1. **问题描述**

找一个有全球新冠病毒数据的网站，爬取其中的数据（禁止使用数据接口直接获取数据）。要求爬取从 2020 年 12 月 1 日开始的连续 15 天的数据，国家数不少于 100 个。

1. **实验环境**

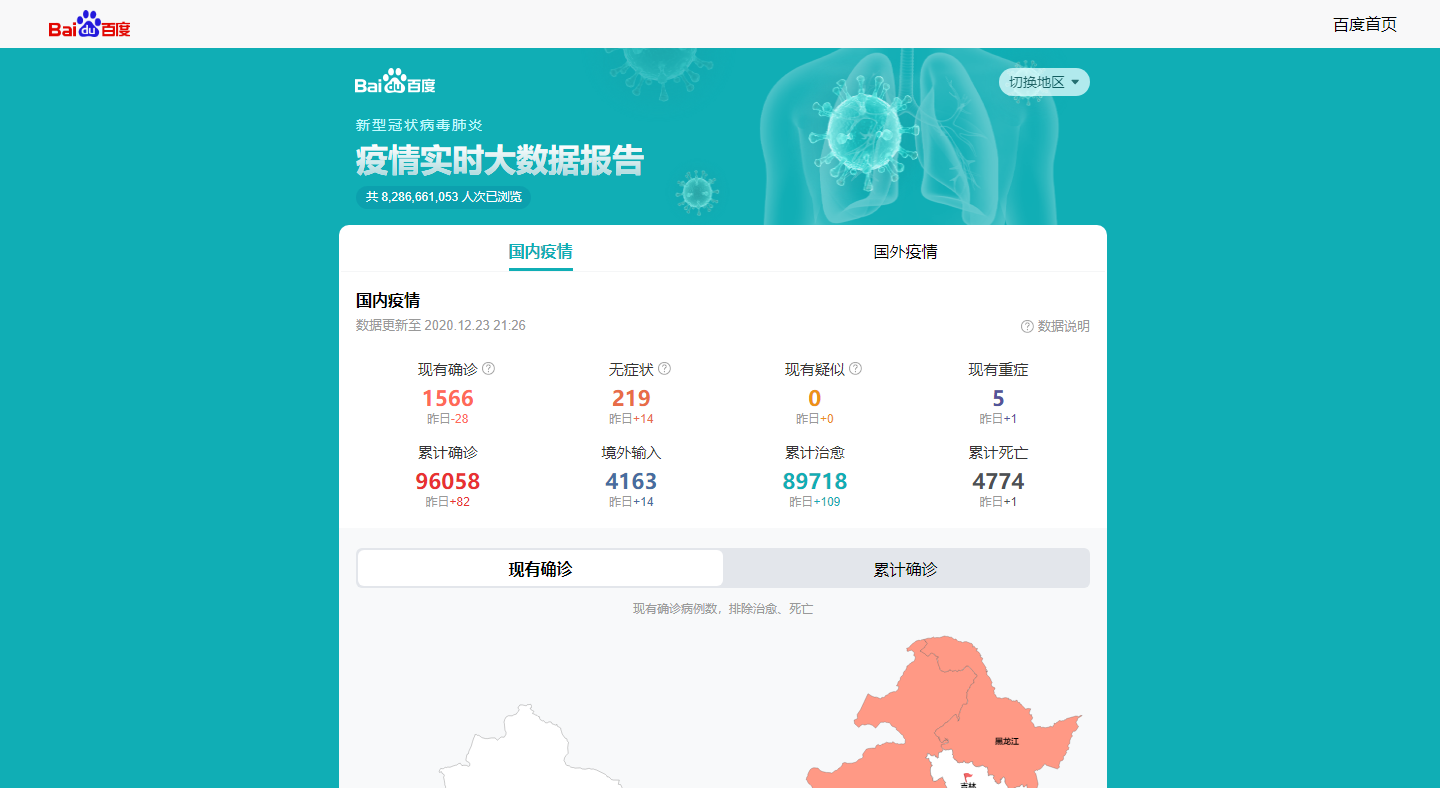
Microsoft Windows 10 版本18363

PyCharm 2020.2.1 (Community Edition)

Python 3.8(Scrapy 2.4.0 + numpy 1.19.4 + pandas 1.1.4 + matplotlib 3.3.3)

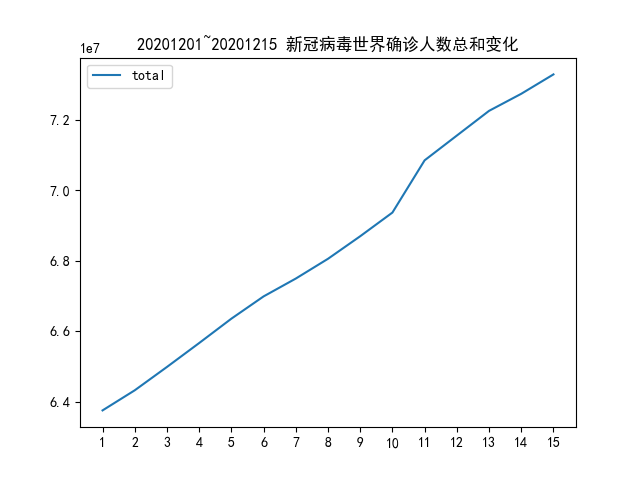
1. **实验步骤及结果**
2. 标明你的数据来源：包括网址和首页截图

https://voice.baidu.com/act/newpneumonia/newpneumonia/



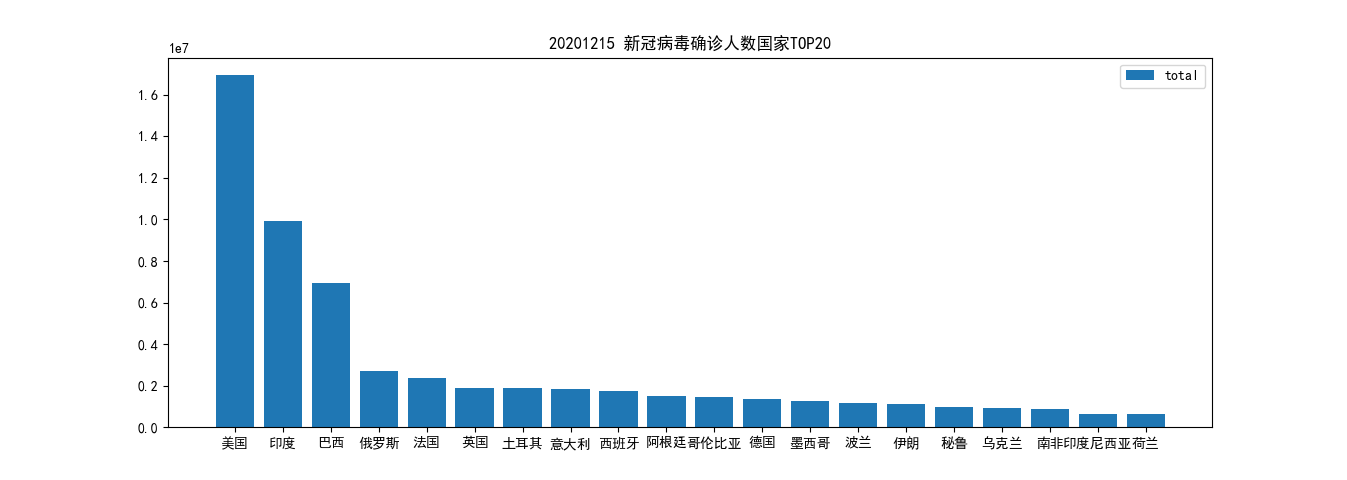


1. 数据分析和展示应包括：
   1. 15 天中，全球新冠疫情的总体变化趋势；



import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
total = [] # 记录确诊变化总数  
date = [i for i in range(1, 16)] # 记录日期序号  
  
for i in range(1, 16):  
 fileNameStr = './202012' + str(i).zfill(2) + '.csv' # 产生文件名进行读取  
 df = pd.read\_csv(fileNameStr, encoding='utf-8')  
  
 df['total'] = df['total'].astype(np.int)  
 total.append(df['total'].sum()) # 计算所有国家的确诊总数  
  
plt.title("20201201~20201215 新冠病毒世界确诊人数总和变化")  
plt.xticks(date)  
plt.plot(date, total, linestyle='-', label='total')  
plt.legend()  
plt.show()

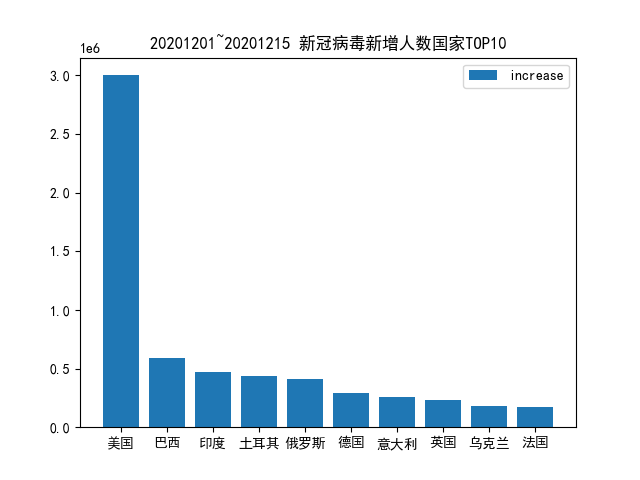
* 1. 累计确诊数排名前 20 的国家名称及其数量；



import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
fileNameStr = './20201215.csv' # 读取最后一天的疫情数据  
df = pd.read\_csv(fileNameStr, encoding='utf-8')  
  
df['total'] = df['total'].astype(np.int)  
df.sort\_values('total', inplace=True, ascending=False) # 按照确诊总数降序排序  
df = df.reset\_index(drop=True) # 重置序号  
  
country = []  
total = []  
  
for i in range(20): # 记录前20的国家名称和确诊总数  
 country.append(df['country'][i])  
 total.append(df['total'][i])  
  
plt.title("20201215 新冠病毒确诊人数国家TOP20")  
plt.bar(country, total, label='total')  
plt.legend()  
plt.show()

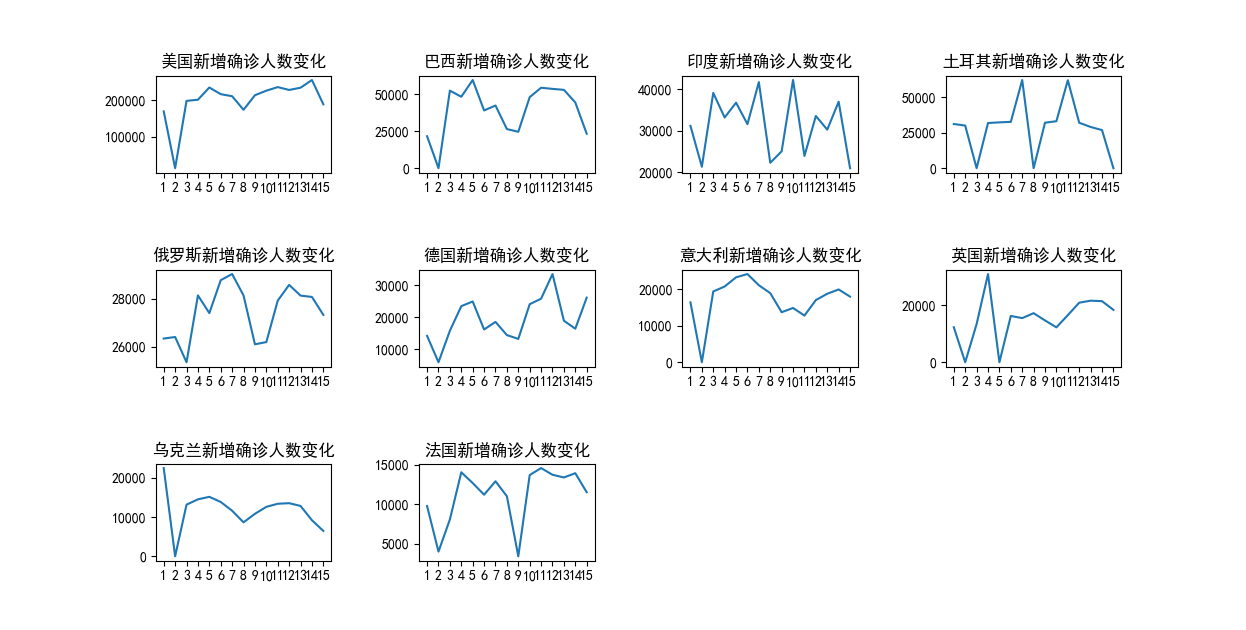
* 1. 15 天中，每日新增确诊数累计排名前 10 个国家的每日新增确诊数据的曲线图；

首先获取新增确诊数前10的国家，结果如下：



import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
res = {}  
  
for i in range(1, 16): # 统计15天的新增人数  
 fileNameStr = './202012' + str(i).zfill(2) + '.csv' # 产生文件名进行读取  
 df = pd.read\_csv(fileNameStr, encoding='utf-8')  
  
 df['increase'] = df['increase'].astype(np.int)  
  
 for idx in range(len(df)): # 遍历所有国家  
 if df['country'][idx] in res.keys():  
 res[df['country'][idx]] = res[df['country'][idx]] + df['increase'][idx]  
 else:  
 res[df['country'][idx]] = df['increase'][idx]  
  
lst = sorted(res.items(), key=lambda x:x[1], reverse=True) # 按新增人数进行排序  
country = []  
increase = []  
for i in range(10): # 取出前10的国家  
 country.append(lst[i][0])  
 increase.append(lst[i][1])  
  
plt.title("20201201~20201215 新冠病毒新增人数国家TOP10")  
plt.bar(country, increase, label='increase')  
plt.legend()  
plt.show()

之后再对这十个国家绘制新增人数变化图，结果如下：



import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
country = ['美国', '巴西', '印度', '土耳其', '俄罗斯', '德国', '意大利', '英国', '乌克兰', '法国']  
date = [i for i in range(1, 16)] # 记录日期序号  
  
fig, ax = plt.subplots()  
sub = 0  
  
for name in country:  
 sub = sub + 1  
 increase = []  
 for i in range(1, 16):  
 fileNameStr = './202012' + str(i).zfill(2) + '.csv' # 产生文件名进行读取  
 df = pd.read\_csv(fileNameStr, encoding='utf-8')  
 df['increase'] = df['increase'].astype(np.int)  
  
 val = df[df['country'].isin([name])]['increase']  
 val = val.reset\_index(drop=True)  
 increase.append(val[0])  
  
 plt.subplot(3, 4, sub)  
 plt.xticks(date)  
 plt.title(name + '新增确诊人数变化')  
 plt.plot(date, increase, linestyle='-')  
  
plt.subplots\_adjust(left=None, bottom=None, right=None, top=None, wspace=0.5, hspace=1)  
plt.show()

* 1. 累计确诊人数占国家总人口比例最高的 10 个国家；

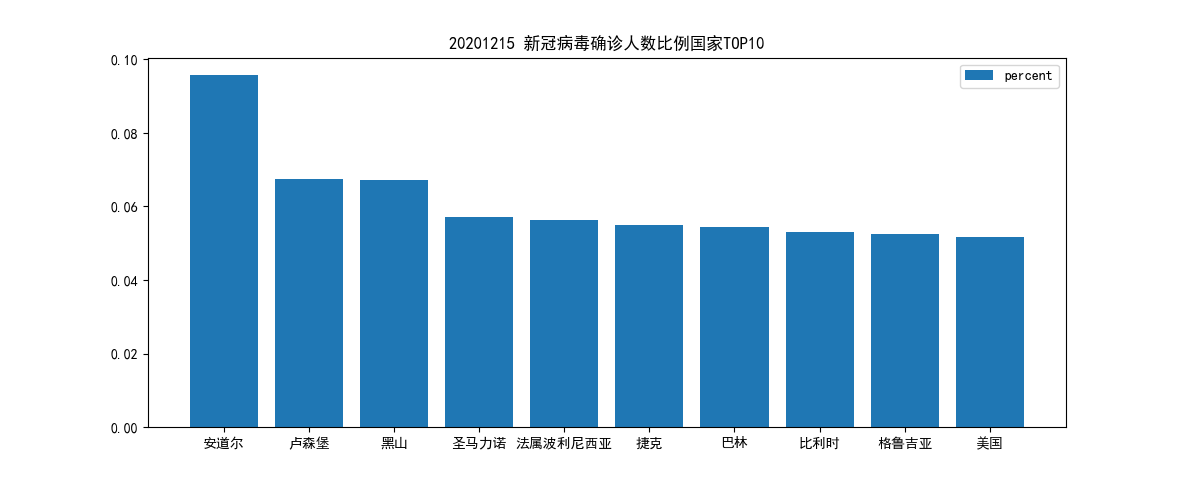
首先需要爬取全世界各国家的人口数据，使用的网站如下：

<https://www.kylc.com/stats/global/yearly_overview/g_population_total.html>

爬虫的核心代码如下：

class PopSpider(scrapy.Spider):  
 name = 'pop'  
 allowed\_domains = ['kylc.com']  
 start\_urls = ['https://www.kylc.com/stats/global/yearly\_overview/g\_population\_total.html']  
  
 def parse(self, response):  
 item = PopulationItem()  
 for each in response.xpath("//table[@class='table']/tbody/tr"):  
 item['country'] = each.xpath("td[2]/text()").extract()  
 if item['country']:  
 num = each.xpath("td[5]/text()").extract()[0]  
 item['population'] = num[num.find('(') + 1:num.find(')')].replace(',', '')  
 yield item  
 pass

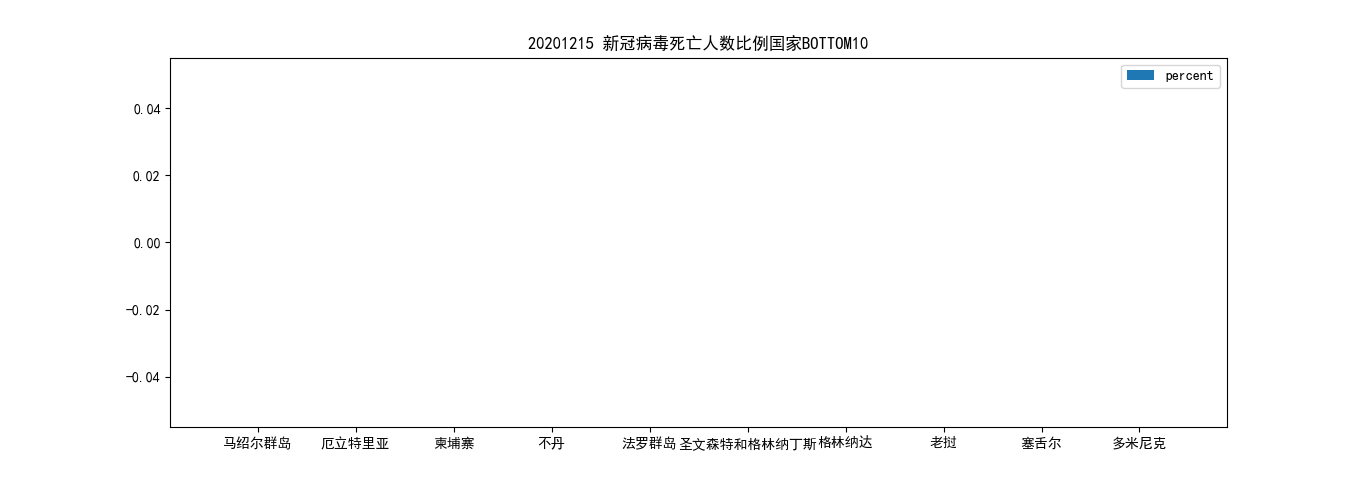
计算比例并绘制直方图如下：



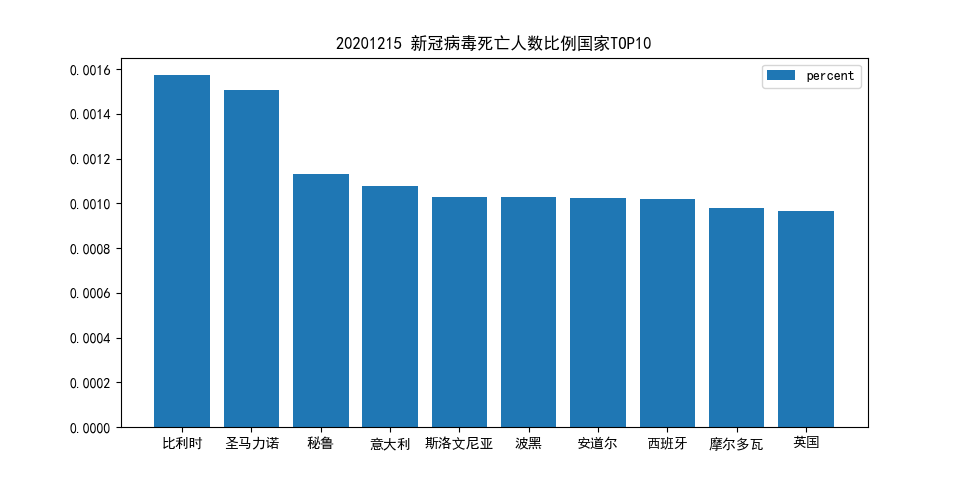
import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
fileNameStr = './20201215.csv' # 读取最后一天的疫情数据  
df = pd.read\_csv(fileNameStr, encoding='utf-8')  
population = pd.read\_csv('./population.csv', encoding='utf-8') # 读取世界人口数据  
  
df['total'] = df['total'].astype(np.int)  
df['percent'] = None # 新增一列用于存储确诊比例  
  
for i in range(len(df)): # 从人口数据中若找到该国家则计算比例  
 if not population[population['country'].isin([df['country'][i]])]['population'].empty:  
 df['percent'][i] = int(df['total'][i]) / int(population[population['country'].isin([df['country'][i]])]['population'])  
  
df.sort\_values('percent', inplace=True, ascending=False) # 按照确诊比例降序排序  
df = df.reset\_index(drop=True) # 重置序号  
  
country = []  
percent = []  
  
plt.title("20201215 新冠病毒确诊人数比例国家TOP10")  
for i in range(10): # 记录前10的国家名称和确诊比例  
 country.append(df['country'][i])  
 percent.append(df['percent'][i])  
plt.bar(country, percent, label='percent')  
plt.legend()  
plt.show()

* 1. 死亡率（累计死亡人数/累计确诊人数）最低的 10 个国家；

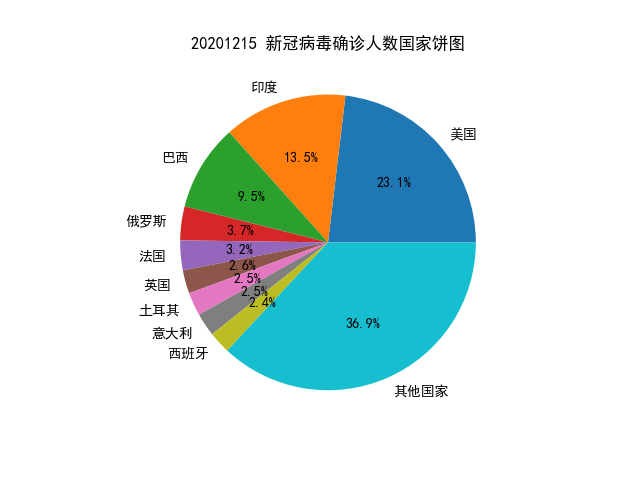
事实上从获取到的数据来看，有大量的国家都没有死亡病例数据（全部统计在治愈/新增中），因而直接展示的话结果如下：



因而此处将此题改为最高的10个国家较为合适，结果如下：

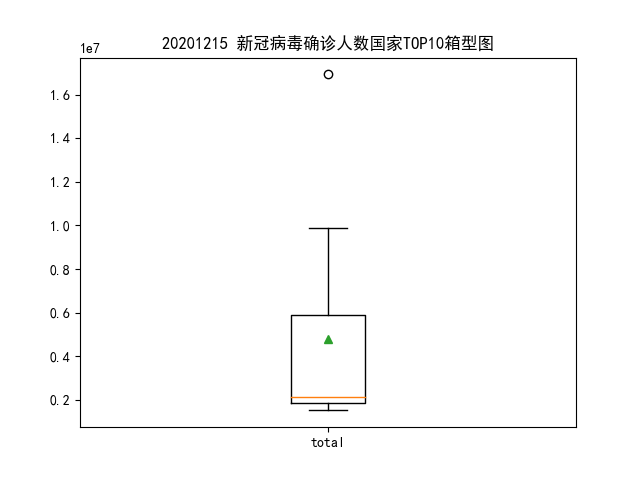


* 1. 用饼图展示各个国家的累计确诊人数的比例（你爬取的所有国家，数据较小的国家可以合并处理）；

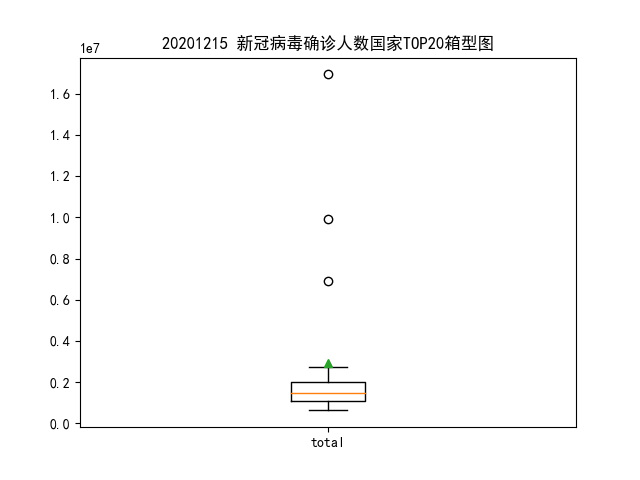


import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
fileNameStr = './20201215.csv' # 读取最后一天的疫情数据  
df = pd.read\_csv(fileNameStr, encoding='utf-8')  
  
df['total'] = df['total'].astype(np.int)  
df.sort\_values('total', inplace=True, ascending=False) # 按照确诊总数降序排序  
df = df.reset\_index(drop=True) # 重置序号  
  
country = []  
total = []  
remain = 73294020 # 由题a获取的15日的世界确诊总数  
  
for i in range(9): # 记录前10的国家名称和确诊总数  
 country.append(df['country'][i])  
 total.append(df['total'][i])  
 remain = remain - int(df['total'][i])  
  
country.append('其他国家') # 加入其他国家数据  
total.append(remain)  
  
plt.title("20201215 新冠病毒确诊人数国家饼图")  
plt.pie(total, labels=country, autopct='%1.1f%%')  
plt.show()

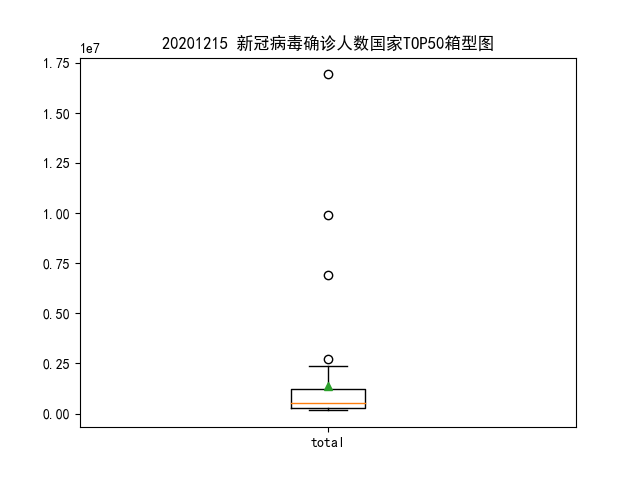
* 1. 展示全球各个国家累计确诊人数的箱型图，要有平均值；



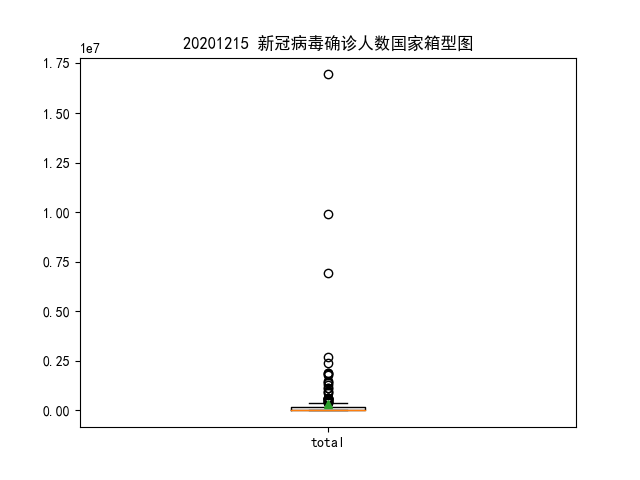
前10国家中美国被标记为异常值，均值在5000000左右。



前20国家中美国、印度、巴西被标记为异常值，均值在3000000左右。



前50国家中美国、印度、巴西、俄罗斯被标记为异常值，均值在1500000左右。

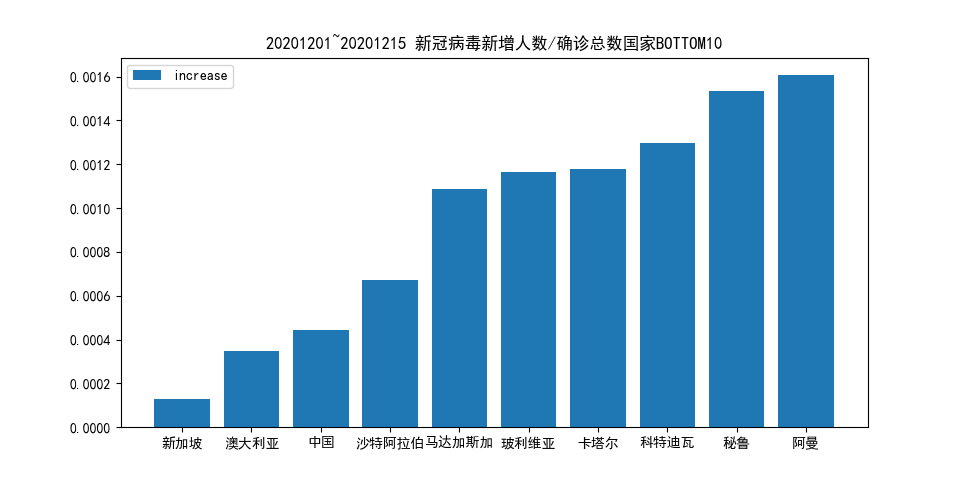


全部国家的箱型图。

import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
fileNameStr = './20201215.csv' # 读取最后一天的疫情数据  
df = pd.read\_csv(fileNameStr, encoding='utf-8')  
  
df['total'] = df['total'].astype(np.int)  
df.sort\_values('total', inplace=True, ascending=False) # 按照确诊总数降序排序  
df = df.reset\_index(drop=True) # 重置序号  
  
country = []  
total = []  
  
for i in range(10): # 记录前10的国家名称和确诊总数  
 country.append(df['country'][i])  
 total.append(df['total'][i])  
  
plt.title("20201215 新冠病毒确诊人数国家TOP10箱型图")  
plt.boxplot(total, labels=['total'], showmeans=True)  
plt.show()

* 1. 其它你希望分析和展示的数据。

展示的数据为1~15日新增确诊数/总确诊数的均值的直方图，即每日新增人数在已有总数中的比例，由于较低的确诊量难以反映出真实的应对水平，故另加一条件确诊总数需大于10000。



import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
res = {}  
  
for i in range(1, 16): # 统计15天的新增人数  
 fileNameStr = './202012' + str(i).zfill(2) + '.csv' # 产生文件名进行读取  
 df = pd.read\_csv(fileNameStr, encoding='utf-8')  
  
 df['increase'] = df['increase'].astype(np.int)  
 df['total'] = df['total'].astype(np.int)  
  
 for idx in range(len(df)): # 遍历所有国家  
 if df['total'][idx] < 10000:  
 continue  
 if df['country'][idx] in res.keys():  
 res[df['country'][idx]] = res[df['country'][idx]] + (df['increase'][idx] / df['total'][idx])  
 else:  
 res[df['country'][idx]] = df['increase'][idx] / df['total'][idx]  
  
for i in res:  
 res[i] = res[i] / 15  
  
lst = sorted(res.items(), key=lambda x:x[1], reverse=False) # 按比例进行排序  
country = []  
percent = []  
for i in range(10): # 取出前10的国家  
 country.append(lst[i][0])  
 percent.append(lst[i][1])  
  
plt.title("20201201~20201215 新冠病毒新增人数/总人数国家BOTTOM10")  
plt.bar(country, percent, label='increase')  
plt.legend()  
plt.show()

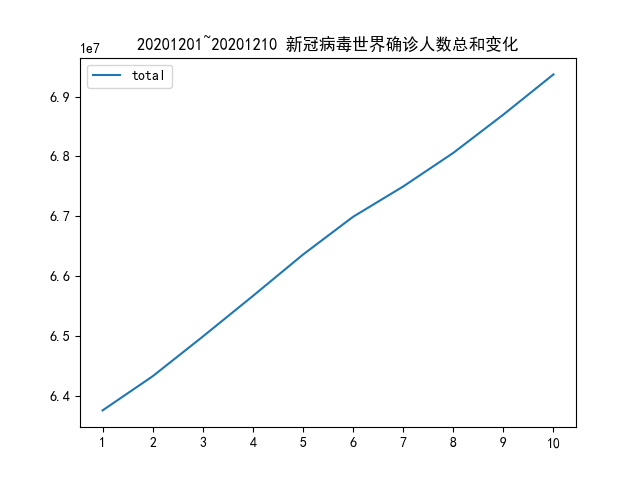
1. 根据以上数据，列出全世界应对新冠疫情最好的 10 个国家，并说明你的理由。

由h题得，应对水平应该反映在已有病例的基础上每日新增的比例小，即相比于已有的病例而言，新增量占比更小，这一般是由于严格的出入境管制、对疫情反弹的快速应对、以及较高的医疗水平，因而使用新增确诊数/总确诊数对应对水平进行评估。

得出这十个国家为新加坡、澳大利亚、中国、沙特阿拉伯、马达加斯加、玻利维亚、卡塔尔、科特迪瓦、秘鲁、阿曼。

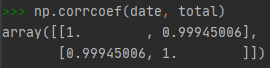
1. 针对全球累计确诊数，利用前 10 天采集到的数据做后 5 天的预测，并与实际数据进行对比。说明你预测的方法，并分析与实际数据的差距和原因。

首先代码同a题中一致，观察前十日的变化：

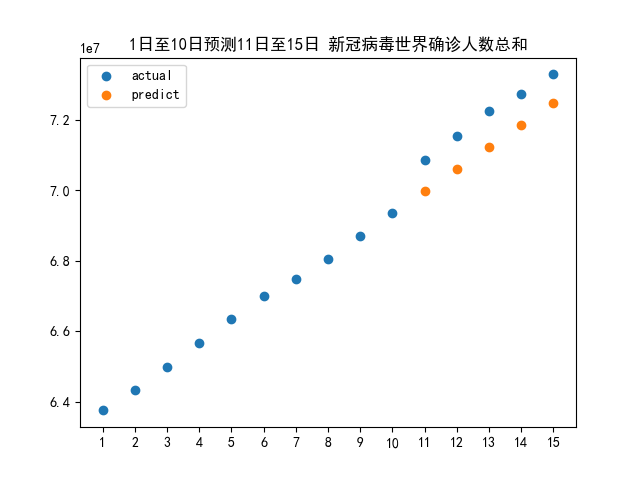


观察该变化趋势的序列数据，可以发现有明显的趋势，不存在季节性，可以从线性趋势预测着手。

其线性相关系数接近1，有着极强的线性相关性。



可以确定使用sklearn中的linear\_model进行预测。



import pandas as pd  
import numpy as np  
from matplotlib import pyplot as plt  
from sklearn import datasets, linear\_model  
from pylab import \*  
mpl.rcParams['font.sans-serif'] = ['SimHei']  
  
total = [] # 记录确诊变化总数  
date = [i for i in range(1, 16)] # 记录日期序号  
  
for i in range(1, 16):  
 fileNameStr = './202012' + str(i).zfill(2) + '.csv' # 产生文件名进行读取  
 df = pd.read\_csv(fileNameStr, encoding='utf-8')  
  
 df['total'] = df['total'].astype(np.int)  
 total.append(df['total'].sum()) # 计算所有国家的确诊总数  
  
total = np.array(total)  
date = np.array(date)  
ptotal = total[:10].reshape((10, 1)) # 截取前10个确诊人数进行预测  
pdate = date[:10].reshape((10, 1)) # 截取前10个日期进行预测  
  
model = linear\_model.LinearRegression()  
model.fit(pdate, ptotal) # 使用线性模型进行拟合  
  
predict = model.predict(date[10:].reshape(5, 1)) # 预测后5个日期的确诊人数  
predict = predict.flatten().tolist()  
  
plt.title("1日至10日预测11日至15日 新冠病毒世界确诊人数总和")  
plt.xticks(date)  
plt.scatter(date, total, label='actual') # 绘制15天的实际确诊人数  
plt.scatter(date[10:], predict, label='predict') # 绘制后5天的预测确诊人数  
plt.legend()  
plt.show()

结合真实数据，确实在11日出现了一个小幅的骤升，导致线性模型得出的预测值偏低，这主要是由于仅仅10日的数据仍然难以预测总体的变化趋势，对于像这种突发性的变化无从预测，也是在情理之中的。

1. 请贴上爬虫程序的核心代码、数据处理及数据展示的核心代码（应有足够的注释）；

如上过程中有附。

1. **心得与体会**

通过本次对新冠疫情的作业探究，在作业中的几道题目中，运用到了本学期课程中python基本语法、爬虫、数据预处理、数据可视化、数据分析等一系列知识。

这一系列的操作充分展现出python通过包的扩展，对于数据爬取、分析处理等常用功能有着极强的可实践性，充分展现出作为一门当今热门的脚本语言是何以能够如此迅速发展的原因，让我在日常中使用python扩展出了许多思路。

非常感谢老师在本学期的精心备课与教导，以及本次充实的作业设计！