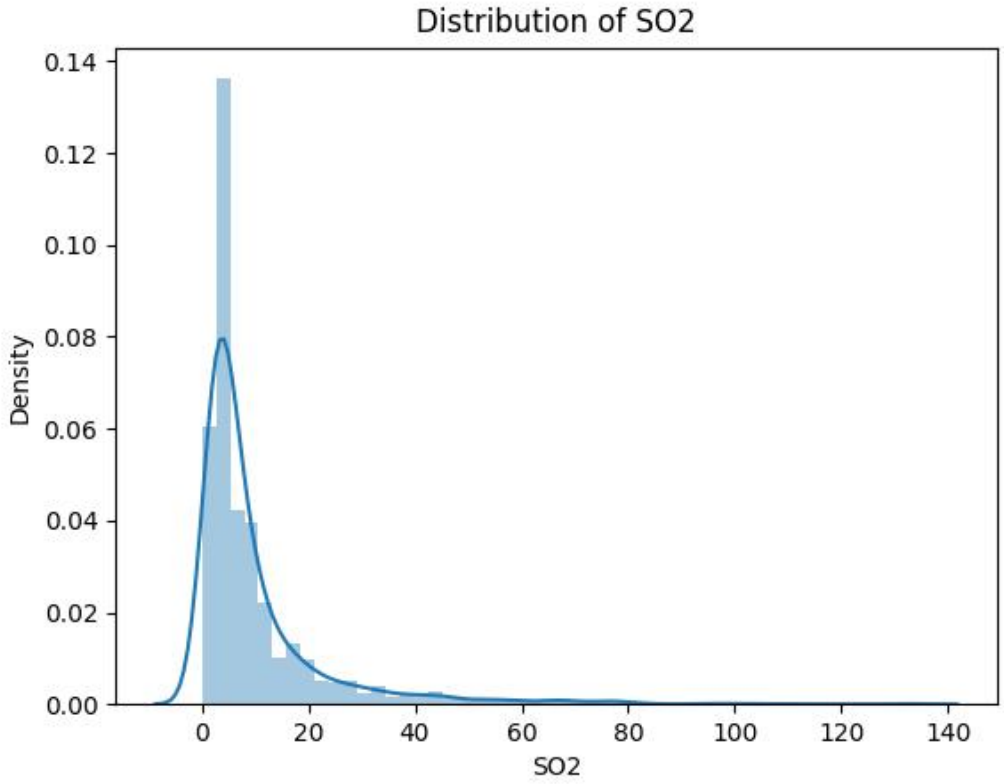

Distribution of CO
Density
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
CO


Distribution of NO2
Density
0.025
0.020
0.015
0.010
0.005
0.000
0
25
50
75
100
125
150
175
NO2


Distribution of O3_8h
Density
0.010
0.008
0.006
0.004
0.002
0.000
−50
0
50
100
150
200
250
300
350
O3_8h


Distribution of PM2.5
Density
0.014
0.012
0.010
0.008
0.006
0.004
0.002
0.000
0
100
200
300
400
500
PM2.5



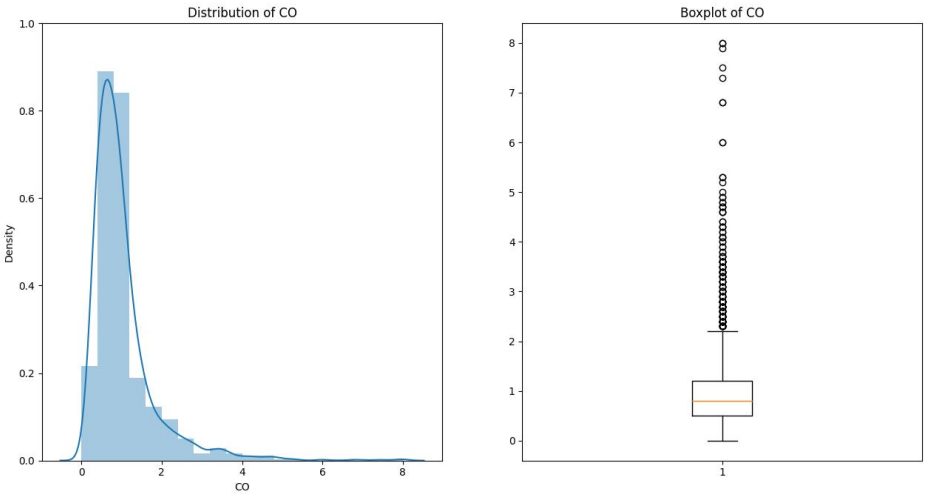


其中CO有一部分样本的频率大于1，显然不正确，因此对CO数据进行异常分析。

改进代码为：

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# 加载数据
df = pd.read_csv('北京空气质量及天气情况缺失版.csv')
```

```
# 定义属性列表
numeric_cols = ['AQI', 'PM2.5', 'PM10', 'SO2', 'NO2', 'O3_8h']
float_col = ['CO']

# 相关系数
corr = df[numeric_cols].corr()

# 单变量分布
for col in numeric_cols + float_col:

    plt.subplot(1, 2, 1)

    if col == 'CO':
        sns.distplot(df[col], bins=20)
        # 修改最大值为1
        plt.ylim(0, 1)
    else:
        sns.distplot(df[col])

    plt.title('Distribution of ' + col)

    # boxplot for CO
    if col == 'CO':
        plt.subplot(1, 2, 2)
        plt.boxplot(df[col])
        plt.title('Boxplot of CO')

    plt.show()
```



原来之前用的是numeric_cols，因为CO是浮点类型的数据，需要用float_col = ['CO']来定义属性列表。

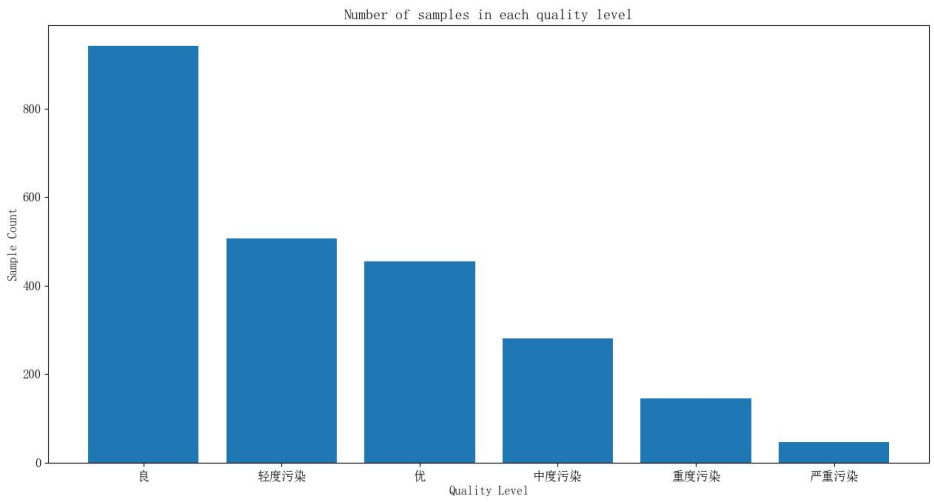
修改后直方图正确。但是从CO的箱线图中看出有较多超过上限的异常值。

非数值属性分析：

1.质量等级:

获取质量等级列,转换为分类型供计数分析,使用value_counts()函数统计各值出现频次,绘制垂直柱状图展示各级别样本量,xlabel和title添加注释.

```
import matplotlib.pyplot as plt
import pandas as pd

# 加载数据
df = pd.read_csv('北京空气质量及天气情况缺失版.csv')

# 将quality_level列转换为分类型
df['质量等级'] = df['质量等级'].astype('category')

# 计数各级别出现次数
level_count = df['质量等级'].value_counts()\

# 设置支持中文的字体
font_chinese = {'family': 'SimSun',
                'weight': 'bold',
                'size': 12}

plt.rc('font', **font_chinese)

# 绘制频数柱状图
plt.bar(level_count.index, level_count)
plt.title('Number of samples in each quality level')
plt.xlabel('Quality Level')
plt.ylabel('Sample Count')

plt.show()
```



可以看出空气质量等级的分布，良最多，轻度污染和优较多，中度污染及以上较少，整体空气质量较好。

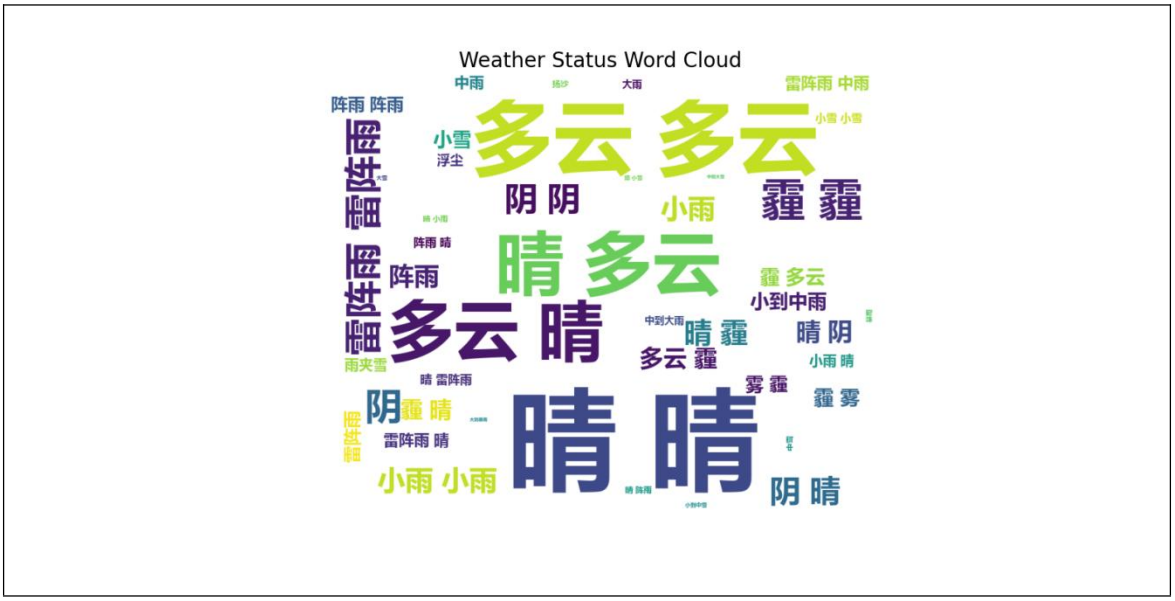
2.天气状况:

天气词云图：

通过join将列表拼成一个字符串输国词云绘制，设置字体、背景等词云参数，展示生成的词云图，title添加标题。

```
import pandas as pd
from wordcloud import WordCloud
import matplotlib.pyplot as plt

df = pd.read_csv('北京空气质量及天气情况缺失版.csv')

weather_list = df['天气状况'].tolist()

# 替换文本中所有'/'为'/#'
text = ' '.join([item.replace('/', '/#') for item in weather_list])

wordcloud = WordCloud(
    font_path='C:/Windows/Fonts/msyhbd.ttc',
    background_color='white',
    max_font_size=200,
    width=1000,
    height=800
).generate(text)

plt.imshow(wordcloud)
plt.axis('off')
plt.title('Weather Status Word Cloud', fontsize=20)

plt.show()
```

词云图结果：

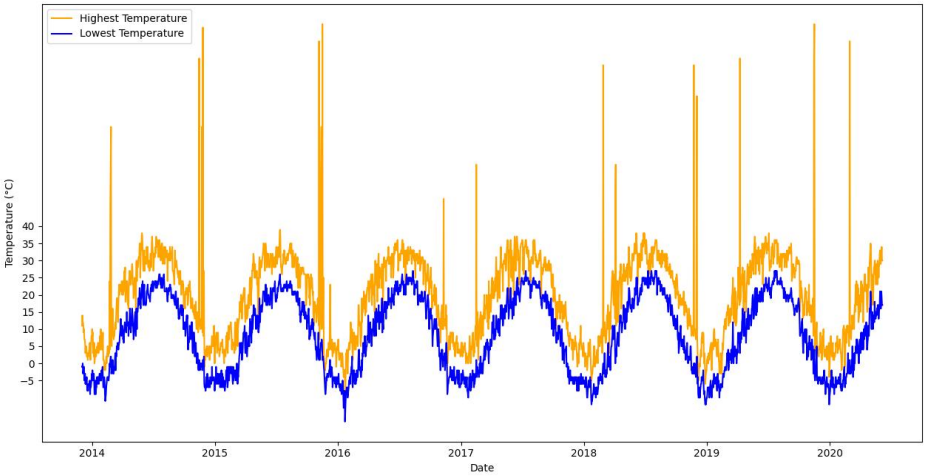


从词云图中可以看出，最多的天气状况是晴/晴和多云/多云。

3.气温:

将数据集中的“气温属性”拆分为最高气温“temp_max”和最低气温“temp_min”,绘制两种气温随着日期的变化图像。

```python
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

# 读取数据
df = pd.read_csv('北京空气质量及天气情况缺失版.csv')

# 拆分气温属性
df['气温'] = df['气温'].str.split('/')
print(df['气温'])

df['temp_max'] = df['气温'].str[0]
df['temp_min'] = df['气温'].str[1]
df['temp_min'] = df['temp_min'].str[:-1]
df['temp_max'] = df['temp_max'].str[:-1]
df['temp_max'] = df['temp_max'].astype(float)
df['temp_min'] = df['temp_min'].astype(float)

# 日期格式转换
df['日期'] = pd.to_datetime(df['日期'])


# 在同一张图中画出最高、最低温度变化曲线
plt.plot(df['日期'], df['temp_max'], '-', label="Highest Temperature")
plt.plot(df['日期'], df['temp_min'], '-', label="Lowest Temperature")
plt.legend()
plt.yticks(np.arange(-5, 45, 5))
plt.xlabel("Date")
plt.ylabel("Temperature (°C)")
plt.show()
```

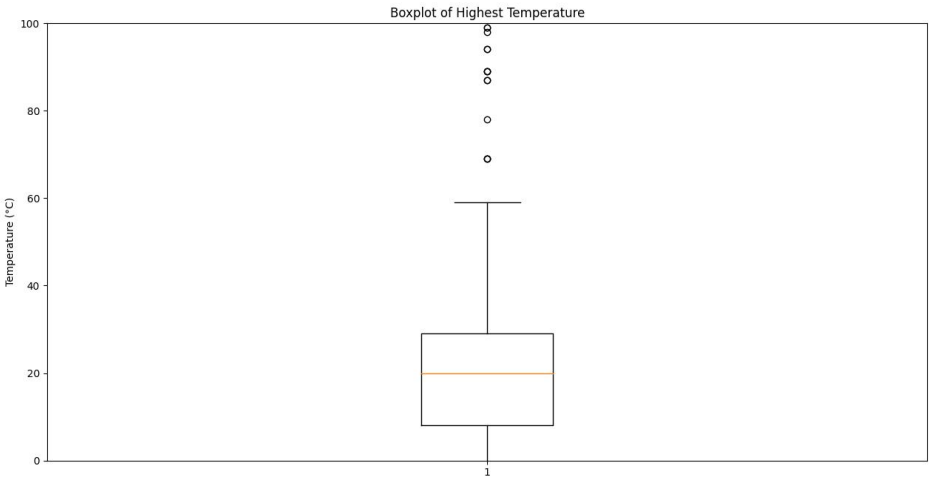


从结果可以看出，气温变化在年际度量上具有周期性。

最高气温有几个点明显过高，属于异常点。

为此分析一下最高气温的箱线图。

```
# 绘制箱线图
plt.boxplot(df['temp_max'])
plt.ylim(0, 100)  # 设定y轴上限为100°C
plt.title('Boxplot of Highest Temperature')
plt.ylabel('Temperature (°C)')
plt.show()
```



可以观察到有少数几个异常点。

4．分析结果总结

1. 各属性特征

- 日期:显示检测日期,数据完整无遗漏

- AQI:数值在80左右最集中，集中在20-200之间。与PM2.5,PM10和CO关联性较强。

- 质量等级：良最多，轻度污染和优较多，整体质量等级较好。

- PM2.5：集中分布在0~100以内

- 天气状况:每个日期记录早晚2种天气类型,共有晴、多云、阴、小雨等情况。由词云图可以看出天气状况的出现情况，多云/多云和晴/晴比较多。

- 最高气温:度数数据类型,值基本上在0-40°C范围内变化，有少数异常点。

- 最低气温:度数数据类型,值基本上在-5-25°C范围内变化。

2. 数据质量问题

- 天气状况中包含的两种类型采用"/"分隔,无法直接使用

- 气温数据中的最高气温存在少量过高异常值

- 空气质量数据采集站点信息缺失

3. 特征选择方向

- 将天气状况、气温处理为两个独立变量提升可解释度

- 剔除异常的气温数据降低噪声

- 综合利用日期、天气和空气质量数据挖掘变化规律

5．结论

通过对北京地区空气质量数据的探索性分析,我得到以下结论:

我对各属性特征进行统计描述分析,发现日期特征完整且AQI属性与pm2.5等元素关联强,气温数据存在少量异常值。此外,天气状况包含多个子特征需要进一步提取。

其次,我识别到数据质量上的几个问题点,包括天气状况格式不便直接利用,少量气温异常值可能会影响分析。此外,站点位置信息的缺失也需进行补充。

最后,从特征选择角度看,应将天气状况细分为多个独立变量,同时去除气温异常值降低噪音。时间序列模型和树模型价值保有挖掘变化规律的潜力。

总体来说,我对空气质量数据进行了初步探索,并识别出数据质量和特征提取方面的待优化点。这为后续数据预处理及建模工作奠定了基础,也为深入研究提供了线索。

针对后续的一些工作建议：

1.数据预处理阶段:

将天气状况特征拆分为两个独立变量提取阶段和类型

提取气温数据中的数值,剔除异常点

采集站点信息缺失,可尝试用地理位置等额外数据填充

2.模型选择方向:

考虑时间序列模型利用日期特征预测空气质量变化趋势

树模型如随机森林可挖掘日期、天气和空气质量之间的关联规则

深度学习方法如LSTM有望捕捉复杂变化规律

3.后续工作:

构建清洗后的数据集

选择不同模型进行建模和结果对比

分析模型结果,查找影响因素寻找改进空气质量的措施

4.项目延伸:

考虑添加空间位置特征如城市等

通过多源数据融合提升预测能力

建立动态调整预警系统