程设第二次作业

20377383 樊思涵

前言：看了同学们的展示作业，意识到自己的作业有很多不足，这次作业决定学习展示作业中函数注释的写法，改变代码风格。

任务 1.

准备活动：

调用的库：

```
import numpy as np
import jieba
import re
from matplotlib import pyplot as plt

```

将需要用到的文件地址作为全局变量

```
#文件名用作全局变量
filename_0=r'C:\Users\LF\Desktop\weibo.txt\weibo.txt'
filename_0_test=r'C:\Users\LF\Desktop\weibo.txt\test_weibo.txt'
filename_anger=r'C:\Users\LF\Desktop\emotion_lexicon\anger.txt'
filename_disgust=r'C:\Users\LF\Desktop\emotion_lexicon\disgust.txt'
filename_fear=r'C:\Users\LF\Desktop\emotion_lexicon\fear.txt'
filename_joy=r'C:\Users\LF\Desktop\emotion_lexicon\joy.txt'
filename_sadness=r'C:\Users\LF\Desktop\emotion_lexicon\sadness.txt'
filename_emotions=[filename_anger,filename_disgust,filename_fear,filename_joy,filename_sadness]

```

补充：

```
EMO=['anger','disgust','fear','joy','sadness']
TIMEMODE=['week','hour','month']
```

其中，test_weibo.txt 为 473 条小样本量测试用微博数据

按行读入文件

```
def fread_document(filename) -> list:
    """
    按行读入txt文件并返回列表
    ->返回一维列表
    :filename:原始文档目录
    """
    print("----------正在导入数据----------")
    f=open(filename,encoding='UTF-8')
    line = f.readline().strip() #读取第一行,不用读入列表
    txt=[]
    while line:  # 直到读取完文件
        line = f.readline().strip()  # 读取一行文件，包括换行符
        txt.append(line)
    f.close()  # 关闭文件
    if txt[-1]=='':
        txt.pop()
    print("----------数据导入完成----------")
    return txt
```

处理重复微博

```python
def fdelete_repetition_txt(txt) -> list:
    """
    删除列表中的重复元素
    ->返回与原始列表相同结构的列表
    :txt:需删除重复元素的列表
    """
    if len(txt) != len(set(txt)):
        print("有重复微博")
        txt = [x for x in set(txt)]
    return txt
```

展示结果

```
PS E:\code\py_code> python -u "e:\code\py_code\week3\week3.py"
----------正在导入数据----------
----------数据导入完成----------
有重复微博
处理前数据有2422485项.
处理重复数据后有272883项.
----------正在添加情绪词----------
```

没想到这么多重复数据！

对数据进行分割降噪处理
Step1.对\t 分割，得到 column = 4 的列表

```python
def fcut_txt(txt,n) -> list:
    """
    将按行读入的数据按'\t'切割
    ->返回column=4的二维列表
    :txt:按行读入的数据列表
    :n:txt的长度
    """
    data=[]
    for i in range(n):
        data.append(txt[i].split("\t"))
    return data
```

Step2.删除每条微博后都有的”我在这里：xxxxx”

```python
def fdelete_url_data(data,n) -> list:
    """
    删除每条微博后相同的url地址
    ->返回与data相同的column=4的二位列表
    :data:切割好的二维列表
    :n:二维列表长度
    """
    sentence=[]    #专门用来储存微博内容
    #根据字符串格式切割
    for i in range(n):
        #print(data[i][1])
        sentence.append(data[i][1].rsplit(' ',1)[0])
    ##print(sentence)
    return sentence

```

Step3.使用正则表达式删除一些无意义的文本

```python
def fclean_sentence(sentence,n) -> list:
    """
    使用正则表达式对文本降噪
    ->返回与sentence相同的一维列表
    :sentence:
    :n:列表长度
    """
    for i in range(n):
        sentence[i] = re.sub(r"(回复)?(//)?\s*@\S*?\s*(:| |$)"," ",sentence[i])  # 去除正文中的@和回复/转发中的用户名
        sentence[i] = re.sub(r'[\S]+\.(cn|net|com|org|info|edu|gov|uk|de|ca|jp|fr|au|us|ru|ch|it|nel|se|no|es|mil)[\S]*\s?','',sentence[i])# 去除网址
        sentence[i] = re.sub(r"\s+", " ", sentence[i]) # 合并正文中过多的空格
    return sentence
```

过滤效果如下：

```
--------------------
原文本:[39.87458, 116.499259]  我们奋斗，我们努力，我们拼命，为的不就是未来能给自己，给家人，给我们的女人一个交代么？二十多岁的时候，我们奋斗过，努力过，不曾后悔，不会后悔！ 我在:http://t.cn/zRb0MGk 2922414950.0    Fri Oct 11 08:40:55 +0800 2013
处理后文本:我们奋斗，我们努力，我们拼命，为的不就是未来能给自己，给家人，给我们的女人一个交代么？二十多岁的时候，我们奋斗过，努力过，不曾后悔，不会后悔！
--------------------
原文本:[39.87371, 116.5]      每天一片保湿面膜，秋季强烈补湿，皮肤好状态要坚持几天哦 🍓🍓🍓 我在:http://t.cn/z8muVQT       2132134765.0    Fri Oct 11 23:28:22 +0800 2013
处理后文本:每天一片保湿面膜，秋季强烈补湿，皮肤好状态要坚持几天哦 🍓🍓🍓
--------------------
原文本:[39.881402, 116.545022]  "@civo 带我遨历世界。用邀请码""7CUN""注册，http://t.cn/zYem2n9 ，我们将成为好友，还送你10张船票。"     3837449589.0    Fri Oct 11 23:58:07 +0800 2013
处理后文本:"
--------------------
原文本:[39.886351, 116.564568]  好想苗苗，想和苗苗玩，想让苗苗亲亲我，想听苗苗叫我干妈[太开心][太开心][太开心]看看我拍的视频都要亲上去了，@神梨子 怎么办太萌了！ 我在:http://t.cn/zRGCARE      1739650674.0    Fri Oct 11 23:52:39 +0800 2013
处理后文本:好想苗苗，想和苗苗玩，想让苗苗亲亲我，想听苗苗叫我干妈[太开心][太开心][太开心]看看我拍的视频都要亲上去了， 怎么办太萌了！
--------------------
原文本:[39.8743335, 116.6536015]      时间一天一天过的好快！！提前预祝我自己生日快乐！[蛋糕][蛋糕][蛋糕] 我在:http://t.cn/zRG7hDf     2494530527.0    Fri Oct 11 12:42:44 +0800 2013
处理后文本:时间一天一天过的好快！！提前预祝我自己生日快乐！[蛋糕][蛋糕][蛋糕]
--------------------
原文本:[39.88377, 116.66844]    对你，不知道怎么说，只是希望可以做永远的好朋友哦！呵呵！ 我在这里:http://t.cn/z8AUe0V   3805590882.0    Fri Oct 11 23:59:40 +0800 2013
处理后文本:对你，不知道怎么说，只是希望可以做永远的好朋友哦！呵呵！
--------------------
原文本:[39.88277, 116.668442]   552撒以后的火影忍者片头曲怎么听起来悲宠窜台到了"类少女战士"[黑线]一点也不符合热血派的火影风格～[哼] 我在这里:http://t.cn/z8AUe0V       2069782294.0    Fri Oct 11 20:40:11 +0800 2013
处理后文本:552撒以后的火影忍者片头曲怎么听起来悲宠窜台到了"类少女战士"[黑线]一点也不符合热血派的火影风格～[哼]
--------------------
原文本:[39.88377, 116.66844]    哦，可悲的我，可悲的不被赞同的恋爱，可悲的只有两个女儿的父母，可悲的我还有一个远嫁的姐姐。 我在这里:http://t.cn/z8AUe0V 2648658580.0    Fri Oct 11 01:22:20 +0800 2013
处理后文本:哦，可悲的我，可悲的不被赞同的恋爱，可悲的只有两个女儿的父母，可悲的我还有一个远嫁的姐姐。
--------------------
原文本:[39.88377, 116.66844]    四翼妈说，你下班回家等我，待会我找你玩去啦，我浪痛快的跳舞应了，现在糟糟郝打呼呼了，四翼妈你还没来，你退拣呢是吧。[哼] 我在这里:http://t.cn/z8AUe0V   3170273287.0    Fri Oct 11 21:30:23 +0800 2013
处理后文本:四翼妈说，你下班回家等我，待会我找你玩去啦，我浪痛快的跳舞应了，现在糟糟郝打呼呼了，四翼妈你还没来，你退拣呢是吧。[哼]
--------------------
原文本:[39.882221, 116.66896]  晚上就到手了  无比的happy[哈哈][哈哈][哈哈] 我在:http://t.cn/zQjD4tb    2551411957.0    Fri Oct 11 15:45:05 +0800 2013
处理后文本:晚上就到手了 无比的happy[哈哈][哈哈][哈哈]
--------------------
```

分析：在删除无意义文本时，不能删除所有的表情，因为在情绪分词中记录了表情，不同表情显然能够表达不同的情绪。

任务 2.
在分词表中添加情绪词

```python
def faddword(filename_emotions):
    """
    jieba添加自定义字典
    """
    print("----------正在添加情绪词----------")
    for i in filename_emotions:
        jieba.load_userdict(i)

```

创建闭包函数
取出现次数最多的 emotion 为对应微博的情绪

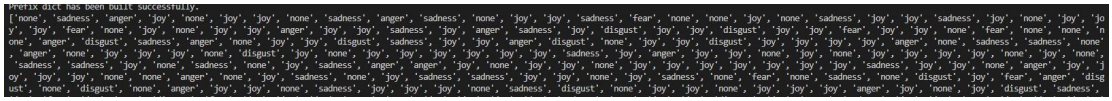
```python
def fdic_emotion(filename_emotions,data,sentence,n):
    """
    闭包函数返回情绪，时间，地址列表函数
    ->函数名
    :filename_emotions:情绪地址列表
    :data:column=4的全信息列表
    :sentence:处理后的文本字符
    :n:列表长度
    """
    emodict = []
    for i in filename_emotions:
        file = open(i,'r',encoding='utf-8')
        emodict.append([line.strip() for line in file.readlines()])
        file.close()
    def splitword():  #分词获取情绪以及对应的时间地点
        nonlocal emodict
        emotion_list,time_list,address_list = [],[],[]
        for i in range(n):
            emotion_dict = {'anger':0,'disgust':0,'fear':0,'joy':0,'sadness':0}
            splitword = jieba.lcut(sentence[i])
            #print(splitword)
```

```
            for word in splitword:
                if word in emodict[0]:
                    emotion_dict['anger']+=1
                elif word in emodict[1]:
                    emotion_dict['disgust']+=1
                elif word in emodict[2]:
                    emotion_dict['fear']+=1
                elif word in emodict[3]:
                    emotion_dict['joy'] +=1
                elif word in emodict[4]:
                    emotion_dict['sadness']+=1
            if max(emotion_dict.values())==0:
                emotion = 'none'
            else:
                emotion = max(emotion_dict,key=emotion_dict.get)
            emotion_list.append(emotion)
            time_list.append(data[i][3])
            address_list.append(data[i][0])
        print(emotion_list,end = '')
        return emotion_list,time_list,address_list
    return splitword

```

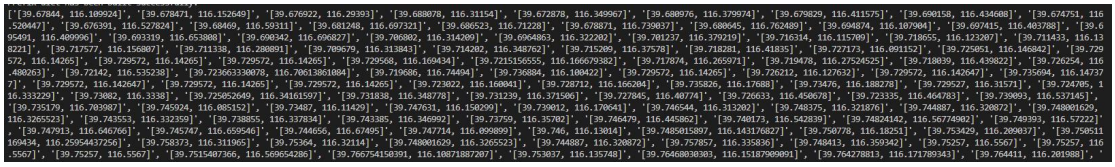
结果展示（用测试数据文本）

Emotion_list



time_list

address_list



任务 3.

创建小时、星期、月份字典，统计每条的属性

```python
def fplotime(emotion,time_mode,emotion_list,time_list):
    """
    生成指定情绪指定模式的情绪统计图
    :emotion:目标的情绪的字符串
    :time_mode:目标的时间图表模式
    :emotion_list:微博情绪列表
    :time_list:微博时间列表
    """
    week = ['Mon','Tue','Wed','Thu','Fri','Sat','Sun']
    week_dict = {}
    week_dict = week_dict.fromkeys(week,0)

    month = ['Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sep','Oct','Nov','Dec']
    month_dict = {}
    month_dict = month_dict.fromkeys(month,0)

    hour = ['{:0>2d}'.format(i) for i in range(24)]
    hour_dict = {}
    hour_dict = hour_dict.fromkeys(hour,0)
```

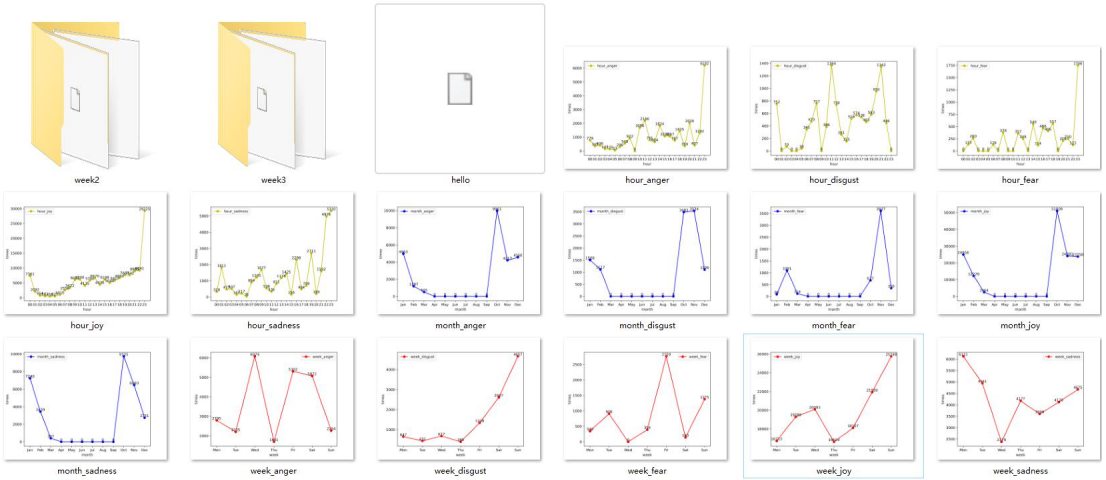
对三种不同模式进行计数（仅仅贴出 week 代码）

```python
    if time_mode == 'week':
        for tm in time_list:
            if emotion_list[time_list.index(tm)] == emotion:
                week_dict[tm[0]] += 1
        week_value = []
        for value in week_dict.values():
            week_value.append(value)
        plt.plot(week,week_value,'o-',color='r',label='week_{}'.format(emotion))
        plt.xlabel("week")#横坐标名字
        plt.ylabel("times")#纵坐标名字
        plt.legend(loc = "best")#图例
        for a,b in zip(week,week_value):
            plt.text(a,b+1,b,ha = 'center',va = 'bottom',fontsize=10)
        #print(week_dict)
```

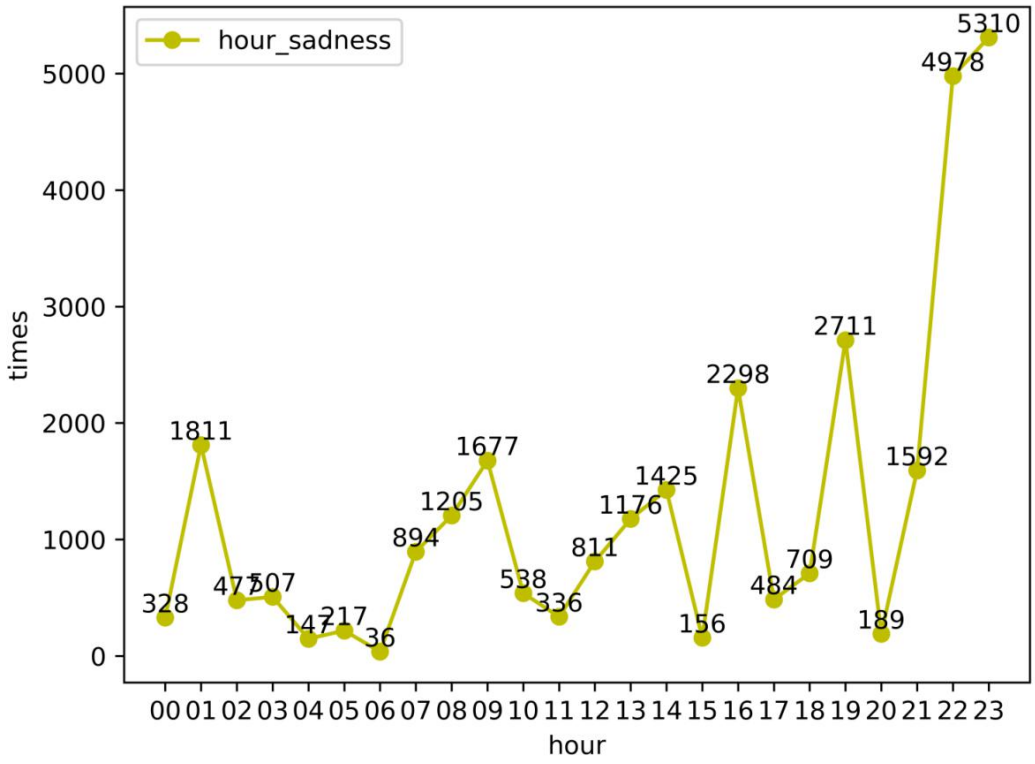
输出并保存图像

```python
        plt.savefig('{}_{}.png'.format(time_mode,emotion),dpi=800)
        #plt.show()
```

一共输出 15 张图

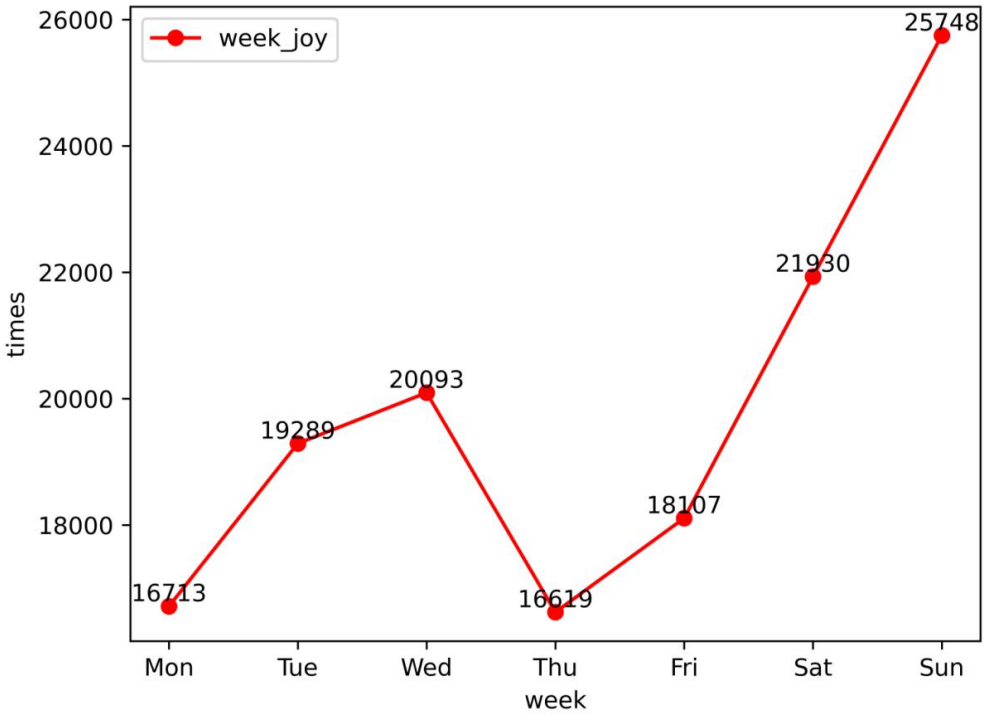


举例分析 1：



观察 24 小时内的悲伤情绪图，发现 22.23 点的悲伤微博数明显多于其他时间的微博数，考虑到经常有人在晚上“深夜 emo”，该图像很好地证明了这一点。凌晨由于发微博人数问题导致数据较低也属于正常现象。

举例分析 2：



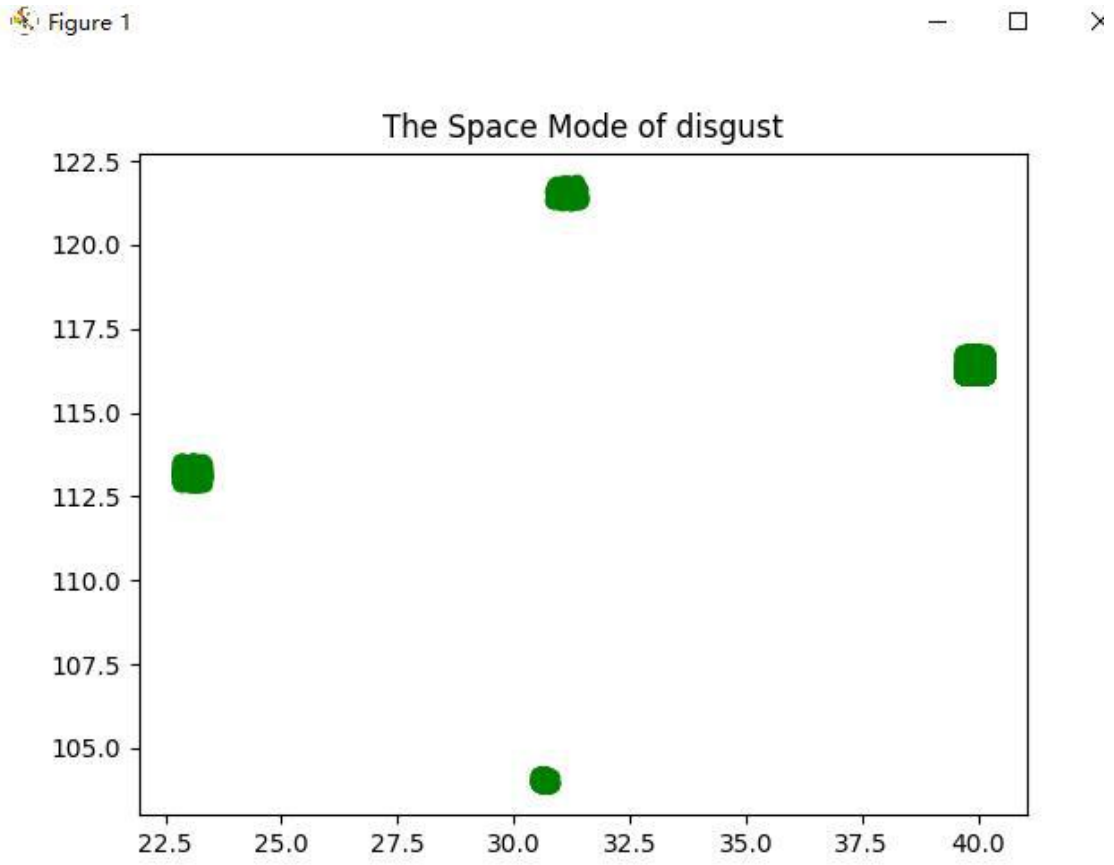
观察周一到周五的心情，周六和周日明显高于工作日，而周一是工作日的第一天，相应的数据也近似最低，

任务 6.

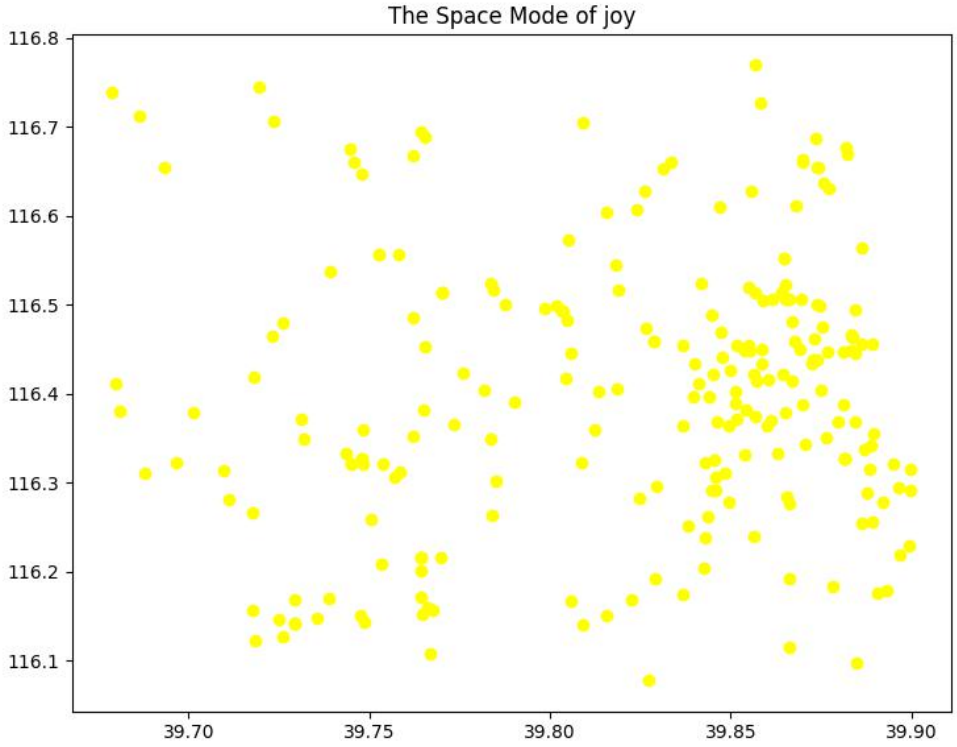
提取每条微博的地点坐标，并画出散点图

```
221 def emotion_admode(address_list,emotion_list,n):
222     """
223     画出位置分布图
224     :address_list:微博地址列表
225     :emotion_list:微博情绪列表
226     :n:列表长度
227     """
228     x_list=[]
229     y_list=[]
230     add_emotion = []
231     color_list = ['red','green','purple','yellow','blue']
232     for i in range(n):
233         zuobiao = address_list[i].split(',')
234         x_list.append(float(zuobiao[0][1:]))
235         y_list.append(float(zuobiao[1][:-1]))
236         add_emotion.append(emotion_list[i])
237     for i in range(5):
238         x=[]
239         y=[]
240         for j in range(n):
241             if EMO[i] == add_emotion[j]:
242                 x.append(x_list[j])
243                 y.append(y_list[j])
244         plt.scatter(x,y, c = color_list[i])
245         plt.title('The Space Mode of ' + EMO[i])
246         plt.show()
```

读入全部的数据



发现明显分成了四类，可以用肉眼聚类分析得知分成四个部分，需要将不同地区的数据分别分析才能得到有价值的信息。
通过筛选特定部分的微博获得同类的数据，再次进行可视化分析：



观察发现情绪为 joy 的微博有一定的聚集效应，如图中的右侧明显更为集中，这一现象可能与情绪的传染性有关，值得进一步研究。

Main 函数如下：

```
248  def main():
249      """
250      main函数
251      """
252      txt=fread_document(filename_0_test) #读取txt文件
253      m=len(txt)
254      txt=fdelete_repetition_txt(txt) #删除重复项
255      n=len(txt) #记录项数
256      print("处理前数据有%d项.\n处理重复数据后有%d项." %(m,n))
257      data=fcut_txt(txt,n) #简单切割内容、地址、时间
258      n=len(data)#更新项数
259      print(n)
260      sentence = fdelete_url_data(data,n) #简单分词处理
261      sentence = fclean_sentence(sentence,n) #正则表达式降噪
262      #faddword(filename_emotions) #添加情绪词
263      f=fdic_emotion(filename_emotions,data,sentence,n)
264      emotion_list,time_list,address_list = f()
265      time_list_new=ftime_list(time_list,n)
266      '''
267      for emotion in EMO:
268          for timemode in TIMEMODE:
269              fplotime(emotion,timemode,emotion_list,time_list_new)
270      '''
271      emotion_admode(address_list,emotion_list,n)
272
273  if __name__ == '__main__':
274      main()
```

Ref：

逐行读取 txt 文件

https://cloud.tencent.com/developer/article/2080154

学习正则表达式

https://blog.csdn.net/weixin_42550871/article/details/108943529

https://blog.csdn.net/bang152101/article/details/89284249

使用正则表达式切除 url 和域名

https://www.lmlphp.com/user/151450/article/item/8191109/