Python程序设计实验报告

数据预处理实验

姓　名：王仕和

学　号：2019213681

日　期：2021.12.21

Python程序设计实验报告
数据预处理实验

本实验完整代码附在压缩包中，并在下列链接可以查看：https://github.com/wshprimy/scrapy-spiders/tree/master/2/

1 作业1：北京链家新房

1.1 项目创建

首先创建项目

```
scrapy startproject Lianjia
```

然后进入项目目录Lianjia下，生成一个spider模板

```
cd Lianjia
scrapy genspider lianjia bj.fang.lianjia.com
```

1.2 Item模块

编辑items.py，定义一个LianjiaItem类，用于描述爬取到数据的结构。

```python
class LianjiaItem(scrapy.Item):
    name = scrapy.Field()
    loc_0 = scrapy.Field()
    loc_1 = scrapy.Field()
    loc_2 = scrapy.Field()
    count = scrapy.Field()
    area = scrapy.Field()
    unit_price = scrapy.Field()
    total_price = scrapy.Field()
    pass
```

1.3 spider模块

编辑spiders/lianjia.py，书写爬虫主程序。
首先根据题目要求，输入起始url

```python
class LianjiaSpider(scrapy.Spider):
    name = 'lianjia'
    allowed_domains = ['bj.fang.lianjia.com']
    start_urls = ['https://bj.fang.lianjia.com/loupan/']
```

在编写主程序时，xpath元素的获取和公共前缀分析，通过for循环爬取各个房源与“数据获取”实验中类似，故不再赘述，详见“数据获取”实验报告。
下面我将着重强调：爬取信息的提取及存储。

1. 首先获取名字、三级地理位置、房型，分别使用.strip()去除字符串字段所有前后空格。

2. 对于面积，我们首先获取面积字段，根据'空格'使用split()分隔得到“建面 ”后面的内容，
根据要求保留面积最小值，根据'-'使用split()分隔得到最小面积，然后使用.strip()去除字符串字段所有前后空格。

3. 对于价格，有两种情况，给定均价或给定总价。首先获得给定的后缀若为“元/㎡(均价)”则为均价，否则则为总价。

    (a) 若给定均价，首先按照根据'-'使用split()分隔得到最小均价，然后使用.strip()去除字符串字段所有前后空格。
    然后根据面积 /* 均价 = 总价，并进行单位换算的到总价。

    (b) 若给定总价，首先按照根据'-'使用split()分隔得到最小总价，然后使用.strip()去除字符串字段所有前后空格。
    然后根据均价 = 总价 / 面积，并进行单位换算的到均价。
    此处需要注意不同分支的unit_price和total_price在items中的LianjiaItem项中的相随顺序要保证一致。

最后将总价保留四位小数，转至piplines输出。

其中上述模块使用try包裹，若出现错误，则表示部分数值存在空值，我采取了直接丢弃整条记录的方式，使用except打印错误信息后继续循环。

```python
def parse(self, response):
    houses = response.xpath('/html/body/div[3]/ul[2]/li')
    for house in houses:
        try:
            item = LianjiaItem()
            item['name'] = house.xpath('./div/div[1]/a/text()').get().strip()
            loc = house.xpath('./div/div[2]')
            item['loc_0'] = loc.xpath('./span[1]/text()').get().strip()
            item['loc_1'] = loc.xpath('./span[2]/text()').get().strip()
            item['loc_2'] = loc.xpath('./a/text()').get().strip()
            item['count'] = house.xpath('./div/a/span[1]/text()').get().strip()
            area = house.xpath('./div/div[3]/span/text()').get()
            area = area.split(' ')[1].split('-')[0].rstrip('㎡')
            item['area'] = int(area)
            flag = house.xpath('./div/div[6]/div[1]/span[2]/text()').get().strip()
            if flag == '元/㎡(均价)':
                unit_price = house.xpath('./div/div[6]/div[1]/span[1]/text()').get()
                unit_price = unit_price.split('-')[0].strip()
                item['unit_price'] = int(unit_price)
                item['total_price'] = item['unit_price'] * item['area'] / 10000
            else:
                total_price = house.xpath('./div/div[6]/div[1]/span[1]/text()').get()
                total_price = total_price.split('-')[0].strip()
                item['unit_price'] = 0
                item['total_price'] = float(total_price)
                item['unit_price'] = round(item['total_price'] / item['area'] * 10000)
            item['total_price'] = '{:.4f}'.format(item['total_price'])
            yield item
        except:
            self.logger.info('Recieved an item containing null values')
```

当前页面所有房源爬取完毕后，根据我对DownloaderMiddleware模块的定义，我使用了一个小trick来获取下一页。
我定义response.url存储的为下一页的url，若其值为'http://none/'则表示已为最后一页，可以退出爬虫。
其中DownloaderMiddleware模块会在后续详细说明。

```
self.logger.info(f'Recieved next_page: {response.url}')
if response.url == 'http://none/':
    # 由于HtmlResponse要求url必须为一个合法的url，故我们定义'http://none/'为结束的标志
    pass
else:
    yield scrapy.Request(response.url)
```

1.4 pipelines模块

编辑pipelines.py，定义一个LianjiaPipeline类，用于处理yield item返回的已爬取数据。

首先打开settings.py，将pipelines启用，优先级为默认。

```
ITEM_PIPELINES = {
   'Lianjia.pipelines.LianjiaPipeline': 300,
}
```

LianjiaPipeline共有两个函数：

1. process_item用于处理每次yield item返回的已爬取数据
2. close_spider用于最后将数据写入文件

process_item直接将每次yield item返回的数据转换成一个list，然后append进入等待写入csv文件的数组。
close_spider首先创建一个csv表头，然后将表头和数据分别写入CSV中。
这里我为了方便在Windows下使用office-excel查看，使用了utf-8-sig编码。

```
class LianjiaPipeline:
    items = []

    def process_item(self, item, spider):
        item_list = list(ItemAdapter(item).asdict().values())
        self.items.append(item_list)
        return item

    def close_spider(self, spider):
        headers = ['名称', '地理位置1', '地理位置2', '地理位置3', '房型', '面积', '均价', '总价']
        # ['name', 'location_area', 'location_town', 'location_exact', 'type', 'area', 'price', 'price_type']
        with open('loupan.csv', 'w', newline='', encoding='utf-8-sig') as file:
            file_csv = csv.writer(file)
            file_csv.writerow(headers)
            file_csv.writerows(self.items)
            file.close()
```

1.5 DownloaderMiddleware模块

编辑middlewares.py，修改已经定义但未实现的LianjiaDownloaderMiddleware类中的函数process_request，用于处理任何通过下载中间件的request。

首先打开settings.py，将DownloaderMiddleware启用，优先级为默认。

```
DOWNLOADER_MIDDLEWARES = {
   'Lianjia.middlewares.LianjiaDownloaderMiddleware': 543,
}
```

下载中间件的功能是使用selenium提供的webdriver将动态网页转换为静态，我需要的部分为下一页按钮是否可用，以及下一页链接。
首先在默认配置基础上添加部分配置，然后启动一个基于Chrome的driver，获取当前request的url，等待至多5s以便网页完全渲染，获得网页html源代码存储至page_source。
然后根据名称'next'获取按钮，尝试点击，等待十秒，既使得网页能够成功跳转，更主要是为了防止触发网站的反爬虫策略，此时网页即为我需要的下一页url。
上述过程包裹在一个try中，若出现错误则表示下一页按钮点击失败，将下一页标记为'http://none/'，表示已到达结尾，可以退出爬虫。
最终返回预保存的本页html源代码和下一页网页的url即可。

```
def process_request(self, request, spider):
    driver_options = webdriver.ChromeOptions()
    driver_options.add_argument('--headless')
    driver_options.add_argument('--disable-gpu')
    driver_options.add_argument('--window-size=1920,1080')
    driver = webdriver.Chrome(options = driver_options)
    driver.get(request.url)
    driver.implicitly_wait(5)
    page_source = driver.page_source
    try:
        button = driver.find_element_by_class_name('next')
        button.click()
        time.sleep(10)
        next_page = driver.current_url
    except:
        next_page = 'http://none/'
        # 由于HtmlResponse要求url必须为一个合法的url，故我们定义'http://none/'为结束的标志
    driver.quit()
    return HtmlResponse(url=next_page, body=page_source, request=request, encoding='utf-8')
```

北京链家新房的数据获取部分结束，下面进行数据预处理部分。

1.6 数据预处理

此部分位于process.py
首先解决输出时列名不对齐问题，然后使用pandas的read_csv读入csv，得到一个dataframe类型变量。

```
pd.set_option('display.unicode.east_asian_width', True)
data = pd.read_csv('./loupan.csv', encoding='utf-8-sig')
```

然后依次输出总价最贵/便宜的房子，中位数；以及均价最贵/便宜的房子，中位数。代码，结果如下：

```
print('--------------------')
print('总价最贵的房子为：')
totalmax_id = data.loc[:, '总价'].idxmax()
print(data.loc[totalmax_id])

print('--------------------')
print('总价最便宜的房子为：')
totalmin_id = data.loc[:, '总价'].idxmin()
print(data.loc[totalmin_id])

print('--------------------')
print('总价的中位数：')
totalmin_id = data.loc[:, '总价'].idxmin()
print('{:.4f}'.format(data.loc[:, '总价'].median()))

print('--------------------')
print('均价最贵的房子为：')
totalmax_id = data.loc[:, '均价'].idxmax()
print(data.loc[totalmax_id])

print('--------------------')
print('均价最便宜的房子为：')
totalmin_id = data.loc[:, '均价'].idxmin()
print(data.loc[totalmin_id])

print('--------------------')
print('单价的中位数：')
totalmin_id = data.loc[:, '均价'].idxmin()
print('{:.4f}'.format(data.loc[:, '均价'].median()))
```

```
--------------------
总价最贵的房子为：
```

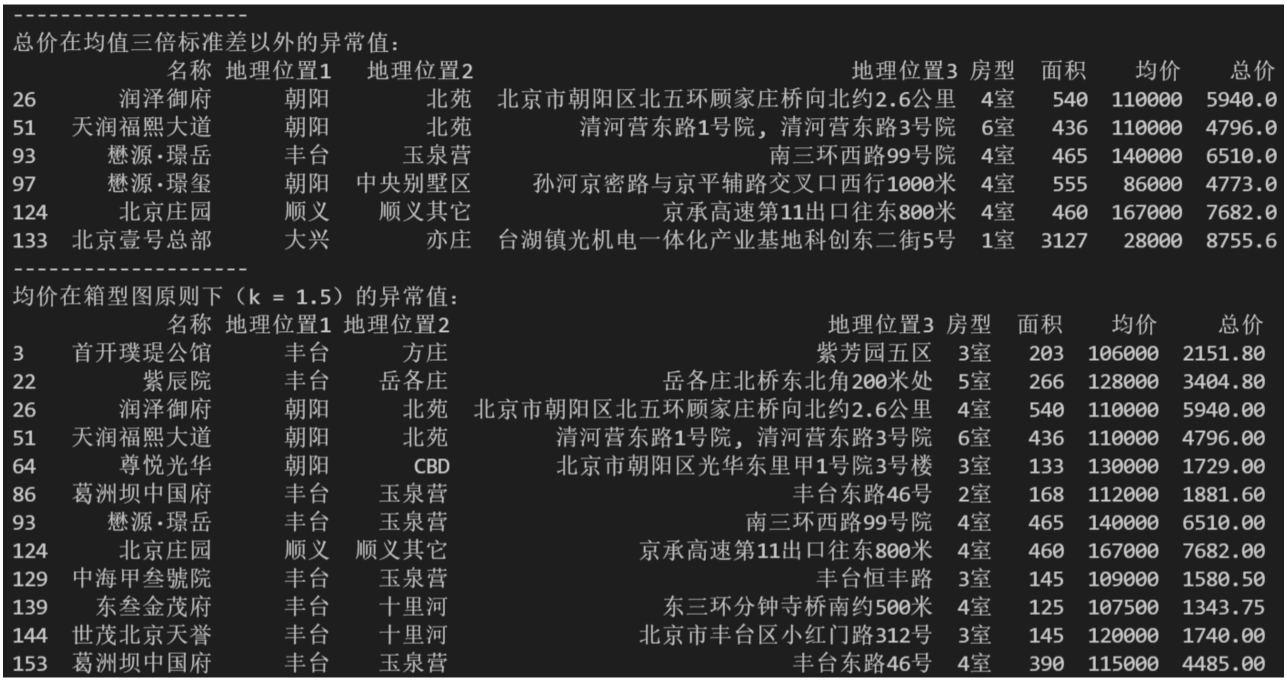
```
名称                                    北京壹号总部
地理位置1                                       大兴
地理位置2                                       亦庄
地理位置3    台湖镇光机电一体化产业基地科创东二街5号
房型                                             1室
面积                                            3127
均价                                           28000
总价                                          8755.6
Name: 133, dtype: object
--------------------
总价最便宜的房子为:
名称                     长海御墅
地理位置1                    房山
地理位置2                房山其它
地理位置3    长沟国家湿地公园南侧
房型                          1室
面积                           70
均价                        15000
总价                        105.0
Name: 141, dtype: object
--------------------
总价的中位数:
560.7000
--------------------
均价最贵的房子为:
名称                          北京庄园
地理位置1                         顺义
地理位置2                     顺义其它
地理位置3    京承高速第11出口往东800米
房型                               4室
面积                               460
均价                            167000
总价                            7682.0
Name: 124, dtype: object
--------------------
均价最便宜的房子为:
名称                     长海御墅
地理位置1                    房山
地理位置2                房山其它
地理位置3    长沟国家湿地公园南侧
房型                          1室
面积                           70
均价                        15000
总价                        105.0
Name: 141, dtype: object
--------------------
单价的中位数:
47000.0000
```

然后输出总价在均值三倍标准差以外的异常值，均价在箱型图原则下（k = 1.5）的异常值，均价离散化处理。

```python
print('--------------------')
print('总价在均值三倍标准差以外的异常值：')
down = data['总价'].mean() - 3 * data['总价'].std()
up = data['总价'].mean() + 3 * data['总价'].std()
print(data.loc[(data['总价'] < down) | (data['总价'] > up)])

print('--------------------')
print('均价在箱型图原则下（k = 1.5）的异常值：')
k = 1.5
q1 = data['均价'].quantile(q=0.25)
q3 = data['均价'].quantile(q=0.75)
down = q1 - k * (q3 - q1)
up = q3 + k * (q3 - q1)
print(data.loc[(data['均价'] < down) | (data['均价'] > up)])

print('--------------------')
print('均价离散化处理：')
avgs = [0, 20000, 40000, 60000, 80000, 100000, 120000, 140000, 160000,
180000]
cuts = pd.cut(data['均价'], avgs)
print(pd.value_counts(cuts))
print('--------------------')
```



```
--------------------
总价在均值三倍标准差以外的异常值：
          名称 地理位置1   地理位置2                       地理位置3 房型   面积    均价    总价
26      润泽御府    朝阳      北苑  北京市朝阳区北五环顾家庄桥向北约2.6公里  4室   540  110000  5940.0
51    天润福熙大道    朝阳      北苑       清河营东路1号院，清河营东路3号院  6室   436  110000  4796.0
93     懋源·璟岳    丰台     玉泉营               南三环西路99号院  4室   465  140000  6510.0
97     懋源·璟玺    朝阳   中央别墅区    孙河京密路与京平辅路交叉口西行1000米  4室   555   86000  4773.0
124     北京庄园    顺义    顺义其它            京承高速第11出口往东800米  4室   460  167000  7682.0
133   北京壹号总部    大兴      亦庄  台湖镇光机电一体化产业基地科创东二街5号  1室  3127   28000  8755.6
--------------------
均价在箱型图原则下（k = 1.5）的异常值：
          名称 地理位置1 地理位置2                       地理位置3 房型  面积    均价     总价
3     首开璞瑅公馆    丰台    方庄                     紫芳园五区  3室  203  106000  2151.80
22       紫辰院    丰台   岳各庄             岳各庄北桥东北角200米处  5室  266  128000  3404.80
26      润泽御府    朝阳    北苑  北京市朝阳区北五环顾家庄桥向北约2.6公里  4室  540  110000  5940.00
51    天润福熙大道    朝阳    北苑       清河营东路1号院，清河营东路3号院  6室  436  110000  4796.00
64      尊悦光华    朝阳    CBD       北京市朝阳区光华东里甲1号院3号楼  3室  133  130000  1729.00
86     葛洲坝中国府    丰台   玉泉营                  丰台东路46号  2室  168  112000  1881.60
93     懋源·璟岳    丰台   玉泉营               南三环西路99号院  4室  465  140000  6510.00
124     北京庄园    顺义  顺义其它            京承高速第11出口往东800米  4室  460  167000  7682.00
129    中海甲叁號院    丰台   玉泉营                     丰台恒丰路  3室  145  109000  1580.50
139     东叁金茂府    丰台   十里河            东三环分钟寺桥南约500米  4室  125  107500  1343.75
144    世茂北京天誉    丰台   十里河             北京市丰台区小红门路312号  3室  145  120000  1740.00
153    葛洲坝中国府    丰台   玉泉营                  丰台东路46号  4室  390  115000  4485.00
```

```
均价离散化处理:
                    均价        百分比
(0, 20000]            4   2.339181
(20000, 40000]       64  37.426901
(40000, 60000]       55  32.163743
(60000, 80000]       29  16.959064
(80000, 100000]       7   4.093567
(100000, 120000]      8   4.678363
(120000, 140000]      3   1.754386
(140000, 160000]      0   0.000000
(160000, 180000]      1   0.584795
--------------------
```

观察发现:

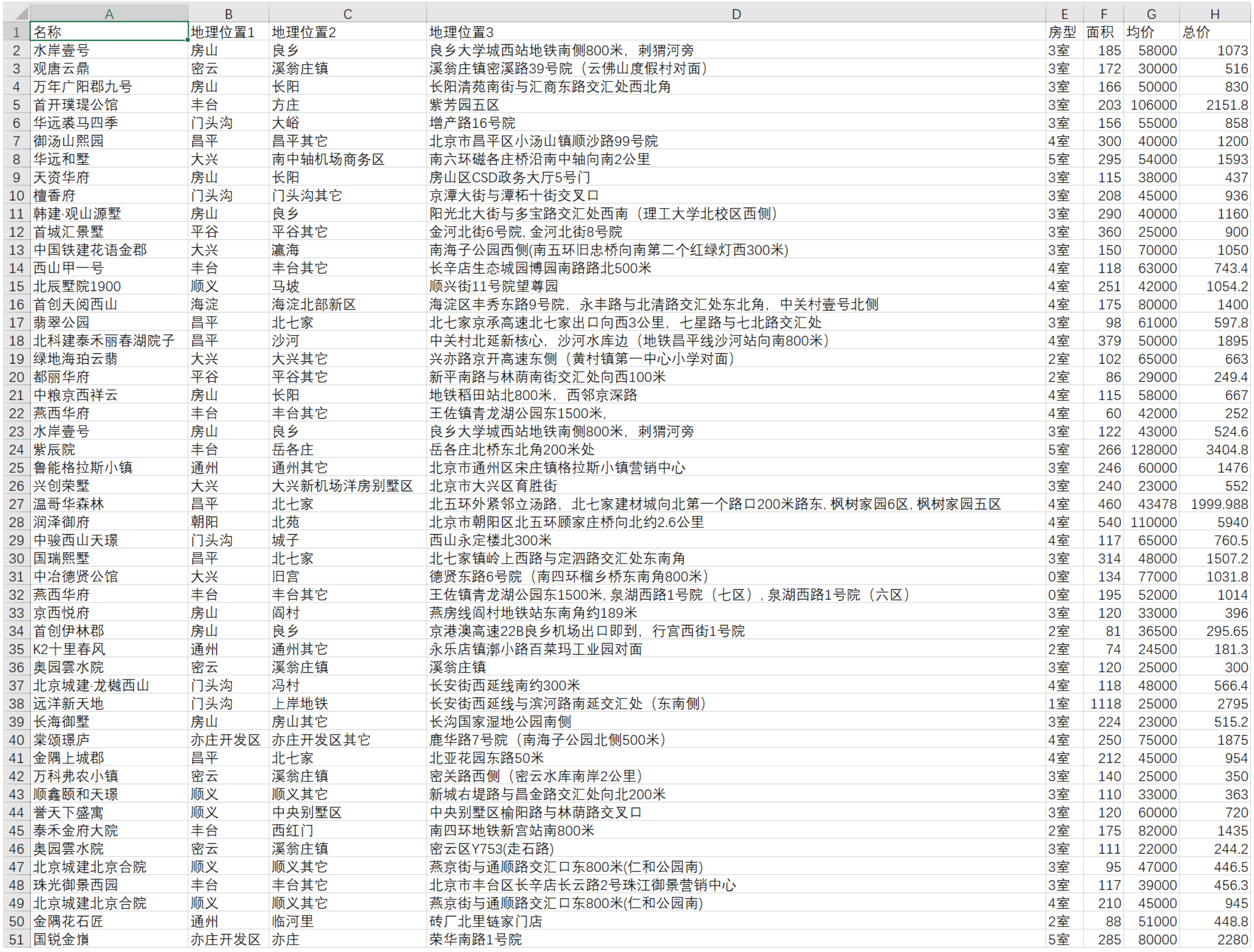
1. 总价在均值三倍标准差以外的异常值，均为超过均值三倍的异常值，所有房屋均具有面积巨大的特点，大部分房屋同时也具有均价高的特点。
2. 均价在箱型图原则下（k = 1.5）的异常值，均为超过箱型图上边缘的异常值，这些房屋具有地段优越，交通便利的特点，大部分房屋同时也具有总价高的特点。

离散化区间长度20000，设定为以下区间:

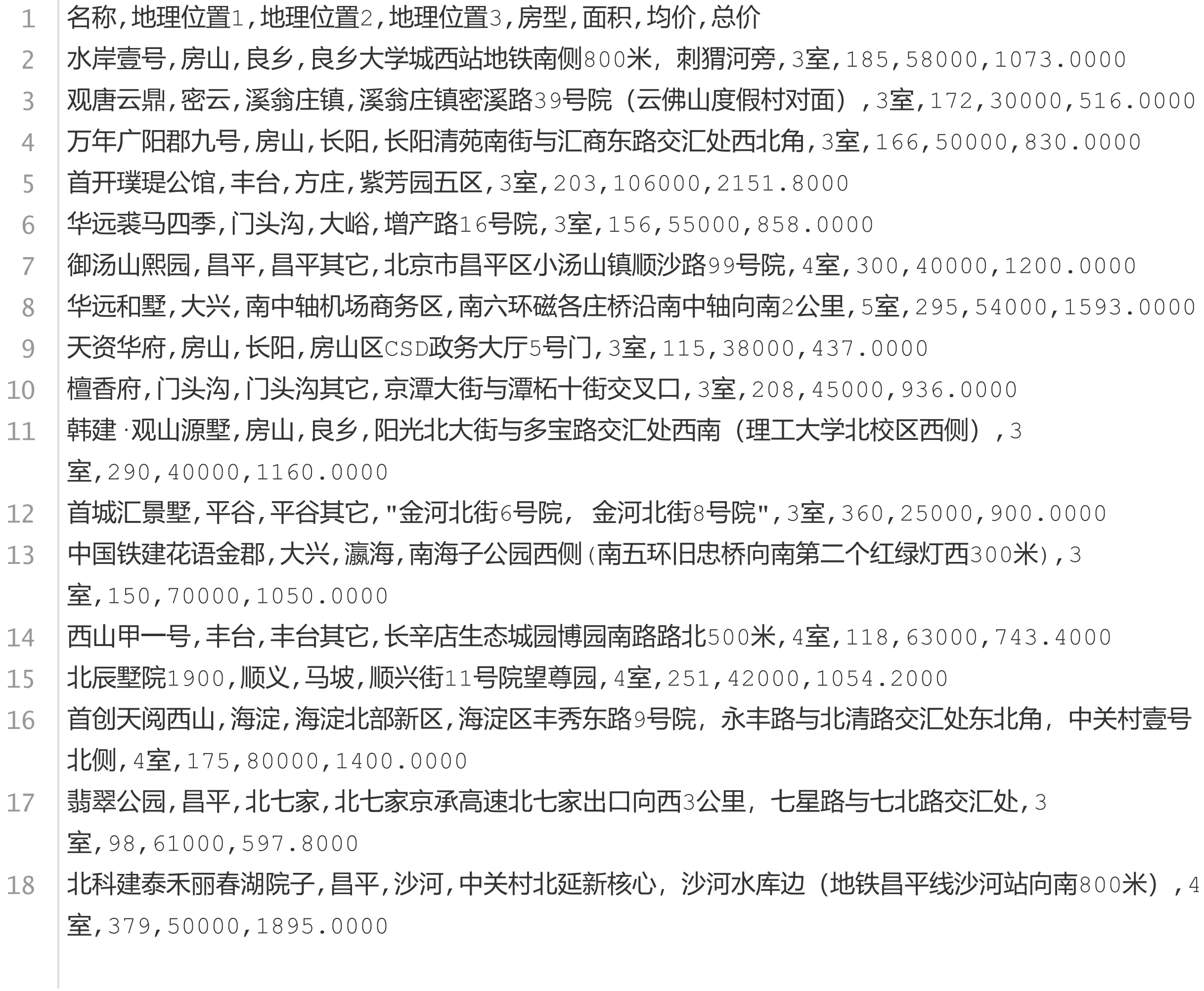
```
avgs = [0, 20000, 40000, 60000, 80000, 100000, 120000, 140000, 160000, 180000]
```

每隔两万元设定一个区间较为合理，使得划分的区间不至于过多，又能够体现出均价的分布情况。

1.7 数据



	A	B	C	D	E	F	G	H
1	名称	地理位置1	地理位置2	地理位置3	房型	面积	均价	总价
2	水岸壹号	房山	良乡	良乡大学城西站地铁南侧800米，刺猬河旁	3室	185	58000	1073
3	观唐云鼎	密云	溪翁庄镇	溪翁庄镇密溪路39号院（云佛山度假村对面）	3室	172	30000	516
4	万年广阳郡九号	房山	长阳	长阳清苑南街与汇商东路交汇处西北角	3室	166	50000	830
5	首开璞瑅公馆	丰台	方庄	紫芳园五区	3室	203	106000	2151.8
6	华远裘马四季	门头沟	大峪	增产路16号院	3室	156	55000	858
7	御汤山熙园	昌平	昌平其它	北京市昌平区小汤山镇顺沙路99号院	4室	300	40000	1200
8	华远和墅	大兴	南中轴机场商务区	南六环磁各庄桥沿南中轴向南2公里	5室	295	54000	1593
9	天资华府	房山	长阳	房山区CSD政务大厅5号门	3室	115	38000	437
10	檀香府	门头沟	门头沟其它	京潭大街与潭柘十街交叉口	3室	208	45000	936
11	韩建·观山源墅	房山	良乡	阳光北大街与多宝路交汇处西南（理工大学北校区西侧）	3室	290	40000	1160
12	首城汇景墅	平谷	平谷其它	金河北街6号院, 金河北街8号院	3室	360	25000	900
13	中国铁建花语金郡	大兴	瀛海	南海子公园西侧(南五环旧忠桥向南第二个红绿灯西300米)	3室	150	70000	1050
14	西山甲一号	丰台	丰台其它	长辛店生态城园博园南路路北500米	4室	118	63000	743.4
15	北辰墅院1900	顺义	马坡	顺兴街11号院望尊园	4室	251	42000	1054.2
16	首创天阅西山	海淀	海淀北部新区	海淀区丰秀东路9号院，永丰路与北清路交汇处东北角，中关村壹号北侧	4室	175	80000	1400
17	翡翠公园	昌平	北七家	北七家京承高速北七家出口向西3公里，七星路与七北路交汇处	3室	98	61000	597.8
18	北科建泰禾丽春湖院子	昌平	沙河	中关村北延新核心，沙河水库边（地铁昌平线沙河站向南800米）	4室	379	50000	1895
19	绿地海珀云翡	大兴	大兴其它	兴亦路京开高速东侧（黄村镇第一中心小学对面）	2室	102	65000	663
20	都丽华府	平谷	平谷其它	新平南路与林荫南街交汇处向西100米	2室	86	29000	249.4
21	中粮京西祥云	房山	长阳	地铁稻田站北800米，西邻京深路	4室	115	58000	667
22	燕西华府	丰台	丰台其它	王佐镇青龙湖公园东1500米,	4室	60	42000	252
23	水岸壹号	房山	良乡	良乡大学城西站地铁南侧800米，刺猬河旁	3室	122	43000	524.6
24	紫辰院	丰台	岳各庄	岳各庄北桥东北角200米处	5室	266	128000	3404.8
25	鲁能格拉斯小镇	通州	通州其它	北京市通州区宋庄镇格拉斯小镇营销中心	3室	246	60000	1476
26	兴创荣墅	大兴	大兴新机场洋房别墅区	北京市大兴区育胜街	3室	240	23000	552
27	温哥华森林	昌平	北七家	北五环外紧邻立汤路，北七家建材城向北第一个路口200米路东, 枫树家园6区, 枫树家园五区	4室	460	43478	1999.988
28	润泽御府	朝阳	北苑	北京市朝阳区北五环顾家庄桥向北约2.6公里	4室	540	110000	5940
29	中骏西山天璟	门头沟	城子	西山永定楼北300米	4室	117	65000	760.5
30	国瑞熙墅	昌平	北七家	北七家镇岭上西路与定泗路交汇处东南角	3室	314	48000	1507.2
31	中冶德贤公馆	大兴	旧宫	德贤东路6号院（南四环榴乡桥东南角800米）	0室	134	77000	1031.8
32	燕西华府	丰台	丰台其它	王佐镇青龙湖公园东1500米, 泉湖西路1号院（七区）, 泉湖西路1号院（六区）	0室	195	52000	1014
33	京西悦府	房山	阎村	燕房线阎村地铁站东南角约189米	3室	120	33000	396
34	首创伊林郡	房山	良乡	京港澳高速22B良乡机场出口即到，行宫西街1号院	2室	81	36500	295.65
35	K2十里春风	通州	通州其它	永乐店镇漷小路百菜玛工业园对面	2室	74	24500	181.3
36	奥园雲水院	密云	溪翁庄镇	溪翁庄镇	3室	120	25000	300
37	北京城建·龙樾西山	门头沟	冯村	长安街西延线南约300米	4室	118	48000	566.4
38	远洋新天地	门头沟	上岸地铁	长安街西延线与滨河路南延交汇处（东南侧）	1室	1118	25000	2795
39	长海御墅	房山	房山其它	长沟国家湿地公园南侧	3室	224	23000	515.2
40	棠颂璟庐	亦庄开发区	亦庄开发区其它	鹿华路7号院（南海子公园北侧500米）	4室	250	75000	1875
41	金隅上城郡	昌平	北七家	北亚花园东路50米	4室	212	45000	954
42	万科弗农小镇	密云	溪翁庄镇	密关路西侧（密云水库南岸2公里）	3室	140	25000	350
43	顺鑫颐和天璟	顺义	顺义其它	新城右堤路与昌金路交汇处向北200米	3室	110	33000	363
44	誉天下盛寓	顺义	中央别墅区	中央别墅区榆阳路与林荫路交叉口	3室	120	60000	720
45	泰禾金府大院	丰台	西红门	南四环地铁新宫站南800米	2室	175	82000	1435
46	奥园雲水院	密云	溪翁庄镇	密云区Y753(走石路)	3室	111	22000	244.2
47	北京城建北京合院	顺义	顺义其它	燕京街与通顺路交汇口东800米(仁和公园南)	3室	95	47000	446.5
48	珠光御景西园	丰台	丰台其它	北京市丰台区长辛店长云路2号珠江御景营销中心	3室	117	39000	456.3
49	北京城建北京合院	顺义	顺义其它	燕京街与通顺路交汇口东800米(仁和公园南)	4室	210	45000	945
50	金隅花石匠	通州	临河里	砖厂北里链家门店	2室	88	51000	448.8
51	国锐金嶺	亦庄开发区	亦庄	荣华南路1号院	5室	285	80000	2280



```
名称,地理位置1,地理位置2,地理位置3,房型,面积,均价,总价
水岸壹号,房山,良乡,良乡大学城西站地铁南侧800米，刺猬河旁,3室,185,58000,1073.0000
观唐云鼎,密云,溪翁庄镇,溪翁庄镇密溪路39号院（云佛山度假村对面）,3室,172,30000,516.0000
万年广阳郡九号,房山,长阳,长阳清苑南街与汇商东路交汇处西北角,3室,166,50000,830.0000
首开璞瑅公馆,丰台,方庄,紫芳园五区,3室,203,106000,2151.8000
华远裘马四季,门头沟,大峪,增产路16号院,3室,156,55000,858.0000
御汤山熙园,昌平,昌平其它,北京市昌平区小汤山镇顺沙路99号院,4室,300,40000,1200.0000
华远和墅,大兴,南中轴机场商务区,南六环磁各庄桥沿南中轴向南2公里,5室,295,54000,1593.0000
天资华府,房山,长阳,房山区CSD政务大厅5号门,3室,115,38000,437.0000
檀香府,门头沟,门头沟其它,京潭大街与潭柘十街交叉口,3室,208,45000,936.0000
韩建·观山源墅,房山,良乡,阳光北大街与多宝路交汇处西南（理工大学北校区西侧）,3室,290,40000,1160.0000
首城汇景墅,平谷,平谷其它,"金河北街6号院，金河北街8号院",3室,360,25000,900.0000
中国铁建花语金郡,大兴,瀛海,南海子公园西侧(南五环旧忠桥向南第二个红绿灯西300米),3室,150,70000,1050.0000
西山甲一号,丰台,丰台其它,长辛店生态城园博园南路路北500米,4室,118,63000,743.4000
北辰墅院1900,顺义,马坡,顺兴街11号院望尊园,4室,251,42000,1054.2000
首创天阅西山,海淀,海淀北部新区,海淀区丰秀东路9号院，永丰路与北清路交汇处东北角，中关村壹号北侧,4室,175,80000,1400.0000
翡翠公园,昌平,北七家,北七家京承高速北七家出口向西3公里，七星路与七北路交汇处,3室,98,61000,597.8000
北科建泰禾丽春湖院子,昌平,沙河,中关村北延新核心，沙河水库边（地铁昌平线沙河站向南800米）,4室,379,50000,1895.0000
```

```
绿地海珀云翡,大兴,大兴其它,兴亦路京开高速东侧（黄村镇第一中心小学对面）,2室,102,65000,663.0000
都丽华府,平谷,平谷其它,新平南路与林荫南街交汇处向西100米,2室,86,29000,249.4000
中粮京西祥云,房山,长阳,地铁稻田站北800米，西邻京深路,4室,115,58000,667.0000
燕西华府,丰台,丰台其它,"王佐镇青龙湖公园东1500米,",4室,60,42000,252.0000
水岸壹号,房山,良乡,良乡大学城西站地铁南侧800米，刺猬河旁,3室,122,43000,524.6000
紫辰院,丰台,岳各庄,岳各庄北桥东北角200米处,5室,266,128000,3404.8000
鲁能格拉斯小镇,通州,通州其它,北京市通州区宋庄镇格拉斯小镇营销中心,3室,246,60000,1476.0000
兴创荣墅,大兴,大兴新机场洋房别墅区,北京市大兴区育胜街,3室,240,23000,552.0000
温哥华森林,昌平,北七家,"北五环外紧邻立汤路，北七家建材城向北第一个路口200米路东，枫树家园6区，枫树家园五区",4室,460,43478,1999.9880
润泽御府,朝阳,北苑,北京市朝阳区北五环顾家庄桥向北约2.6公里,4室,540,110000,5940.0000
中骏西山天璟,门头沟,城子,西山永定楼北300米,4室,117,65000,760.5000
国瑞熙墅,昌平,北七家,北七家镇岭上西路与定泗路交汇处东南角,3室,314,48000,1507.2000
中冶德贤公馆,大兴,旧宫,德贤东路6号院（南四环榴乡桥东南角800米）,0室,134,77000,1031.8000
燕西华府,丰台,丰台其它,"王佐镇青龙湖公园东1500米，泉湖西路1号院（七区），泉湖西路1号院（六区）",0室,195,52000,1014.0000
京西悦府,房山,阎村,燕房线阎村地铁站东南角约189米,3室,120,33000,396.0000
首创伊林郡,房山,良乡,京港澳高速22B良乡机场出口即到，行宫西街1号院,2室,81,36500,295.6500
K2十里春风,通州,通州其它,永乐店镇漷小路百菜玛工业园对面,2室,74,24500,181.3000
奥园雲水院,密云,溪翁庄镇,溪翁庄镇,3室,120,25000,300.0000
北京城建·龙樾西山,门头沟,冯村,长安街西延线南约300米,4室,118,48000,566.4000
远洋新天地,门头沟,上岸地铁,长安街西延线与滨河路南延交汇处（东南侧）,1室,1118,25000,2795.0000
长海御墅,房山,房山其它,长沟国家湿地公园南侧,3室,224,23000,515.2000
棠颂璟庐,亦庄开发区,亦庄开发区其它,鹿华路7号院（南海子公园北侧500米）,4室,250,75000,1875.0000
金隅上城郡,昌平,北七家,北亚花园东路50米,4室,212,45000,954.0000
万科弗农小镇,密云,溪翁庄镇,密关路西侧（密云水库南岸2公里）,3室,140,25000,350.0000
顺鑫颐和天璟,顺义,顺义其它,新城右堤路与昌金路交汇处向北200米,3室,110,33000,363.0000
誉天下盛寓,顺义,中央别墅区,中央别墅区榆阳路与林荫路交叉口,3室,120,60000,720.0000
泰禾金府大院,丰台,西红门,南四环地铁新宫站南800米,2室,175,82000,1435.0000
奥园雲水院,密云,溪翁庄镇,密云区Y753(走石路),3室,111,22000,244.2000
北京城建北京合院,顺义,顺义其它,燕京街与通顺路交汇口东800米(仁和公园南),3室,95,47000,446.5000
珠光御景西园,丰台,丰台其它,北京市丰台区长辛店长云路2号珠江御景营销中心,3室,117,39000,456.3000
北京城建北京合院,顺义,顺义其它,燕京街与通顺路交汇口东800米(仁和公园南),4室,210,45000,945.0000
金隅花石匠,通州,临河里,砖厂北里链家门店,2室,88,51000,448.8000
国锐金嵿,亦庄开发区,亦庄,荣华南路1号院,5室,285,80000,2280.0000
```

2 作业2：2015年北京市PM2.5指数数据集空值

2.1 实验流程

此部分位于process.py
首先解决输出时列名不对齐问题，然后使用pandas的read_csv从原始数据集中读入csv，得到一个dataframe类型变量。使用loc[]将'year'==2015的数据切片后，去除无用的列'No'，然后将其保存至'./BeijingPM2015.csv'，并且不产生额外的编号。

```
pd.set_option('display.unicode.east_asian_width', True)
data = pd.read_csv('./BeijingPM20100101_20151231.csv', encoding='utf-8')
# 数据抽取及存储
data2015 = data.loc[data['year']==2015].drop(columns=['No'])
data2015.to_csv('./BeijingPM2015.csv', index=False, encoding='utf-8')
```

然后使用pandas的read_csv从保存2015年数据的数据集中读入csv，得到一个dataframe类型变量。
首先找到存在空值的列与对应空值数量。
对于列名为以下四个的列：

```
columns = ['PM_Dongsi', 'PM_Dongsihuan', 'PM_Nongzhanguan', 'PM_US Post']
```

我发现了其为一天中四个不同监测点的PM2.5浓度，故使用同一天其他监测点的平均值填充空值，具体实现见下列代码注释。
其他部分与当前其他数据无关联，故使用前一天（上一行）非空值进行填充。

```
data = pd.read_csv('./BeijingPM2015.csv', encoding='utf-8')
print('--------------------')
print('存在的空值列：')
# 找出存在空值的列
print(data.isnull().any())

print('--------------------')
print('列对应的空值数量：')
# 找出对应列空值数量
print(data.isnull().sum(axis=0))

print('--------------------')
columns = ['PM_Dongsi', 'PM_Dongsihuan', 'PM_Nongzhanguan', 'PM_US Post']
# 计算每一行四个pm2.5监测点的平均值
meanpm = data[columns].mean(axis=1)
# 创建一个四个监测点名称到平均值的映射
fill = {}.fromkeys(columns, meanpm)
# 使用映射，将四个监测点的平均值替换某些监测点的空值
data.fillna(value=fill, inplace=True)

# 其他数据使用上一行的非空值填充
data.fillna(method='ffill', inplace=True)
print(data)
```

```python

print('--------------------')
print('空值处理后存在的空值列：')
print(data.isnull().any())

print('--------------------')
print('空值处理后列对应的空值数量：')
print(data.isnull().sum(axis=0))
print('--------------------')

data.to_csv('./BeijingPM2015_cleaned.csv', index=False, encoding='utf-8')
```

2.2 实验结果

2.2.1 代码输出

处理前的空值列:

```
--------------------
存在的空值列:
year               False
month              False
day                False
hour               False
season             False
PM_Dongsi           True
PM_Dongsihuan       True
PM_Nongzhanguan     True
PM_US Post          True
DEWP                True
HUMI                True
PRES                True
TEMP                True
cbwd                True
Iws                 True
precipitation       True
Iprec               True
dtype: bool
--------------------
列对应的空值数量:
year                  0
month                 0
day                   0
hour                  0
season                0
PM_Dongsi           164
PM_Dongsihuan      3295
PM_Nongzhanguan     287
PM_US Post          129
```

```
DEWP                  5
HUMI                339
PRES                339
TEMP                  5
cbwd                  5
Iws                   5
precipitation       459
Iprec               459
dtype: int64
```

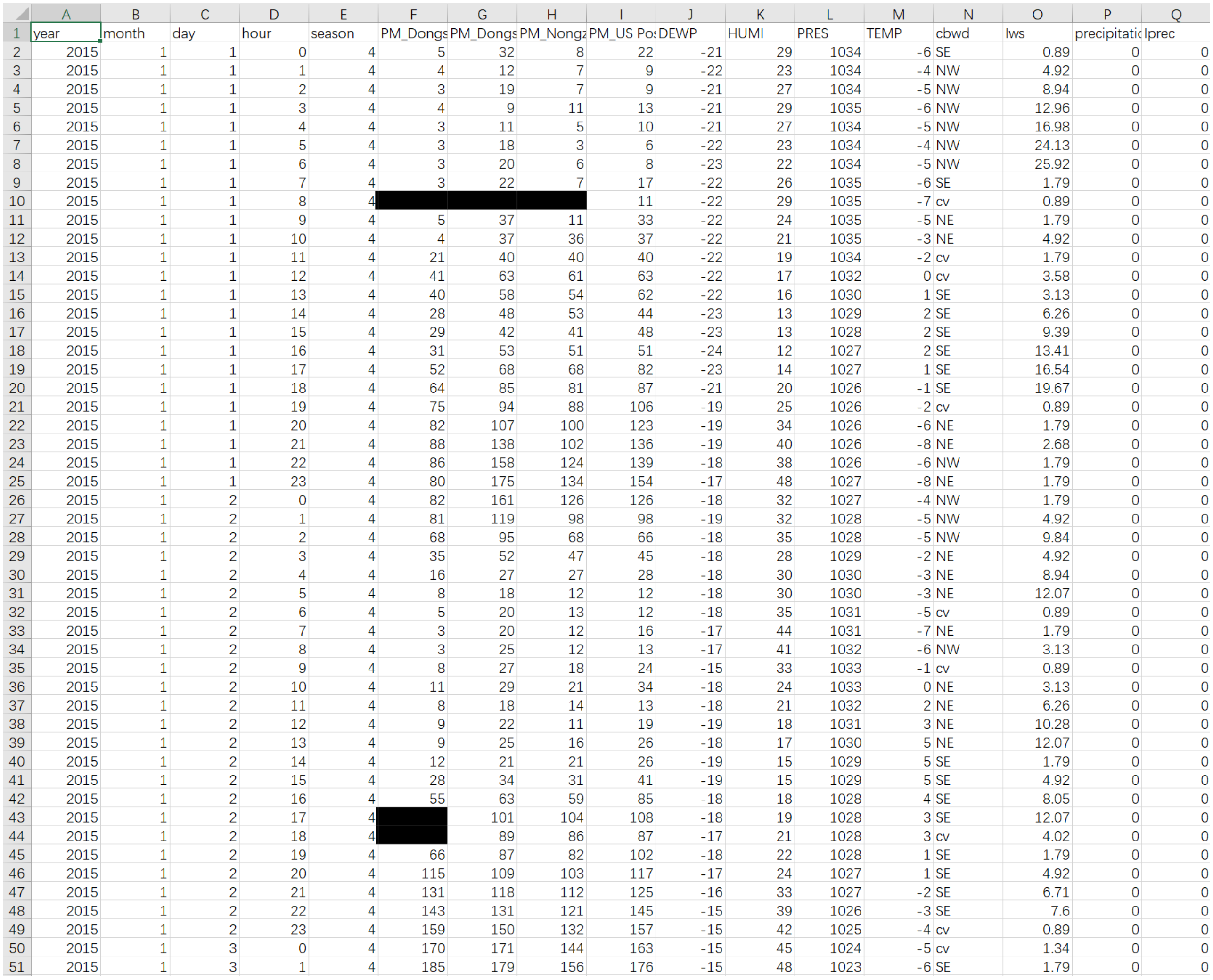
处理后的空值列:

```
--------------------
空值处理后存在的空值列:
year               False
month              False
day                False
hour               False
season             False
PM_Dongsi          False
PM_Dongsihuan      False
PM_Nongzhanguan    False
PM_US Post         False
DEWP               False
HUMI               False
PRES               False
TEMP               False
cbwd               False
Iws                False
precipitation      False
Iprec              False
dtype: bool
--------------------
空值处理后列对应的空值数量:
year               0
month              0
day                0
hour               0
season             0
PM_Dongsi          0
PM_Dongsihuan      0
PM_Nongzhanguan    0
PM_US Post         0
DEWP               0
HUMI               0
PRES               0
TEMP               0
cbwd               0
```

```
Iws                 0
precipitation       0
Iprec               0
dtype: int64
--------------------
```

2.2.2 处理前的2015年csv数据

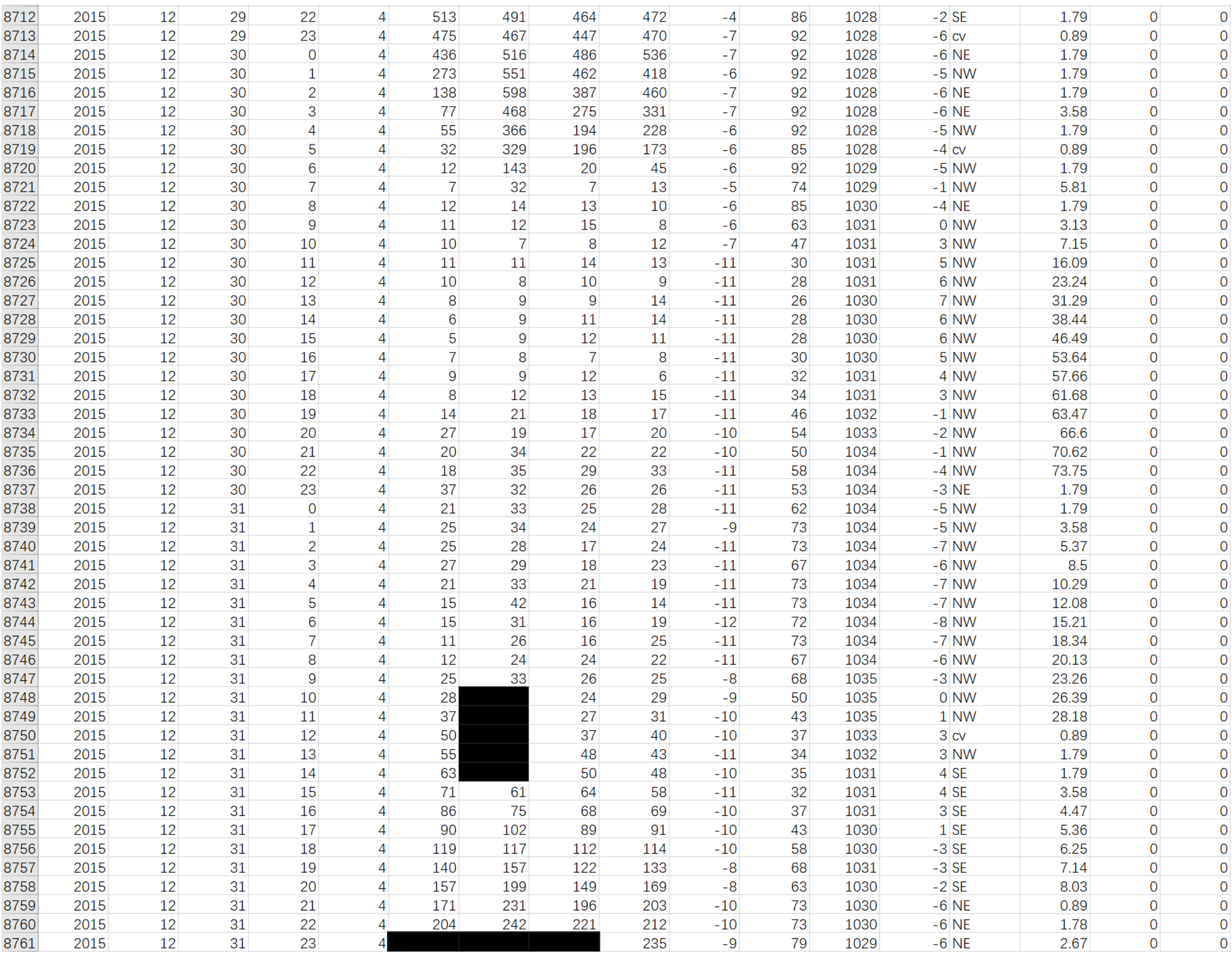
截取前、后各五十行。



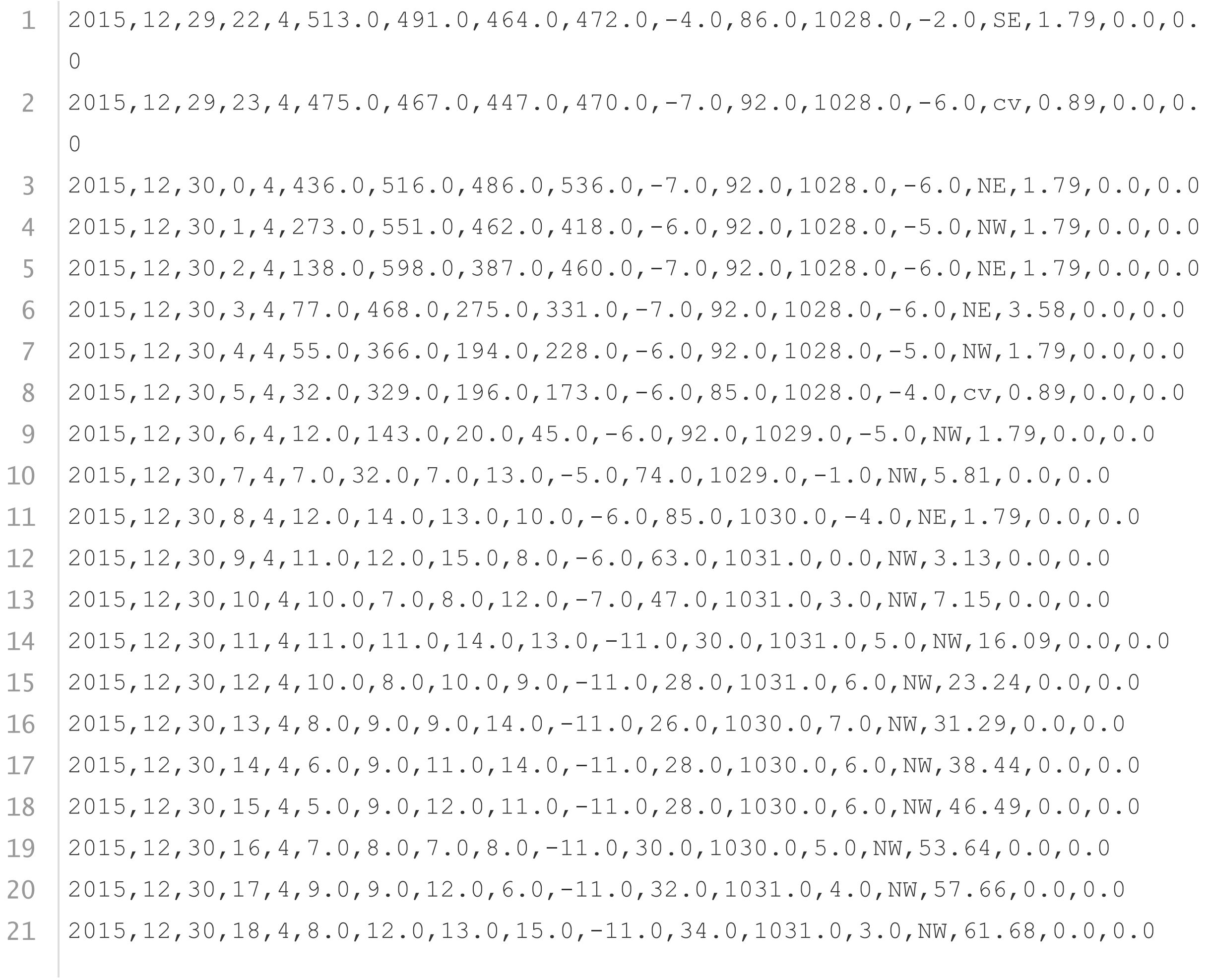
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	year	month	day	hour	season	PM_Dongs	PM_Dongs	PM_Nongz	PM_US Pos	DEWP	HUMI	PRES	TEMP	cbwd	Iws	precipitatic	Iprec
2	2015	1	1	0	4	5	32	8	22	-21	29	1034	-6	SE	0.89	0	0
3	2015	1	1	1	4	4	12	7	9	-22	23	1034	-4	NW	4.92	0	0
4	2015	1	1	2	4	3	19	7	9	-21	27	1034	-5	NW	8.94	0	0
5	2015	1	1	3	4	4	9	11	13	-21	29	1035	-6	NW	12.96	0	0
6	2015	1	1	4	4	3	11	5	10	-21	27	1034	-5	NW	16.98	0	0
7	2015	1	1	5	4	3	18	3	6	-22	23	1034	-4	NW	24.13	0	0
8	2015	1	1	6	4	3	20	6	8	-23	22	1034	-5	NW	25.92	0	0
9	2015	1	1	7	4	3	22	7	17	-22	26	1035	-6	SE	1.79	0	0
10	2015	1	1	8	4	[redacted]	[redacted]	[redacted]	11	-22	29	1035	-7	cv	0.89	0	0
11	2015	1	1	9	4	5	37	11	33	-22	24	1035	-5	NE	1.79	0	0
12	2015	1	1	10	4	4	37	36	37	-22	21	1035	-3	NE	4.92	0	0
13	2015	1	1	11	4	21	40	40	40	-22	19	1034	-2	cv	1.79	0	0
14	2015	1	1	12	4	41	63	61	63	-22	17	1032	0	cv	3.58	0	0
15	2015	1	1	13	4	40	58	54	62	-22	16	1030	1	SE	3.13	0	0
16	2015	1	1	14	4	28	48	53	44	-23	13	1029	2	SE	6.26	0	0
17	2015	1	1	15	4	29	42	41	48	-23	13	1028	2	SE	9.39	0	0
18	2015	1	1	16	4	31	53	51	51	-24	12	1027	2	SE	13.41	0	0
19	2015	1	1	17	4	52	68	68	82	-23	14	1027	1	SE	16.54	0	0
20	2015	1	1	18	4	64	85	81	87	-21	20	1026	-1	SE	19.67	0	0
21	2015	1	1	19	4	75	94	88	106	-19	25	1026	-2	cv	0.89	0	0
22	2015	1	1	20	4	82	107	100	123	-19	34	1026	-6	NE	1.79	0	0
23	2015	1	1	21	4	88	138	102	136	-19	40	1026	-8	NE	2.68	0	0
24	2015	1	1	22	4	86	158	124	139	-18	38	1026	-6	NW	1.79	0	0
25	2015	1	1	23	4	80	175	134	154	-17	48	1027	-8	NE	1.79	0	0
26	2015	1	2	0	4	82	161	126	126	-18	32	1027	-4	NW	1.79	0	0
27	2015	1	2	1	4	81	119	98	98	-19	32	1028	-5	NW	4.92	0	0
28	2015	1	2	2	4	68	95	68	66	-18	35	1028	-5	NW	9.84	0	0
29	2015	1	2	3	4	35	52	47	45	-18	28	1029	-2	NE	4.92	0	0
30	2015	1	2	4	4	16	27	27	28	-18	30	1030	-3	NE	8.94	0	0
31	2015	1	2	5	4	8	18	12	12	-18	30	1030	-3	NE	12.07	0	0
32	2015	1	2	6	4	5	20	13	12	-18	35	1031	-5	cv	0.89	0	0
33	2015	1	2	7	4	3	20	12	16	-17	44	1031	-7	NE	1.79	0	0
34	2015	1	2	8	4	3	25	12	13	-17	41	1032	-6	NW	3.13	0	0
35	2015	1	2	9	4	8	27	18	24	-15	33	1033	-1	cv	0.89	0	0
36	2015	1	2	10	4	11	29	21	34	-18	24	1033	0	NE	3.13	0	0
37	2015	1	2	11	4	8	18	14	13	-18	21	1032	2	NE	6.26	0	0
38	2015	1	2	12	4	9	22	11	19	-19	18	1031	3	NE	10.28	0	0
39	2015	1	2	13	4	9	25	16	26	-18	17	1030	5	NE	12.07	0	0
40	2015	1	2	14	4	12	21	21	26	-19	15	1029	5	SE	1.79	0	0
41	2015	1	2	15	4	28	34	31	41	-19	15	1029	5	SE	4.92	0	0
42	2015	1	2	16	4	55	63	59	85	-18	18	1028	4	SE	8.05	0	0
43	2015	1	2	17	4	[redacted]	101	104	108	-18	19	1028	3	SE	12.07	0	0
44	2015	1	2	18	4	[redacted]	89	86	87	-17	21	1028	3	cv	4.02	0	0
45	2015	1	2	19	4	66	87	82	102	-18	22	1028	1	SE	1.79	0	0
46	2015	1	2	20	4	115	109	103	117	-17	24	1027	1	SE	4.92	0	0
47	2015	1	2	21	4	131	118	112	125	-16	33	1027	-2	SE	6.71	0	0
48	2015	1	2	22	4	143	131	121	145	-15	39	1026	-3	SE	7.6	0	0
49	2015	1	2	23	4	159	150	132	157	-15	42	1025	-4	cv	0.89	0	0
50	2015	1	3	0	4	170	171	144	163	-15	45	1024	-5	cv	1.34	0	0
51	2015	1	3	1	4	185	179	156	176	-15	48	1023	-6	SE	1.79	0	0

```
year,month,day,hour,season,PM_Dongsi,PM_Dongsihuan,PM_Nongzhanguan,PM_US Post,DEWP,HUMI,PRES,TEMP,cbwd,Iws,precipitation,Iprec
2015,1,1,0,4,5.0,32.0,8.0,22.0,-21.0,29.0,1034.0,-6.0,SE,0.89,0.0,0.0
2015,1,1,1,4,4.0,12.0,7.0,9.0,-22.0,23.0,1034.0,-4.0,NW,4.92,0.0,0.0
2015,1,1,2,4,3.0,19.0,7.0,9.0,-21.0,27.0,1034.0,-5.0,NW,8.94,0.0,0.0
2015,1,1,3,4,4.0,9.0,11.0,13.0,-21.0,29.0,1035.0,-6.0,NW,12.96,0.0,0.0
2015,1,1,4,4,3.0,11.0,5.0,10.0,-21.0,27.0,1034.0,-5.0,NW,16.98,0.0,0.0
2015,1,1,5,4,3.0,18.0,3.0,6.0,-22.0,23.0,1034.0,-4.0,NW,24.13,0.0,0.0
2015,1,1,6,4,3.0,20.0,6.0,8.0,-23.0,22.0,1034.0,-5.0,NW,25.92,0.0,0.0
2015,1,1,7,4,3.0,22.0,7.0,17.0,-22.0,26.0,1035.0,-6.0,SE,1.79,0.0,0.0
2015,1,1,8,4,,,,11.0,-22.0,29.0,1035.0,-7.0,cv,0.89,0.0,0.0
2015,1,1,9,4,5.0,37.0,11.0,33.0,-22.0,24.0,1035.0,-5.0,NE,1.79,0.0,0.0
2015,1,1,10,4,4.0,37.0,36.0,37.0,-22.0,21.0,1035.0,-3.0,NE,4.92,0.0,0.0
2015,1,1,11,4,21.0,40.0,40.0,40.0,-22.0,19.0,1034.0,-2.0,cv,1.79,0.0,0.0
```

```
2015,1,1,12,4,41.0,63.0,61.0,63.0,-22.0,17.0,1032.0,0.0,cv,3.58,0.0,0.0
2015,1,1,13,4,40.0,58.0,54.0,62.0,-22.0,16.0,1030.0,1.0,SE,3.13,0.0,0.0
2015,1,1,14,4,28.0,48.0,53.0,44.0,-23.0,13.0,1029.0,2.0,SE,6.26,0.0,0.0
2015,1,1,15,4,29.0,42.0,41.0,48.0,-23.0,13.0,1028.0,2.0,SE,9.39,0.0,0.0
2015,1,1,16,4,31.0,53.0,51.0,51.0,-24.0,12.0,1027.0,2.0,SE,13.41,0.0,0.0
2015,1,1,17,4,52.0,68.0,68.0,82.0,-23.0,14.0,1027.0,1.0,SE,16.54,0.0,0.0
2015,1,1,18,4,64.0,85.0,81.0,87.0,-21.0,20.0,1026.0,-1.0,SE,19.67,0.0,0.0
2015,1,1,19,4,75.0,94.0,88.0,106.0,-19.0,25.0,1026.0,-2.0,cv,0.89,0.0,0.0
2015,1,1,20,4,82.0,107.0,100.0,123.0,-19.0,34.0,1026.0,-6.0,NE,1.79,0.0,0.0
2015,1,1,21,4,88.0,138.0,102.0,136.0,-19.0,40.0,1026.0,-8.0,NE,2.68,0.0,0.0
2015,1,1,22,4,86.0,158.0,124.0,139.0,-18.0,38.0,1026.0,-6.0,NW,1.79,0.0,0.0
2015,1,1,23,4,80.0,175.0,134.0,154.0,-17.0,48.0,1027.0,-8.0,NE,1.79,0.0,0.0
2015,1,2,0,4,82.0,161.0,126.0,126.0,-18.0,32.0,1027.0,-4.0,NW,1.79,0.0,0.0
2015,1,2,1,4,81.0,119.0,98.0,98.0,-19.0,32.0,1028.0,-5.0,NW,4.92,0.0,0.0
2015,1,2,2,4,68.0,95.0,68.0,66.0,-18.0,35.0,1028.0,-5.0,NW,9.84,0.0,0.0
2015,1,2,3,4,35.0,52.0,47.0,45.0,-18.0,28.0,1029.0,-2.0,NE,4.92,0.0,0.0
2015,1,2,4,4,16.0,27.0,27.0,28.0,-18.0,30.0,1030.0,-3.0,NE,8.94,0.0,0.0
2015,1,2,5,4,8.0,18.0,12.0,12.0,-18.0,30.0,1030.0,-3.0,NE,12.07,0.0,0.0
2015,1,2,6,4,5.0,20.0,13.0,12.0,-18.0,35.0,1031.0,-5.0,cv,0.89,0.0,0.0
2015,1,2,7,4,3.0,20.0,12.0,16.0,-17.0,44.0,1031.0,-7.0,NE,1.79,0.0,0.0
2015,1,2,8,4,3.0,25.0,12.0,13.0,-17.0,41.0,1032.0,-6.0,NW,3.13,0.0,0.0
2015,1,2,9,4,8.0,27.0,18.0,24.0,-15.0,33.0,1033.0,-1.0,cv,0.89,0.0,0.0
2015,1,2,10,4,11.0,29.0,21.0,34.0,-18.0,24.0,1033.0,0.0,NE,3.13,0.0,0.0
2015,1,2,11,4,8.0,18.0,14.0,13.0,-18.0,21.0,1032.0,2.0,NE,6.26,0.0,0.0
2015,1,2,12,4,9.0,22.0,11.0,19.0,-19.0,18.0,1031.0,3.0,NE,10.28,0.0,0.0
2015,1,2,13,4,9.0,25.0,16.0,26.0,-18.0,17.0,1030.0,5.0,NE,12.07,0.0,0.0
2015,1,2,14,4,12.0,21.0,21.0,26.0,-19.0,15.0,1029.0,5.0,SE,1.79,0.0,0.0
2015,1,2,15,4,28.0,34.0,31.0,41.0,-19.0,15.0,1029.0,5.0,SE,4.92,0.0,0.0
2015,1,2,16,4,55.0,63.0,59.0,85.0,-18.0,18.0,1028.0,4.0,SE,8.05,0.0,0.0
2015,1,2,17,4,,101.0,104.0,108.0,-18.0,19.0,1028.0,3.0,SE,12.07,0.0,0.0
2015,1,2,18,4,,89.0,86.0,87.0,-17.0,21.0,1028.0,3.0,cv,4.02,0.0,0.0
2015,1,2,19,4,66.0,87.0,82.0,102.0,-18.0,22.0,1028.0,1.0,SE,1.79,0.0,0.0
2015,1,2,20,4,115.0,109.0,103.0,117.0,-17.0,24.0,1027.0,1.0,SE,4.92,0.0,0.0
2015,1,2,21,4,131.0,118.0,112.0,125.0,-16.0,33.0,1027.0,-2.0,SE,6.71,0.0,0.0
2015,1,2,22,4,143.0,131.0,121.0,145.0,-15.0,39.0,1026.0,-3.0,SE,7.6,0.0,0.0
2015,1,2,23,4,159.0,150.0,132.0,157.0,-15.0,42.0,1025.0,-4.0,cv,0.89,0.0,0.0
2015,1,3,0,4,170.0,171.0,144.0,163.0,-15.0,45.0,1024.0,-5.0,cv,1.34,0.0,0.0
2015,1,3,1,4,185.0,179.0,156.0,176.0,-15.0,48.0,1023.0,-6.0,SE,1.79,0.0,0.0
```



8712	2015	12	29	22	4	513	491	464	472	-4	86	1028	-2	SE	1.79	0	0
8713	2015	12	29	23	4	475	467	447	470	-7	92	1028	-6	cv	0.89	0	0
8714	2015	12	30	0	4	436	516	486	536	-7	92	1028	-6	NE	1.79	0	0
8715	2015	12	30	1	4	273	551	462	418	-6	92	1028	-5	NW	1.79	0	0
8716	2015	12	30	2	4	138	598	387	460	-7	92	1028	-6	NE	1.79	0	0
8717	2015	12	30	3	4	77	468	275	331	-7	92	1028	-6	NE	3.58	0	0
8718	2015	12	30	4	4	55	366	194	228	-6	92	1028	-5	NW	1.79	0	0
8719	2015	12	30	5	4	32	329	196	173	-6	85	1028	-4	cv	0.89	0	0
8720	2015	12	30	6	4	12	143	20	45	-6	92	1029	-5	NW	1.79	0	0
8721	2015	12	30	7	4	7	32	7	13	-5	74	1029	-1	NW	5.81	0	0
8722	2015	12	30	8	4	12	14	13	10	-6	85	1030	-4	NE	1.79	0	0
8723	2015	12	30	9	4	11	12	15	8	-6	63	1031	0	NW	3.13	0	0
8724	2015	12	30	10	4	10	7	8	12	-7	47	1031	3	NW	7.15	0	0
8725	2015	12	30	11	4	11	11	14	13	-11	30	1031	5	NW	16.09	0	0
8726	2015	12	30	12	4	10	8	10	9	-11	28	1031	6	NW	23.24	0	0
8727	2015	12	30	13	4	8	9	9	14	-11	26	1030	7	NW	31.29	0	0
8728	2015	12	30	14	4	6	9	11	14	-11	28	1030	6	NW	38.44	0	0
8729	2015	12	30	15	4	5	9	12	11	-11	28	1030	6	NW	46.49	0	0
8730	2015	12	30	16	4	7	8	7	8	-11	30	1030	5	NW	53.64	0	0
8731	2015	12	30	17	4	9	9	12	6	-11	32	1031	4	NW	57.66	0	0
8732	2015	12	30	18	4	8	12	13	15	-11	34	1031	3	NW	61.68	0	0
8733	2015	12	30	19	4	14	21	18	17	-11	46	1032	-1	NW	63.47	0	0
8734	2015	12	30	20	4	27	19	17	20	-10	54	1033	-2	NW	66.6	0	0
8735	2015	12	30	21	4	20	34	22	22	-10	50	1034	-1	NW	70.62	0	0
8736	2015	12	30	22	4	18	35	29	33	-11	58	1034	-4	NW	73.75	0	0
8737	2015	12	30	23	4	37	32	26	26	-11	53	1034	-3	NE	1.79	0	0
8738	2015	12	31	0	4	21	33	25	28	-11	62	1034	-5	NW	1.79	0	0
8739	2015	12	31	1	4	25	34	24	27	-9	73	1034	-5	NW	3.58	0	0
8740	2015	12	31	2	4	25	28	17	24	-11	73	1034	-7	NW	5.37	0	0
8741	2015	12	31	3	4	27	29	18	23	-11	67	1034	-6	NW	8.5	0	0
8742	2015	12	31	4	4	21	33	21	19	-11	73	1034	-7	NW	10.29	0	0
8743	2015	12	31	5	4	15	42	16	14	-11	73	1034	-7	NW	12.08	0	0
8744	2015	12	31	6	4	15	31	16	19	-12	72	1034	-8	NW	15.21	0	0
8745	2015	12	31	7	4	11	26	16	25	-11	73	1034	-7	NW	18.34	0	0
8746	2015	12	31	8	4	12	24	24	22	-11	67	1034	-6	NW	20.13	0	0
8747	2015	12	31	9	4	25	33	26	25	-8	68	1035	-3	NW	23.26	0	0
8748	2015	12	31	10	4	28	[redacted]	24	29	-9	50	1035	0	NW	26.39	0	0
8749	2015	12	31	11	4	37	[redacted]	27	31	-10	43	1035	1	NW	28.18	0	0
8750	2015	12	31	12	4	50	[redacted]	37	40	-10	37	1033	3	cv	0.89	0	0
8751	2015	12	31	13	4	55	[redacted]	48	43	-11	34	1032	3	NW	1.79	0	0
8752	2015	12	31	14	4	63	[redacted]	50	48	-10	35	1031	4	SE	1.79	0	0
8753	2015	12	31	15	4	71	61	64	58	-11	32	1031	4	SE	3.58	0	0
8754	2015	12	31	16	4	86	75	68	69	-10	37	1031	3	SE	4.47	0	0
8755	2015	12	31	17	4	90	102	89	91	-10	43	1030	1	SE	5.36	0	0
8756	2015	12	31	18	4	119	117	112	114	-10	58	1030	-3	SE	6.25	0	0
8757	2015	12	31	19	4	140	157	122	133	-8	68	1031	-3	SE	7.14	0	0
8758	2015	12	31	20	4	157	199	149	169	-8	63	1030	-2	SE	8.03	0	0
8759	2015	12	31	21	4	171	231	196	203	-10	73	1030	-6	NE	0.89	0	0
8760	2015	12	31	22	4	204	242	221	212	-10	73	1030	-6	NE	1.78	0	0
8761	2015	12	31	23	4	[redacted]	[redacted]	[redacted]	235	-9	79	1029	-6	NE	2.67	0	0

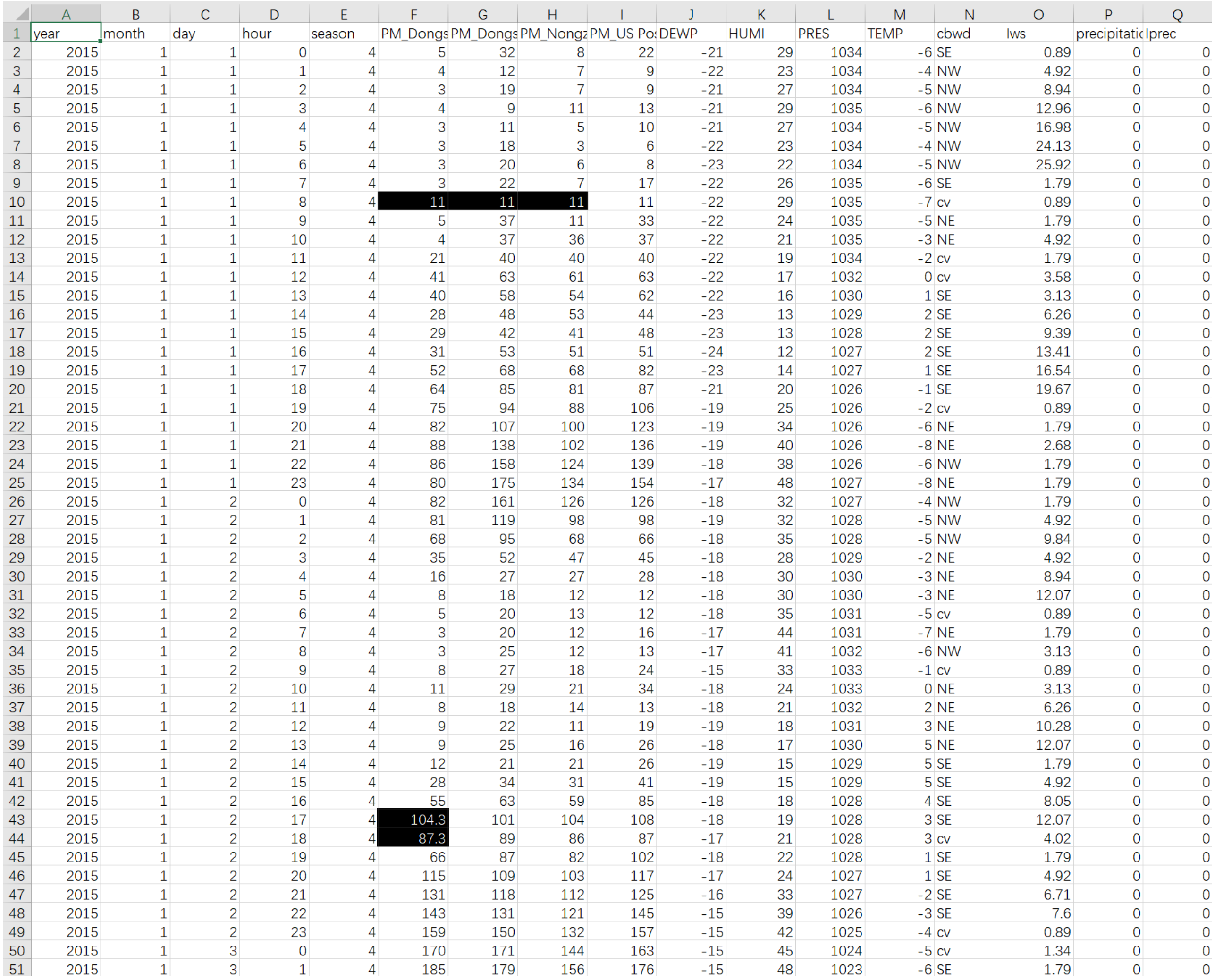


```
2015,12,29,22,4,513.0,491.0,464.0,472.0,-4.0,86.0,1028.0,-2.0,SE,1.79,0.0,0.0
2015,12,29,23,4,475.0,467.0,447.0,470.0,-7.0,92.0,1028.0,-6.0,cv,0.89,0.0,0.0
2015,12,30,0,4,436.0,516.0,486.0,536.0,-7.0,92.0,1028.0,-6.0,NE,1.79,0.0,0.0
2015,12,30,1,4,273.0,551.0,462.0,418.0,-6.0,92.0,1028.0,-5.0,NW,1.79,0.0,0.0
2015,12,30,2,4,138.0,598.0,387.0,460.0,-7.0,92.0,1028.0,-6.0,NE,1.79,0.0,0.0
2015,12,30,3,4,77.0,468.0,275.0,331.0,-7.0,92.0,1028.0,-6.0,NE,3.58,0.0,0.0
2015,12,30,4,4,55.0,366.0,194.0,228.0,-6.0,92.0,1028.0,-5.0,NW,1.79,0.0,0.0
2015,12,30,5,4,32.0,329.0,196.0,173.0,-6.0,85.0,1028.0,-4.0,cv,0.89,0.0,0.0
2015,12,30,6,4,12.0,143.0,20.0,45.0,-6.0,92.0,1029.0,-5.0,NW,1.79,0.0,0.0
2015,12,30,7,4,7.0,32.0,7.0,13.0,-5.0,74.0,1029.0,-1.0,NW,5.81,0.0,0.0
2015,12,30,8,4,12.0,14.0,13.0,10.0,-6.0,85.0,1030.0,-4.0,NE,1.79,0.0,0.0
2015,12,30,9,4,11.0,12.0,15.0,8.0,-6.0,63.0,1031.0,0.0,NW,3.13,0.0,0.0
2015,12,30,10,4,10.0,7.0,8.0,12.0,-7.0,47.0,1031.0,3.0,NW,7.15,0.0,0.0
2015,12,30,11,4,11.0,11.0,14.0,13.0,-11.0,30.0,1031.0,5.0,NW,16.09,0.0,0.0
2015,12,30,12,4,10.0,8.0,10.0,9.0,-11.0,28.0,1031.0,6.0,NW,23.24,0.0,0.0
2015,12,30,13,4,8.0,9.0,9.0,14.0,-11.0,26.0,1030.0,7.0,NW,31.29,0.0,0.0
2015,12,30,14,4,6.0,9.0,11.0,14.0,-11.0,28.0,1030.0,6.0,NW,38.44,0.0,0.0
2015,12,30,15,4,5.0,9.0,12.0,11.0,-11.0,28.0,1030.0,6.0,NW,46.49,0.0,0.0
2015,12,30,16,4,7.0,8.0,7.0,8.0,-11.0,30.0,1030.0,5.0,NW,53.64,0.0,0.0
2015,12,30,17,4,9.0,9.0,12.0,6.0,-11.0,32.0,1031.0,4.0,NW,57.66,0.0,0.0
2015,12,30,18,4,8.0,12.0,13.0,15.0,-11.0,34.0,1031.0,3.0,NW,61.68,0.0,0.0
```

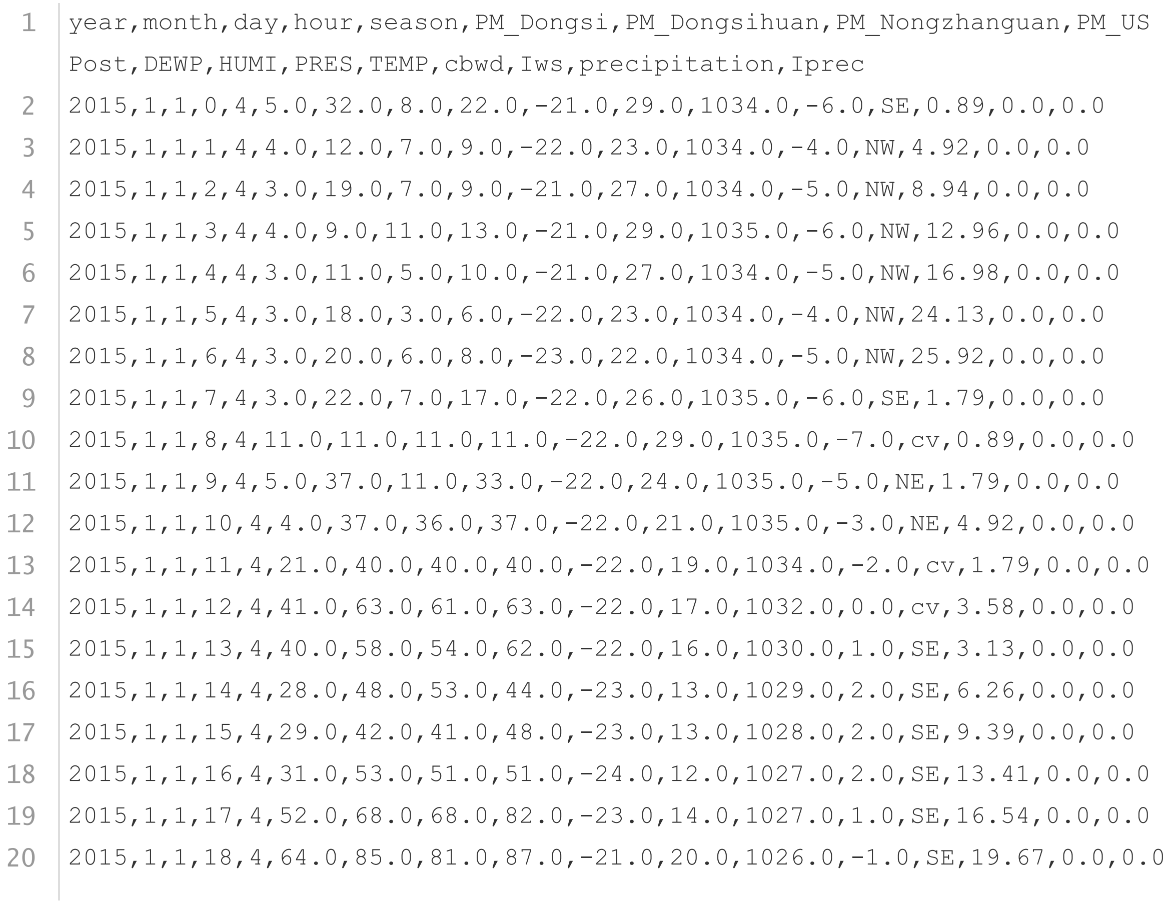
```
2015,12,30,19,4,14.0,21.0,18.0,17.0,-11.0,46.0,1032.0,-1.0,NW,63.47,0.0,0.0
2015,12,30,20,4,27.0,19.0,17.0,20.0,-10.0,54.0,1033.0,-2.0,NW,66.6,0.0,0.0
2015,12,30,21,4,20.0,34.0,22.0,22.0,-10.0,50.0,1034.0,-1.0,NW,70.62,0.0,0.0
2015,12,30,22,4,18.0,35.0,29.0,33.0,-11.0,58.0,1034.0,-4.0,NW,73.75,0.0,0.0
2015,12,30,23,4,37.0,32.0,26.0,26.0,-11.0,53.0,1034.0,-3.0,NE,1.79,0.0,0.0
2015,12,31,0,4,21.0,33.0,25.0,28.0,-11.0,62.0,1034.0,-5.0,NW,1.79,0.0,0.0
2015,12,31,1,4,25.0,34.0,24.0,27.0,-9.0,73.0,1034.0,-5.0,NW,3.58,0.0,0.0
2015,12,31,2,4,25.0,28.0,17.0,24.0,-11.0,73.0,1034.0,-7.0,NW,5.37,0.0,0.0
2015,12,31,3,4,27.0,29.0,18.0,23.0,-11.0,67.0,1034.0,-6.0,NW,8.5,0.0,0.0
2015,12,31,4,4,21.0,33.0,21.0,19.0,-11.0,73.0,1034.0,-7.0,NW,10.29,0.0,0.0
2015,12,31,5,4,15.0,42.0,16.0,14.0,-11.0,73.0,1034.0,-7.0,NW,12.08,0.0,0.0
2015,12,31,6,4,15.0,31.0,16.0,19.0,-12.0,72.0,1034.0,-8.0,NW,15.21,0.0,0.0
2015,12,31,7,4,11.0,26.0,16.0,25.0,-11.0,73.0,1034.0,-7.0,NW,18.34,0.0,0.0
2015,12,31,8,4,12.0,24.0,24.0,22.0,-11.0,67.0,1034.0,-6.0,NW,20.13,0.0,0.0
2015,12,31,9,4,25.0,33.0,26.0,25.0,-8.0,68.0,1035.0,-3.0,NW,23.26,0.0,0.0
2015,12,31,10,4,28.0,,24.0,29.0,-9.0,50.0,1035.0,0.0,NW,26.39,0.0,0.0
2015,12,31,11,4,37.0,,27.0,31.0,-10.0,43.0,1035.0,1.0,NW,28.18,0.0,0.0
2015,12,31,12,4,50.0,,37.0,40.0,-10.0,37.0,1033.0,3.0,cv,0.89,0.0,0.0
2015,12,31,13,4,55.0,,48.0,43.0,-11.0,34.0,1032.0,3.0,NW,1.79,0.0,0.0
2015,12,31,14,4,63.0,,50.0,48.0,-10.0,35.0,1031.0,4.0,SE,1.79,0.0,0.0
2015,12,31,15,4,71.0,61.0,64.0,58.0,-11.0,32.0,1031.0,4.0,SE,3.58,0.0,0.0
2015,12,31,16,4,86.0,75.0,68.0,69.0,-10.0,37.0,1031.0,3.0,SE,4.47,0.0,0.0
2015,12,31,17,4,90.0,102.0,89.0,91.0,-10.0,43.0,1030.0,1.0,SE,5.36,0.0,0.0
2015,12,31,18,4,119.0,117.0,112.0,114.0,-10.0,58.0,1030.0,-3.0,SE,6.25,0.0,0.0
2015,12,31,19,4,140.0,157.0,122.0,133.0,-8.0,68.0,1031.0,-3.0,SE,7.14,0.0,0.0
2015,12,31,20,4,157.0,199.0,149.0,169.0,-8.0,63.0,1030.0,-2.0,SE,8.03,0.0,0.0
2015,12,31,21,4,171.0,231.0,196.0,203.0,-10.0,73.0,1030.0,-6.0,NE,0.89,0.0,0.0
2015,12,31,22,4,204.0,242.0,221.0,212.0,-10.0,73.0,1030.0,-6.0,NE,1.78,0.0,0.0
2015,12,31,23,4,,,,235.0,-9.0,79.0,1029.0,-6.0,NE,2.67,0.0,0.0
```

2.2.3 处理前的2015年csv数据

截取前、后各五十行。

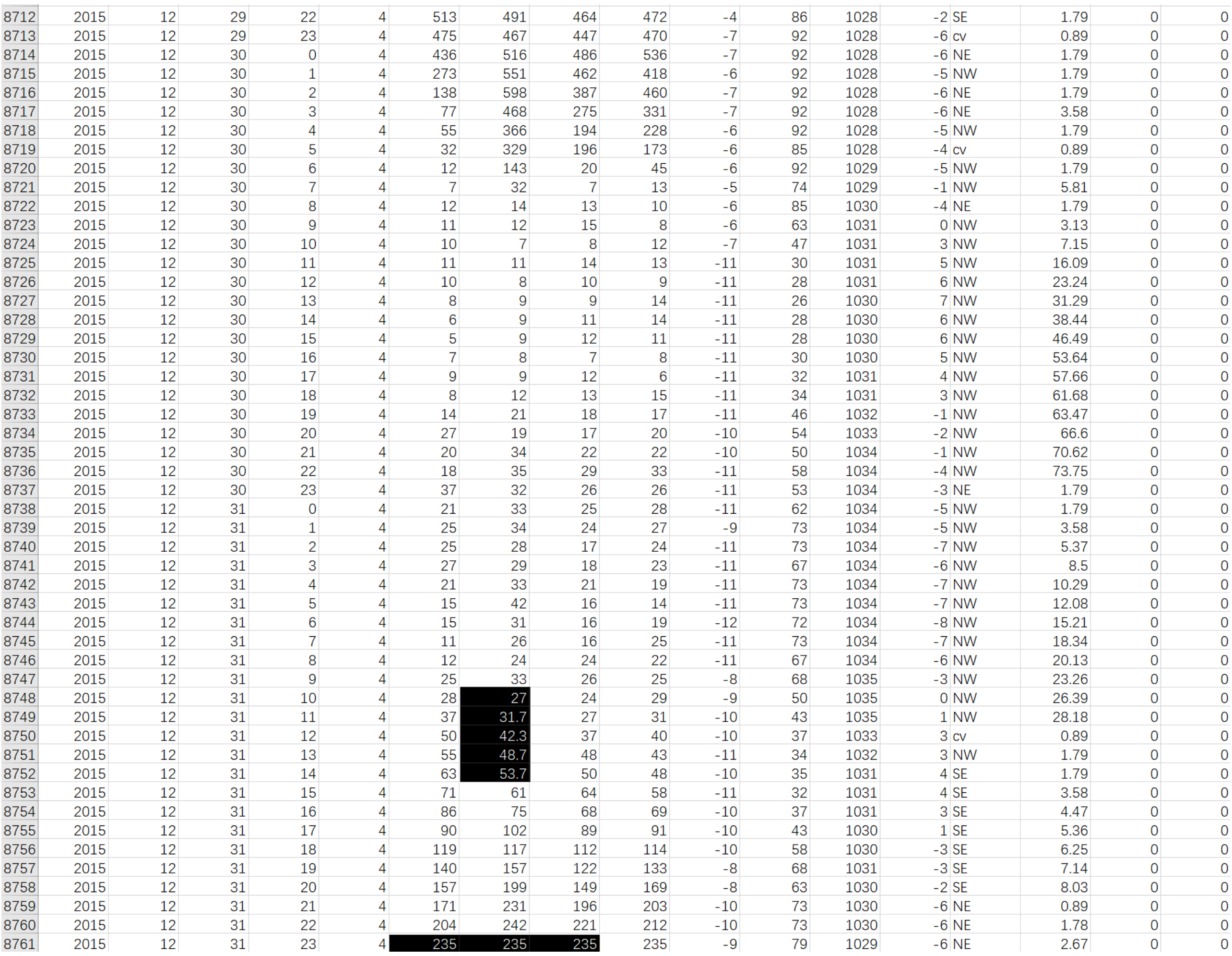


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	year	month	day	hour	season	PM_Dongs	PM_Dongs	PM_Nongz	PM_US Po:	DEWP	HUMI	PRES	TEMP	cbwd	lws	precipitatic	Iprec
2	2015	1	1	0	4	5	32	8	22	-21	29	1034	-6	SE	0.89	0	0
3	2015	1	1	1	4	4	12	7	9	-22	23	1034	-4	NW	4.92	0	0
4	2015	1	1	2	4	3	19	7	9	-21	27	1034	-5	NW	8.94	0	0
5	2015	1	1	3	4	4	9	11	13	-21	29	1035	-6	NW	12.96	0	0
6	2015	1	1	4	4	3	11	5	10	-21	27	1034	-5	NW	16.98	0	0
7	2015	1	1	5	4	3	18	3	6	-22	23	1034	-4	NW	24.13	0	0
8	2015	1	1	6	4	3	20	6	8	-23	22	1034	-5	NW	25.92	0	0
9	2015	1	1	7	4	3	22	7	17	-22	26	1035	-6	SE	1.79	0	0
10	2015	1	1	8	4	11	11	11	11	-22	29	1035	-7	cv	0.89	0	0
11	2015	1	1	9	4	5	37	11	33	-22	24	1035	-5	NE	1.79	0	0
12	2015	1	1	10	4	4	37	36	37	-22	21	1035	-3	NE	4.92	0	0
13	2015	1	1	11	4	21	40	40	40	-22	19	1034	-2	cv	1.79	0	0
14	2015	1	1	12	4	41	63	61	63	-22	17	1032	0	cv	3.58	0	0
15	2015	1	1	13	4	40	58	54	62	-22	16	1030	1	SE	3.13	0	0
16	2015	1	1	14	4	28	48	53	44	-23	13	1029	2	SE	6.26	0	0
17	2015	1	1	15	4	29	42	41	48	-23	13	1028	2	SE	9.39	0	0
18	2015	1	1	16	4	31	53	51	51	-24	12	1027	2	SE	13.41	0	0
19	2015	1	1	17	4	52	68	68	82	-23	14	1027	1	SE	16.54	0	0
20	2015	1	1	18	4	64	85	81	87	-21	20	1026	-1	SE	19.67	0	0
21	2015	1	1	19	4	75	94	88	106	-19	25	1026	-2	cv	0.89	0	0
22	2015	1	1	20	4	82	107	100	123	-19	34	1026	-6	NE	1.79	0	0
23	2015	1	1	21	4	88	138	102	136	-19	40	1026	-8	NE	2.68	0	0
24	2015	1	1	22	4	86	158	124	139	-18	38	1026	-6	NW	1.79	0	0
25	2015	1	1	23	4	80	175	134	154	-17	48	1027	-8	NE	1.79	0	0
26	2015	1	2	0	4	82	161	126	126	-18	32	1027	-4	NW	1.79	0	0
27	2015	1	2	1	4	81	119	98	98	-19	32	1028	-5	NW	4.92	0	0
28	2015	1	2	2	4	68	95	68	66	-18	35	1028	-5	NW	9.84	0	0
29	2015	1	2	3	4	35	52	47	45	-18	28	1029	-2	NE	4.92	0	0
30	2015	1	2	4	4	16	27	27	28	-18	30	1030	-3	NE	8.94	0	0
31	2015	1	2	5	4	8	18	12	12	-18	30	1030	-3	NE	12.07	0	0
32	2015	1	2	6	4	5	20	13	12	-18	35	1031	-5	cv	0.89	0	0
33	2015	1	2	7	4	3	20	12	16	-17	44	1031	-7	NE	1.79	0	0
34	2015	1	2	8	4	3	25	12	13	-17	41	1032	-6	NW	3.13	0	0
35	2015	1	2	9	4	8	27	18	24	-15	33	1033	-1	cv	0.89	0	0
36	2015	1	2	10	4	11	29	21	34	-18	24	1033	0	NE	3.13	0	0
37	2015	1	2	11	4	8	18	14	13	-18	21	1032	2	NE	6.26	0	0
38	2015	1	2	12	4	9	22	11	19	-19	18	1031	3	NE	10.28	0	0
39	2015	1	2	13	4	9	25	16	26	-18	17	1030	5	NE	12.07	0	0
40	2015	1	2	14	4	12	21	21	26	-19	15	1029	5	SE	1.79	0	0
41	2015	1	2	15	4	28	34	31	41	-19	15	1029	5	SE	4.92	0	0
42	2015	1	2	16	4	55	63	59	85	-18	18	1028	4	SE	8.05	0	0
43	2015	1	2	17	4	104.3	101	104	108	-18	19	1028	3	SE	12.07	0	0
44	2015	1	2	18	4	87.3	89	86	87	-17	21	1028	3	cv	4.02	0	0
45	2015	1	2	19	4	66	87	82	102	-18	22	1028	1	SE	1.79	0	0
46	2015	1	2	20	4	115	109	103	117	-17	24	1027	1	SE	4.92	0	0
47	2015	1	2	21	4	131	118	112	125	-16	33	1027	-2	SE	6.71	0	0
48	2015	1	2	22	4	143	131	121	145	-15	39	1026	-3	SE	7.6	0	0
49	2015	1	2	23	4	159	150	132	157	-15	42	1025	-4	cv	0.89	0	0
50	2015	1	3	0	4	170	171	144	163	-15	45	1024	-5	cv	1.34	0	0
51	2015	1	3	1	4	185	179	156	176	-15	48	1023	-6	SE	1.79	0	0

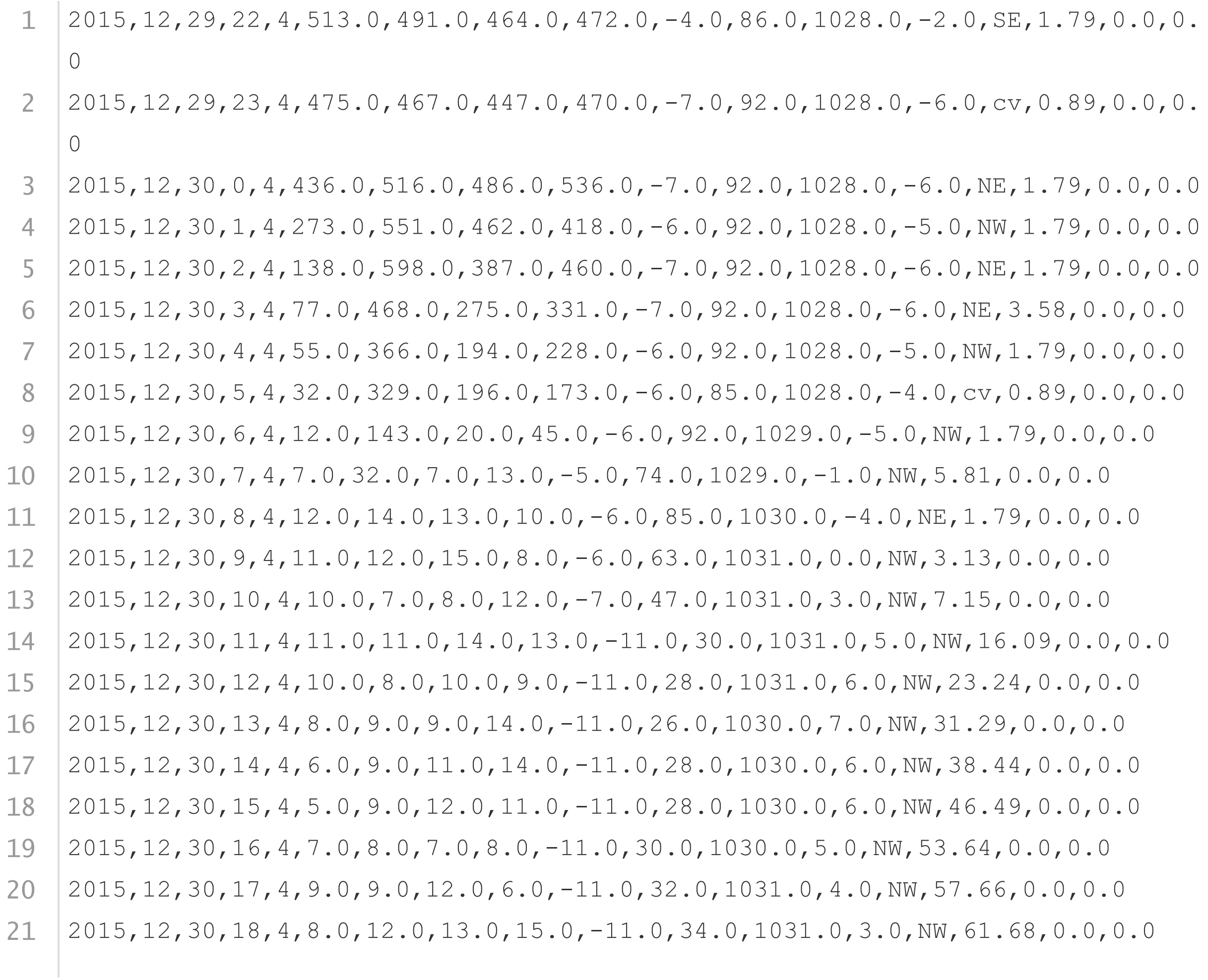


```
year,month,day,hour,season,PM_Dongsi,PM_Dongsihuan,PM_Nongzhanguan,PM_US Post,DEWP,HUMI,PRES,TEMP,cbwd,Iws,precipitation,Iprec
2015,1,1,0,4,5.0,32.0,8.0,22.0,-21.0,29.0,1034.0,-6.0,SE,0.89,0.0,0.0
2015,1,1,1,4,4.0,12.0,7.0,9.0,-22.0,23.0,1034.0,-4.0,NW,4.92,0.0,0.0
2015,1,1,2,4,3.0,19.0,7.0,9.0,-21.0,27.0,1034.0,-5.0,NW,8.94,0.0,0.0
2015,1,1,3,4,4.0,9.0,11.0,13.0,-21.0,29.0,1035.0,-6.0,NW,12.96,0.0,0.0
2015,1,1,4,4,3.0,11.0,5.0,10.0,-21.0,27.0,1034.0,-5.0,NW,16.98,0.0,0.0
2015,1,1,5,4,3.0,18.0,3.0,6.0,-22.0,23.0,1034.0,-4.0,NW,24.13,0.0,0.0
2015,1,1,6,4,3.0,20.0,6.0,8.0,-23.0,22.0,1034.0,-5.0,NW,25.92,0.0,0.0
2015,1,1,7,4,3.0,22.0,7.0,17.0,-22.0,26.0,1035.0,-6.0,SE,1.79,0.0,0.0
2015,1,1,8,4,11.0,11.0,11.0,11.0,-22.0,29.0,1035.0,-7.0,cv,0.89,0.0,0.0
2015,1,1,9,4,5.0,37.0,11.0,33.0,-22.0,24.0,1035.0,-5.0,NE,1.79,0.0,0.0
2015,1,1,10,4,4.0,37.0,36.0,37.0,-22.0,21.0,1035.0,-3.0,NE,4.92,0.0,0.0
2015,1,1,11,4,21.0,40.0,40.0,40.0,-22.0,19.0,1034.0,-2.0,cv,1.79,0.0,0.0
2015,1,1,12,4,41.0,63.0,61.0,63.0,-22.0,17.0,1032.0,0.0,cv,3.58,0.0,0.0
2015,1,1,13,4,40.0,58.0,54.0,62.0,-22.0,16.0,1030.0,1.0,SE,3.13,0.0,0.0
2015,1,1,14,4,28.0,48.0,53.0,44.0,-23.0,13.0,1029.0,2.0,SE,6.26,0.0,0.0
2015,1,1,15,4,29.0,42.0,41.0,48.0,-23.0,13.0,1028.0,2.0,SE,9.39,0.0,0.0
2015,1,1,16,4,31.0,53.0,51.0,51.0,-24.0,12.0,1027.0,2.0,SE,13.41,0.0,0.0
2015,1,1,17,4,52.0,68.0,68.0,82.0,-23.0,14.0,1027.0,1.0,SE,16.54,0.0,0.0
2015,1,1,18,4,64.0,85.0,81.0,87.0,-21.0,20.0,1026.0,-1.0,SE,19.67,0.0,0.0
```

```
2015,1,1,19,4,75.0,94.0,88.0,106.0,-19.0,25.0,1026.0,-2.0,cv,0.89,0.0,0.0
2015,1,1,20,4,82.0,107.0,100.0,123.0,-19.0,34.0,1026.0,-6.0,NE,1.79,0.0,0.0
2015,1,1,21,4,88.0,138.0,102.0,136.0,-19.0,40.0,1026.0,-8.0,NE,2.68,0.0,0.0
2015,1,1,22,4,86.0,158.0,124.0,139.0,-18.0,38.0,1026.0,-6.0,NW,1.79,0.0,0.0
2015,1,1,23,4,80.0,175.0,134.0,154.0,-17.0,48.0,1027.0,-8.0,NE,1.79,0.0,0.0
2015,1,2,0,4,82.0,161.0,126.0,126.0,-18.0,32.0,1027.0,-4.0,NW,1.79,0.0,0.0
2015,1,2,1,4,81.0,119.0,98.0,98.0,-19.0,32.0,1028.0,-5.0,NW,4.92,0.0,0.0
2015,1,2,2,4,68.0,95.0,68.0,66.0,-18.0,35.0,1028.0,-5.0,NW,9.84,0.0,0.0
2015,1,2,3,4,35.0,52.0,47.0,45.0,-18.0,28.0,1029.0,-2.0,NE,4.92,0.0,0.0
2015,1,2,4,4,16.0,27.0,27.0,28.0,-18.0,30.0,1030.0,-3.0,NE,8.94,0.0,0.0
2015,1,2,5,4,8.0,18.0,12.0,12.0,-18.0,30.0,1030.0,-3.0,NE,12.07,0.0,0.0
2015,1,2,6,4,5.0,20.0,13.0,12.0,-18.0,35.0,1031.0,-5.0,cv,0.89,0.0,0.0
2015,1,2,7,4,3.0,20.0,12.0,16.0,-17.0,44.0,1031.0,-7.0,NE,1.79,0.0,0.0
2015,1,2,8,4,3.0,25.0,12.0,13.0,-17.0,41.0,1032.0,-6.0,NW,3.13,0.0,0.0
2015,1,2,9,4,8.0,27.0,18.0,24.0,-15.0,33.0,1033.0,-1.0,cv,0.89,0.0,0.0
2015,1,2,10,4,11.0,29.0,21.0,34.0,-18.0,24.0,1033.0,0.0,NE,3.13,0.0,0.0
2015,1,2,11,4,8.0,18.0,14.0,13.0,-18.0,21.0,1032.0,2.0,NE,6.26,0.0,0.0
2015,1,2,12,4,9.0,22.0,11.0,19.0,-19.0,18.0,1031.0,3.0,NE,10.28,0.0,0.0
2015,1,2,13,4,9.0,25.0,16.0,26.0,-18.0,17.0,1030.0,5.0,NE,12.07,0.0,0.0
2015,1,2,14,4,12.0,21.0,21.0,26.0,-19.0,15.0,1029.0,5.0,SE,1.79,0.0,0.0
2015,1,2,15,4,28.0,34.0,31.0,41.0,-19.0,15.0,1029.0,5.0,SE,4.92,0.0,0.0
2015,1,2,16,4,55.0,63.0,59.0,85.0,-18.0,18.0,1028.0,4.0,SE,8.05,0.0,0.0
2015,1,2,17,4,104.3,101.0,104.0,108.0,-18.0,19.0,1028.0,3.0,SE,12.07,0.0,0.0
2015,1,2,18,4,87.3,89.0,86.0,87.0,-17.0,21.0,1028.0,3.0,cv,4.02,0.0,0.0
2015,1,2,19,4,66.0,87.0,82.0,102.0,-18.0,22.0,1028.0,1.0,SE,1.79,0.0,0.0
2015,1,2,20,4,115.0,109.0,103.0,117.0,-17.0,24.0,1027.0,1.0,SE,4.92,0.0,0.0
2015,1,2,21,4,131.0,118.0,112.0,125.0,-16.0,33.0,1027.0,-2.0,SE,6.71,0.0,0.0
2015,1,2,22,4,143.0,131.0,121.0,145.0,-15.0,39.0,1026.0,-3.0,SE,7.6,0.0,0.0
2015,1,2,23,4,159.0,150.0,132.0,157.0,-15.0,42.0,1025.0,-4.0,cv,0.89,0.0,0.0
2015,1,3,0,4,170.0,171.0,144.0,163.0,-15.0,45.0,1024.0,-5.0,cv,1.34,0.0,0.0
2015,1,3,1,4,185.0,179.0,156.0,176.0,-15.0,48.0,1023.0,-6.0,SE,1.79,0.0,0.0
```



8712	2015	12	29	22	4	513	491	464	472	-4	86	1028	-2	SE	1.79	0	0
8713	2015	12	29	23	4	475	467	447	470	-7	92	1028	-6	cv	0.89	0	0
8714	2015	12	30	0	4	436	516	486	536	-7	92	1028	-6	NE	1.79	0	0
8715	2015	12	30	1	4	273	551	462	418	-6	92	1028	-5	NW	1.79	0	0
8716	2015	12	30	2	4	138	598	387	460	-7	92	1028	-6	NE	1.79	0	0
8717	2015	12	30	3	4	77	468	275	331	-7	92	1028	-6	NE	3.58	0	0
8718	2015	12	30	4	4	55	366	194	228	-6	92	1028	-5	NW	1.79	0	0
8719	2015	12	30	5	4	32	329	196	173	-6	85	1028	-4	cv	0.89	0	0
8720	2015	12	30	6	4	12	143	20	45	-6	92	1029	-5	NW	1.79	0	0
8721	2015	12	30	7	4	7	32	7	13	-5	74	1029	-1	NW	5.81	0	0
8722	2015	12	30	8	4	12	14	13	10	-6	85	1030	-4	NE	1.79	0	0
8723	2015	12	30	9	4	11	12	15	8	-6	63	1031	0	NW	3.13	0	0
8724	2015	12	30	10	4	10	7	8	12	-7	47	1031	3	NW	7.15	0	0
8725	2015	12	30	11	4	11	11	14	13	-11	30	1031	5	NW	16.09	0	0
8726	2015	12	30	12	4	10	8	10	9	-11	28	1031	6	NW	23.24	0	0
8727	2015	12	30	13	4	8	9	9	14	-11	26	1030	7	NW	31.29	0	0
8728	2015	12	30	14	4	6	9	11	14	-11	28	1030	6	NW	38.44	0	0
8729	2015	12	30	15	4	5	9	12	11	-11	28	1030	6	NW	46.49	0	0
8730	2015	12	30	16	4	7	8	7	8	-11	30	1030	5	NW	53.64	0	0
8731	2015	12	30	17	4	9	9	12	6	-11	32	1031	4	NW	57.66	0	0
8732	2015	12	30	18	4	8	12	13	15	-11	34	1031	3	NW	61.68	0	0
8733	2015	12	30	19	4	14	21	18	17	-11	46	1032	-1	NW	63.47	0	0
8734	2015	12	30	20	4	27	19	17	20	-10	54	1033	-2	NW	66.6	0	0
8735	2015	12	30	21	4	20	34	22	22	-10	50	1034	-1	NW	70.62	0	0
8736	2015	12	30	22	4	18	35	29	33	-11	58	1034	-4	NW	73.75	0	0
8737	2015	12	30	23	4	37	32	26	26	-11	53	1034	-3	NE	1.79	0	0
8738	2015	12	31	0	4	21	33	25	28	-11	62	1034	-5	NW	1.79	0	0
8739	2015	12	31	1	4	25	34	24	27	-9	73	1034	-5	NW	3.58	0	0
8740	2015	12	31	2	4	25	28	17	24	-11	73	1034	-7	NW	5.37	0	0
8741	2015	12	31	3	4	27	29	18	23	-11	67	1034	-6	NW	8.5	0	0
8742	2015	12	31	4	4	21	33	21	19	-11	73	1034	-7	NW	10.29	0	0
8743	2015	12	31	5	4	15	42	16	14	-11	73	1034	-7	NW	12.08	0	0
8744	2015	12	31	6	4	15	31	16	19	-12	72	1034	-8	NW	15.21	0	0
8745	2015	12	31	7	4	11	26	16	25	-11	73	1034	-7	NW	18.34	0	0
8746	2015	12	31	8	4	12	24	24	22	-11	67	1034	-6	NW	20.13	0	0
8747	2015	12	31	9	4	25	33	26	25	-8	68	1035	-3	NW	23.26	0	0
8748	2015	12	31	10	4	28	27	24	29	-9	50	1035	0	NW	26.39	0	0
8749	2015	12	31	11	4	37	31.7	27	31	-10	43	1035	1	NW	28.18	0	0
8750	2015	12	31	12	4	50	42.3	37	40	-10	37	1033	3	cv	0.89	0	0
8751	2015	12	31	13	4	55	48.7	48	43	-11	34	1032	3	NW	1.79	0	0
8752	2015	12	31	14	4	63	53.7	50	48	-10	35	1031	4	SE	1.79	0	0
8753	2015	12	31	15	4	71	61	64	58	-11	32	1031	4	SE	3.58	0	0
8754	2015	12	31	16	4	86	75	68	69	-10	37	1031	3	SE	4.47	0	0
8755	2015	12	31	17	4	90	102	89	91	-10	43	1030	1	SE	5.36	0	0
8756	2015	12	31	18	4	119	117	112	114	-10	58	1030	-3	SE	6.25	0	0
8757	2015	12	31	19	4	140	157	122	133	-8	68	1031	-3	SE	7.14	0	0
8758	2015	12	31	20	4	157	199	149	169	-8	63	1030	-2	SE	8.03	0	0
8759	2015	12	31	21	4	171	231	196	203	-10	73	1030	-6	NE	0.89	0	0
8760	2015	12	31	22	4	204	242	221	212	-10	73	1030	-6	NE	1.78	0	0
8761	2015	12	31	23	4	235	235	235	235	-9	79	1029	-6	NE	2.67	0	0



```
2015,12,29,22,4,513.0,491.0,464.0,472.0,-4.0,86.0,1028.0,-2.0,SE,1.79,0.0,0.0
2015,12,29,23,4,475.0,467.0,447.0,470.0,-7.0,92.0,1028.0,-6.0,cv,0.89,0.0,0.0
2015,12,30,0,4,436.0,516.0,486.0,536.0,-7.0,92.0,1028.0,-6.0,NE,1.79,0.0,0.0
2015,12,30,1,4,273.0,551.0,462.0,418.0,-6.0,92.0,1028.0,-5.0,NW,1.79,0.0,0.0
2015,12,30,2,4,138.0,598.0,387.0,460.0,-7.0,92.0,1028.0,-6.0,NE,1.79,0.0,0.0
2015,12,30,3,4,77.0,468.0,275.0,331.0,-7.0,92.0,1028.0,-6.0,NE,3.58,0.0,0.0
2015,12,30,4,4,55.0,366.0,194.0,228.0,-6.0,92.0,1028.0,-5.0,NW,1.79,0.0,0.0
2015,12,30,5,4,32.0,329.0,196.0,173.0,-6.0,85.0,1028.0,-4.0,cv,0.89,0.0,0.0
2015,12,30,6,4,12.0,143.0,20.0,45.0,-6.0,92.0,1029.0,-5.0,NW,1.79,0.0,0.0
2015,12,30,7,4,7.0,32.0,7.0,13.0,-5.0,74.0,1029.0,-1.0,NW,5.81,0.0,0.0
2015,12,30,8,4,12.0,14.0,13.0,10.0,-6.0,85.0,1030.0,-4.0,NE,1.79,0.0,0.0
2015,12,30,9,4,11.0,12.0,15.0,8.0,-6.0,63.0,1031.0,0.0,NW,3.13,0.0,0.0
2015,12,30,10,4,10.0,7.0,8.0,12.0,-7.0,47.0,1031.0,3.0,NW,7.15,0.0,0.0
2015,12,30,11,4,11.0,11.0,14.0,13.0,-11.0,30.0,1031.0,5.0,NW,16.09,0.0,0.0
2015,12,30,12,4,10.0,8.0,10.0,9.0,-11.0,28.0,1031.0,6.0,NW,23.24,0.0,0.0
2015,12,30,13,4,8.0,9.0,9.0,14.0,-11.0,26.0,1030.0,7.0,NW,31.29,0.0,0.0
2015,12,30,14,4,6.0,9.0,11.0,14.0,-11.0,28.0,1030.0,6.0,NW,38.44,0.0,0.0
2015,12,30,15,4,5.0,9.0,12.0,11.0,-11.0,28.0,1030.0,6.0,NW,46.49,0.0,0.0
2015,12,30,16,4,7.0,8.0,7.0,8.0,-11.0,30.0,1030.0,5.0,NW,53.64,0.0,0.0
2015,12,30,17,4,9.0,9.0,12.0,6.0,-11.0,32.0,1031.0,4.0,NW,57.66,0.0,0.0
2015,12,30,18,4,8.0,12.0,13.0,15.0,-11.0,34.0,1031.0,3.0,NW,61.68,0.0,0.0
```

```
2015,12,30,19,4,14.0,21.0,18.0,17.0,-11.0,46.0,1032.0,-1.0,NW,63.47,0.0,0.0
2015,12,30,20,4,27.0,19.0,17.0,20.0,-10.0,54.0,1033.0,-2.0,NW,66.6,0.0,0.0
2015,12,30,21,4,20.0,34.0,22.0,22.0,-10.0,50.0,1034.0,-1.0,NW,70.62,0.0,0.0
2015,12,30,22,4,18.0,35.0,29.0,33.0,-11.0,58.0,1034.0,-4.0,NW,73.75,0.0,0.0
2015,12,30,23,4,37.0,32.0,26.0,26.0,-11.0,53.0,1034.0,-3.0,NE,1.79,0.0,0.0
2015,12,31,0,4,21.0,33.0,25.0,28.0,-11.0,62.0,1034.0,-5.0,NW,1.79,0.0,0.0
2015,12,31,1,4,25.0,34.0,24.0,27.0,-9.0,73.0,1034.0,-5.0,NW,3.58,0.0,0.0
2015,12,31,2,4,25.0,28.0,17.0,24.0,-11.0,73.0,1034.0,-7.0,NW,5.37,0.0,0.0
2015,12,31,3,4,27.0,29.0,18.0,23.0,-11.0,67.0,1034.0,-6.0,NW,8.5,0.0,0.0
2015,12,31,4,4,21.0,33.0,21.0,19.0,-11.0,73.0,1034.0,-7.0,NW,10.29,0.0,0.0
2015,12,31,5,4,15.0,42.0,16.0,14.0,-11.0,73.0,1034.0,-7.0,NW,12.08,0.0,0.0
2015,12,31,6,4,15.0,31.0,16.0,19.0,-12.0,72.0,1034.0,-8.0,NW,15.21,0.0,0.0
2015,12,31,7,4,11.0,26.0,16.0,25.0,-11.0,73.0,1034.0,-7.0,NW,18.34,0.0,0.0
2015,12,31,8,4,12.0,24.0,24.0,22.0,-11.0,67.0,1034.0,-6.0,NW,20.13,0.0,0.0
2015,12,31,9,4,25.0,33.0,26.0,25.0,-8.0,68.0,1035.0,-3.0,NW,23.26,0.0,0.0
2015,12,31,10,4,28.0,27.0,24.0,29.0,-9.0,50.0,1035.0,0.0,NW,26.39,0.0,0.0
2015,12,31,11,4,37.0,31.7,27.0,31.0,-10.0,43.0,1035.0,1.0,NW,28.18,0.0,0.0
2015,12,31,12,4,50.0,42.3,37.0,40.0,-10.0,37.0,1033.0,3.0,cv,0.89,0.0,0.0
2015,12,31,13,4,55.0,48.7,48.0,43.0,-11.0,34.0,1032.0,3.0,NW,1.79,0.0,0.0
2015,12,31,14,4,63.0,53.7,50.0,48.0,-10.0,35.0,1031.0,4.0,SE,1.79,0.0,0.0
2015,12,31,15,4,71.0,61.0,64.0,58.0,-11.0,32.0,1031.0,4.0,SE,3.58,0.0,0.0
2015,12,31,16,4,86.0,75.0,68.0,69.0,-10.0,37.0,1031.0,3.0,SE,4.47,0.0,0.0
2015,12,31,17,4,90.0,102.0,89.0,91.0,-10.0,43.0,1030.0,1.0,SE,5.36,0.0,0.0
2015,12,31,18,4,119.0,117.0,112.0,114.0,-10.0,58.0,1030.0,-3.0,SE,6.25,0.0,0.0
2015,12,31,19,4,140.0,157.0,122.0,133.0,-8.0,68.0,1031.0,-3.0,SE,7.14,0.0,0.0
2015,12,31,20,4,157.0,199.0,149.0,169.0,-8.0,63.0,1030.0,-2.0,SE,8.03,0.0,0.0
2015,12,31,21,4,171.0,231.0,196.0,203.0,-10.0,73.0,1030.0,-6.0,NE,0.89,0.0,0.0
2015,12,31,22,4,204.0,242.0,221.0,212.0,-10.0,73.0,1030.0,-6.0,NE,1.78,0.0,0.0
2015,12,31,23,4,235.0,235.0,235.0,235.0,-9.0,79.0,1029.0,-6.0,NE,2.67,0.0,0.0
```

3 附件：源代码

3.1 作业1: middlewares.py

```python
# Define here the models for your spider middleware
#
# See documentation in:
# https://docs.scrapy.org/en/latest/topics/spider-middleware.html

from scrapy import signals

# useful for handling different item types with a single interface
```

```python
from itemadapter import is_item, ItemAdapter
from scrapy.http import HtmlResponse
from selenium import webdriver
import time

class LianjiaSpiderMiddleware:
    # Not all methods need to be defined. If a method is not defined,
    # scrapy acts as if the spider middleware does not modify the
    # passed objects.

    @classmethod
    def from_crawler(cls, crawler):
        # This method is used by Scrapy to create your spiders.
        s = cls()
        crawler.signals.connect(s.spider_opened, signal=signals.spider_opened)
        return s

    def process_spider_input(self, response, spider):
        # Called for each response that goes through the spider
        # middleware and into the spider.

        # Should return None or raise an exception.
        return None

    def process_spider_output(self, response, result, spider):
        # Called with the results returned from the Spider, after
        # it has processed the response.

        # Must return an iterable of Request, or item objects.
        for i in result:
            yield i

    def process_spider_exception(self, response, exception, spider):
        # Called when a spider or process_spider_input() method
        # (from other spider middleware) raises an exception.

        # Should return either None or an iterable of Request or item objects.
        pass

    def process_start_requests(self, start_requests, spider):
        # Called with the start requests of the spider, and works
        # similarly to the process_spider_output() method, except
        # that it doesn’t have a response associated.

        # Must return only requests (not items).
        for r in start_requests:
            yield r

    def spider_opened(self, spider):
        spider.logger.info('Spider opened: %s' % spider.name)


class LianjiaDownloaderMiddleware:
    # Not all methods need to be defined. If a method is not defined,
    # scrapy acts as if the downloader middleware does not modify the
    # passed objects.

    @classmethod
    def from_crawler(cls, crawler):
        # This method is used by Scrapy to create your spiders.
        s = cls()
        crawler.signals.connect(s.spider_opened, signal=signals.spider_opened)
        return s

    def process_request(self, request, spider):
        # Called for each request that goes through the downloader
        # middleware.
        driver_options = webdriver.ChromeOptions()
        driver_options.add_argument('--headless')
        driver_options.add_argument('--disable-gpu')
        driver_options.add_argument('--window-size=1920,1080')
        driver = webdriver.Chrome(options = driver_options)
        driver.get(request.url)
        driver.implicitly_wait(5)
        page_source = driver.page_source
        try:
            button = driver.find_element_by_class_name('next')
            button.click()
            time.sleep(10)
            next_page = driver.current_url
        except:
            next_page = 'http://none/'
            # 由于HtmlResponse要求url必须为一个合法的url，故我们定义'http://none/'为结束的标志
        driver.quit()
        return HtmlResponse(url=next_page, body=page_source, request=request, encoding='utf-8')

        # Must either:
        # - return None: continue processing this request
        # - or return a Response object
        # - or return a Request object
        # - or raise IgnoreRequest: process_exception() methods of
        #   installed downloader middleware will be called

    def process_response(self, request, response, spider):
        # Called with the response returned from the downloader.

        # Must either;
        # - return a Response object
        # - return a Request object
        # - or raise IgnoreRequest
        return response

    def process_exception(self, request, exception, spider):
        # Called when a download handler or a process_request()
        # (from other downloader middleware) raises an exception.

        # Must either:
        # - return None: continue processing this exception
        # - return a Response object: stops process_exception() chain
        # - return a Request object: stops process_exception() chain
        pass

    def spider_opened(self, spider):
        spider.logger.info('Spider opened: %s' % spider.name)

```

3.2 作业1: lianjia.py

```
import scrapy
from Lianjia.items import LianjiaItem

class LianjiaSpider(scrapy.Spider):
    name = 'lianjia'
    allowed_domains = ['bj.fang.lianjia.com']
    start_urls = ['https://bj.fang.lianjia.com/loupan/']

    def parse(self, response):
        houses = response.xpath('/html/body/div[3]/ul[2]/li')
        for house in houses:
            try:
                item = LianjiaItem()
                item['name'] = house.xpath('./div/div[1]/a/text()').get().strip()
                loc = house.xpath('./div/div[2]')
                item['loc_0'] = loc.xpath('./span[1]/text()').get().strip()
                item['loc_1'] = loc.xpath('./span[2]/text()').get().strip()
                item['loc_2'] = loc.xpath('./a/text()').get().strip()
                item['count'] = house.xpath('./div/a/span[1]/text()').get().strip()
```

```python
                area = house.xpath('./div/div[3]/span/text()').get()
                area = area.split(' ')[1].split('-')[0].rstrip('㎡')
                item['area'] = int(area)
                flag = house.xpath('./div/div[6]/div[1]/span[2]/text()').get().strip()
                if flag == '元/㎡(均价)':
                    unit_price = house.xpath('./div/div[6]/div[1]/span[1]/text()').get()
                    unit_price = unit_price.split('-')[0].strip()
                    item['unit_price'] = int(unit_price)
                    item['total_price'] = item['unit_price'] * item['area'] / 10000
                else:
                    total_price = house.xpath('./div/div[6]/div[1]/span[1]/text()').get()
                    total_price = total_price.split('-')[0].strip()
                    item['unit_price'] = 0
                    item['total_price'] = float(total_price)
                    item['unit_price'] = round(item['total_price'] / item['area'] * 10000)
                item['total_price'] = '{:.4f}'.format(item['total_price'])
                yield item
            except:
                self.logger.info('Recieved an item containing null values')

        self.logger.info(f'Recieved next_page: {response.url}')
        if response.url == 'http://none/':
            # 由于HtmlResponse要求url必须为一个合法的url，故我们定义'http://none/'为结束的标志
            pass
        else:
            yield scrapy.Request(response.url)
```

3.3 作业1: process.py

```python
import pandas as pd

pd.set_option('display.unicode.east_asian_width', True)
data = pd.read_csv('./loupan.csv', encoding='utf-8-sig')

print('--------------------')
print('总价最贵的房子为：')
totalmax_id = data.loc[:, '总价'].idxmax()
print(data.loc[totalmax_id])

print('--------------------')
print('总价最便宜的房子为：')
totalmin_id = data.loc[:, '总价'].idxmin()
```

```python
print(data.loc[totalmin_id])

print('--------------------')
print('总价的中位数：')
totalmin_id = data.loc[:, '总价'].idxmin()
print('{:.4f}'.format(data.loc[:, '总价'].median()))

print('--------------------')
print('均价最贵的房子为：')
totalmax_id = data.loc[:, '均价'].idxmax()
print(data.loc[totalmax_id])

print('--------------------')
print('均价最便宜的房子为：')
totalmin_id = data.loc[:, '均价'].idxmin()
print(data.loc[totalmin_id])

print('--------------------')
print('单价的中位数：')
totalmin_id = data.loc[:, '均价'].idxmin()
print('{:.4f}'.format(data.loc[:, '均价'].median()))

print('--------------------')
print('总价在均值三倍标准差以外的异常值：')
down = data['总价'].mean() - 3 * data['总价'].std()
up = data['总价'].mean() + 3 * data['总价'].std()
print(data.loc[(data['总价'] < down) | (data['总价'] > up)])

print('--------------------')
print('均价在箱型图原则下（k = 1.5）的异常值：')
k = 1.5
q1 = data['均价'].quantile(q=0.25)
q3 = data['均价'].quantile(q=0.75)
down = q1 - k * (q3 - q1)
up = q3 + k * (q3 - q1)
print(data.loc[(data['均价'] < down) | (data['均价'] > up)])

print('--------------------')
print('均价离散化处理：')
avgs = [0, 20000, 40000, 60000, 80000, 100000, 120000, 140000, 160000, 180000]
cuts = pd.cut(data['均价'], avgs)
calc = pd.value_counts(cuts).to_frame()
total = calc.iloc[:, 0].sum()
calc['百分比'] = 100 * calc.iloc[:, 0] / total
calc.sort_index(axis=0, ascending=True, inplace=True)
print(calc)
print('--------------------')
```

3.4 作业2: process.py

```python
import pandas as pd

pd.set_option('display.unicode.east_asian_width', True)
data = pd.read_csv('./BeijingPM20100101_20151231.csv', encoding='utf-8')
# 数据抽取及存储
data2015 = data.loc[data['year']==2015].drop(columns=['No'])
data2015.to_csv('./BeijingPM2015.csv', index=False, encoding='utf-8')

data = pd.read_csv('./BeijingPM2015.csv', encoding='utf-8')
print('--------------------')
print('存在的空值列：')
# 找出存在空值的列
print(data.isnull().any())

print('--------------------')
print('列对应的空值数量：')
# 找出对应列空值数量
print(data.isnull().sum(axis=0))

print('--------------------')
columns = ['PM_Dongsi', 'PM_Dongsihuan', 'PM_Nongzhanguan', 'PM_US Post']
# 计算每一行四个pm2.5监测点的平均值
meanpm = round(data[columns].mean(axis=1), 1)
# 创建一个四个监测点名称到平均值的映射
fill = {}.fromkeys(columns, meanpm)
# 使用映射，将四个监测点的平均值替换某些监测点的空值
data.fillna(value=fill, inplace=True)

# 其他数据使用上一行的非空值填充
data.fillna(method='ffill', inplace=True)
print(data)

print('--------------------')
print('空值处理后存在的空值列：')
print(data.isnull().any())

print('--------------------')
print('空值处理后列对应的空值数量：')
print(data.isnull().sum(axis=0))
print('--------------------')

data.to_csv('./BeijingPM2015_cleaned.csv', index=False, encoding='utf-8')
```