1. **问题描述**
2. 通过爬虫爬取链家的新房数据，并进行预处理。

https://bj.fang.lianjia.com/loupan/

• 最终的csv文件，应包括以下字段：名称，地理位置（3个字段分别存储），房型（只保留最小房型），面积（按照最小值），总价（万元，整数），均价（万元，保留小数点后4位）；

• 对于所有字符串字段，要求去掉所有的前后空格；

• 如果有缺失数据，不用填充。

• 找出总价最贵和最便宜的房子，以及总价的中位数

• 找出单价最贵和最便宜的房子，以及单价的中位数

1. 计算北京空气质量数据

1. 汇总计算PM指数年平均值的变化情况

2. 汇总计算10-15年PM指数和温度月平均数据的变化情况

1. **实验环境**

Microsoft Windows 10 版本18363

PyCharm 2020.2.1 (Community Edition)

Python 3.8(Scrapy 2.4.0 + numpy 1.19.4 + pandas 1.1.4)

1. **实验步骤及结果**
2. 通过爬虫爬取链家的新房数据，并进行预处理。

第一步为爬取数据，上一个实验已经处理过，故只贴出spider部分的代码。

import scrapy  
from loupan.items import LoupanItem  
  
class lpSpider(scrapy.Spider):  
 name = "lp"  
 allowed\_domains = ["lianjia.com/loupan"]  
 start\_urls = []  
 for i in range(1, 8):  
 url = 'https://bj.fang.lianjia.com/loupan/pg' + str(i)  
 start\_urls.append(url)  
  
 def parse(self, response):  
 item = LoupanItem()  
  
 for each in response.xpath('//div[@class="resblock-desc-wrapper"]'):  
 item['name'] = each.xpath('div[1]/a/text()').extract()  
 item['location'] = each.xpath('div[2]/span[1]/text()').extract() + each.xpath('div[2]/span[2]/text()').extract() + each.xpath('div[2]/a/text()').extract()  
 item['type'] = each.xpath('a[1]/span[1]/text()').extract()  
 item['area'] = each.xpath('div[3]/span[1]/text()').extract()  
 #item['totalPrice'] = each.xpath('div[6]/div[2]/text()').extract()  
 item['totalPrice'] = ""  
 item['evenPrice'] = each.xpath('div[6]/div[1]/span[1]/text()').extract()  
 if item['name'] and item['area']:  
 yield item

第二步为数据预处理，主要使用numpy转换爬取的数据格式，并寻找最大值、最小值、中位数等，并使用pandas重新将处理后的数据制表。

import numpy as np  
import pandas as pd  
  
fileNameStr = 'result.csv'  
orig\_df = pd.read\_csv(fileNameStr, encoding='utf-8', dtype=str)  
num = len(orig\_df)  
  
orig\_df['area'] = orig\_df['area'].str.strip()  
orig\_df['evenPrice'] = orig\_df['evenPrice'].str.strip()  
orig\_df['location'] = orig\_df['location'].str.strip()  
orig\_df['name'] = orig\_df['name'].str.strip()  
orig\_df['totalPrice'] = orig\_df['totalPrice'].str.strip()  
orig\_df['type'] = orig\_df['type'].str.strip()  
  
for i in range(num):  
 orig\_df.area[i] = orig\_df.area[i].replace("建面 ", "")  
 idx = orig\_df.area[i].find("-")  
 if idx == -1:  
 idx = len(orig\_df.area[i]) - 1  
 orig\_df.area[i] = orig\_df.area[i][0:idx]  
  
orig\_df['area'] = orig\_df['area'].astype(np.float).astype(int)  
orig\_df['evenPrice'] = orig\_df['evenPrice'].astype(np.float)  
orig\_df['totalPrice'] = orig\_df['totalPrice'].astype(np.float)  
df = orig\_df['location'].str.split(',', expand=True)  
  
for i in range(num):  
 orig\_df.totalPrice[i] = round(orig\_df.area[i] \* orig\_df.evenPrice[i] / 10000, 4)  
  
result = pd.concat([orig\_df['name'], df, orig\_df['type'], orig\_df['area'], orig\_df['evenPrice'], orig\_df['totalPrice']], axis=1, ignore\_index=True)  
result.columns = ['name', 'location1', 'location2', 'location3', 'location4', 'location5', 'type', 'area', 'evenPrice', 'totalPrice']  
result.sort\_values(by='evenPrice', ascending=False, inplace=True)  
  
result.to\_csv("result-handled.csv", encoding="utf-8")  
  
print(result.describe())



以下为找出总价、单价的最大、最小、中位数所对应的房源：

print("总价最高：", end="")  
print(orig\_df['totalPrice'].max())  
print(orig\_df.iloc[orig\_df['totalPrice'].idxmax()].values.tolist())  
print("总价最低：", end="")  
print(orig\_df['totalPrice'].min())  
print(orig\_df.iloc[orig\_df['totalPrice'].idxmin()].values.tolist())  
print("总价中位数：", end="")  
print(orig\_df['totalPrice'].median())  
print("单价最高：", end="")  
print(orig\_df['evenPrice'].max())  
print(orig\_df.iloc[orig\_df['evenPrice'].idxmax()].values.tolist())  
print("单价最低：", end="")  
print(orig\_df['evenPrice'].min())  
print(orig\_df.iloc[orig\_df['evenPrice'].idxmin()].values.tolist())  
print("单价中位数：", end="")  
print(orig\_df['evenPrice'].median())

总价最高：5400.0

[540, 100000.0, '朝阳,北苑,北京市朝阳区北五环顾家庄桥向北约2.6公里', '润泽御府', 5400.0, '4室']

总价最低：181.3

[74, 24500.0, '通州,通州其它,永乐店镇漷小路百菜玛工业园对面', 'K2十里春风', 181.3, '2室']

总价中位数：694.075

单价最高：120000.0

[146, 120000.0, '朝阳,望京,望京南湖南路三帆中学对面', '首开国风尚樾', 1752.0, '3室']

单价最低：21000.0

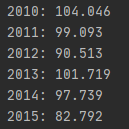
[110, 21000.0, '密云,密云其它,密关路与313省道交汇处', '奥园北京源墅', 231.0, '3室']

单价中位数：52000.0

1. 计算北京空气质量数据
2. 汇总计算PM指数年平均值的变化情况

import numpy as np  
import pandas as pd  
  
fileNameStr = './Beijing.csv'  
df = pd.read\_csv(fileNameStr, encoding='utf-8', dtype=str)  
num = len(df)  
  
sum = 0  
cnt = 0  
nowyear = 2010  
for i in range(num):  
 if int(df['year'][i]) == nowyear:  
 if not df['PM\_US Post'][i] is np.nan:  
 sum = sum + int(df['PM\_US Post'][i])  
 cnt = cnt + 1  
 else:  
 print("%d: %.3f" % (nowyear, sum/cnt))  
 sum = 0  
 cnt = 0  
 nowyear = nowyear + 1  
  
print("%d: %.3f" % (nowyear, sum/cnt))

得到结果：



1. 汇总计算10-15年PM指数和温度月平均数据的变化情况

sumpm = 0  
sumtemp = 0  
cnt = 0  
nowyear = 2010  
nowmonth = 1  
for i in range(num):  
 if int(df['year'][i]) == nowyear and float(df['month'][i]) == nowmonth:  
 if (not df['PM\_US Post'][i] is np.nan) and (not df['TEMP'][i] is np.nan):  
 sumpm = sumpm + int(df['PM\_US Post'][i])  
 sumtemp = sumtemp + float(df['TEMP'][i])  
 cnt = cnt + 1  
 else:  
 print("%d.%d: pm=%.3f, temp=%.3f" % (nowyear, nowmonth, sumpm/cnt, sumtemp/cnt))  
 sumpm = 0  
 sumtemp = 0  
 cnt = 0  
 nowyear = int(df['year'][i])  
 nowmonth = int(df['month'][i])  
  
print("%d.%d: pm=%.3f, temp=%.3f" % (nowyear, nowmonth, sumpm/cnt, sumtemp/cnt))

2010.1: pm=90.404, temp=-6.372

2010.2: pm=97.281, temp=-1.913

2010.3: pm=94.096, temp=3.006

2010.4: pm=80.130, temp=10.810

2010.5: pm=87.095, temp=20.860

2010.6: pm=108.918, temp=24.505

2010.7: pm=123.326, temp=27.734

2010.8: pm=97.796, temp=25.348

2010.9: pm=122.831, temp=22.242

2010.10: pm=118.583, temp=12.283

2010.11: pm=138.468, temp=3.306

2010.12: pm=96.902, temp=-2.062

2011.1: pm=44.874, temp=-5.544

2011.2: pm=150.450, temp=-0.849

2011.3: pm=58.071, temp=6.979

2011.4: pm=91.478, temp=14.696

2011.5: pm=65.008, temp=20.738

2011.6: pm=108.924, temp=25.651

2011.7: pm=107.188, temp=26.461

2011.8: pm=103.734, temp=25.664

2011.9: pm=94.968, temp=19.217

2011.10: pm=145.676, temp=13.208

2011.11: pm=109.234, temp=5.940

2011.12: pm=108.667, temp=-2.307

2012.1: pm=118.647, temp=-4.939

2012.2: pm=84.544, temp=-2.563

2012.3: pm=96.208, temp=5.077

2012.4: pm=87.879, temp=15.478

2012.5: pm=90.726, temp=21.945

2012.6: pm=96.699, temp=24.339

2012.7: pm=80.655, temp=26.560

2012.8: pm=81.273, temp=25.556

2012.9: pm=59.748, temp=20.107

2012.10: pm=94.999, temp=13.311

2012.11: pm=87.369, temp=3.694

2012.12: pm=109.290, temp=-4.347

2013.1: pm=193.612, temp=-5.374

2013.2: pm=123.585, temp=-1.822

2013.3: pm=123.541, temp=5.408

2013.4: pm=65.555, temp=12.246

2013.5: pm=85.223, temp=21.485

2013.6: pm=111.473, temp=23.662

2013.7: pm=68.720, temp=27.062

2013.8: pm=61.969, temp=26.590

2013.9: pm=90.933, temp=20.124

2013.10: pm=106.499, temp=12.753

2013.11: pm=90.427, temp=5.921

2013.12: pm=98.603, temp=-0.346

2014.1: pm=118.863, temp=-0.933

2014.2: pm=174.585, temp=-0.714

2014.3: pm=110.369, temp=9.562

2014.4: pm=95.050, temp=16.825

2014.5: pm=72.206, temp=21.601

2014.6: pm=59.036, temp=24.936

2014.7: pm=89.685, temp=28.008

2014.8: pm=62.691, temp=25.740

2014.9: pm=70.223, temp=20.524

2014.10: pm=140.768, temp=13.338

2014.11: pm=104.071, temp=5.752

2014.12: pm=78.669, temp=-1.368

2015.1: pm=108.031, temp=-1.341

2015.2: pm=96.713, temp=0.907

2015.3: pm=89.311, temp=7.828

2015.4: pm=78.865, temp=15.613

2015.5: pm=59.858, temp=21.530

2015.6: pm=54.336, temp=24.700

2015.7: pm=55.120, temp=26.570

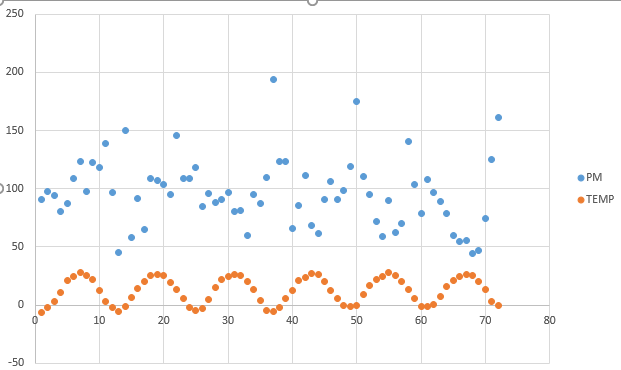
2015.8: pm=44.650, temp=25.823

2015.9: pm=47.142, temp=20.394

2015.10: pm=74.654, temp=13.770

2015.11: pm=124.836, temp=2.896

2015.12: pm=161.469, temp=-0.614



可以直观地看到温度和PM均有周期性变化的形势。

且PM与TEMP有一定负相关关系。即当温度低时，PM浓度较高；温度高时，PM浓度较低。且该种负相关关系越来越明显。

且每年温度的最低值有上升的趋势。

且每年PM的平均值有下降的趋势。

1. **心得与体会**

在本次实验中，通过两道题目的引导，我了解到了python中利用numpy和pandas进行数据分析及处理的基本方法，对工具中常用的函数有了基本的认识和理解。非常感谢老师的实验设计！