



江西财经大学  
JIANGXI UNIVERSITY OF FINANCE AND ECONOMICS

课程名称: Python语言与数据分析

## 课 程 报 告

项目名称 欧洲新冠疫情数据分析

班 级 金融203班

学 号 0204871

姓 名 王惠莹

任课教师 肖 泉

开课学期: 2020 至 2021 学年 第 二 学期

完成时间: 2021 年 7 月 3 日

# 《欧洲新冠疫情数据分析》数据分析报告

## 目 录

### 目录

1 概述.....	1
1.1 研究背景.....	1
1.2 研究意义.....	1
2 数据描述.....	1
2.1 数据来源.....	1
2.2 数据导入.....	1
2.3 数据包含字段.....	2
2.4 数据清理.....	2
2.4.1 数据清理的描述 .....	2
2.4.2 查看是否含有缺失值 .....	2
2.4.3 查看是否含有异常值 .....	3
2.4.3 数据整理.....	3
3 数据分析内容.....	3
3.1 欧洲不同国家累计确诊人数、累计死亡人数、死亡率分析.....	3
3.1.1 该项分析的介绍 .....	3
3.1.2 死亡率公式 .....	3
3.2 法国、意大利、波兰、德国、西班牙确诊人数月增长率对比.....	4
3.2.1 该项分析介绍 .....	4
3.2.2 确诊人数月增长率的计算公式 .....	4
3.3 法国、意大利、波兰、德国、西班牙死亡人数月增长率对比.....	4
3.3.1 该项分析介绍 .....	4
3.3.2 死亡人数月增长率的计算公式 .....	4
4 数据分析图表.....	5
4.1 欧洲不同国家确诊人数以及死亡人数图表 .....	5
4.2 法国、意大利、波兰、德国、西班牙确诊人数月增长率图表.....	7
4.3 法国、意大利、波兰、德国、西班牙确诊人数月增长率图表.....	8
5 数据分析结果.....	8
5.1 对欧洲不同国家确诊人数以及死亡人数分析结果.....	8
5.2 对法国、意大利、波兰、德国、西班牙确诊人数月增长率分析结果.....	8
5.3 对法国、意大利、波兰、德国、西班牙死亡人数月增长率分析结果.....	9
6 总结.....	9
附录-数据分析代码 .....	9

## 1 概述

### 1.1 研究背景

由于 2019 年爆发的新冠肺炎疫情事件,导致欧洲各国的经济下滑,甚至出现了衰退的现象,各国纷纷采取了一些措施。对于欧洲各国而言,防控好疫情就是当前最重要的任务,然而各国颁布的措施有些有效,有些效果很微小甚至没有效果。在这样的背景下,就需要我们运用已有的数据对新冠疫情做出分析,了解各国的哪些措施是有效果的,哪些是没有效果的,对进一步的防控疫情给出建议。

### 1.2 研究意义

本研究将对欧洲各国的确诊人数以及死亡人数、死亡率进行分析,还会对欧洲疫情较为严重的五国——法国、西班牙、德国、波兰、意大利的确诊人数月增长率,以及死亡人数月增长率做出分析,在分析的基础上对各国的疫情防控给出建议。

## 2 数据描述

### 2.1 数据来源

数据来源于 ECDC 官网

网址为 <https://www.ecdc.europa.eu/en/publications-data>

总共 3720 条数据, 11 个字段。

### 2.2 数据导入

下载好数据后使用 Pandas 导入数据, 查看数据集的信息, 快速理解数据。

使用 `read_csv()` 读取数据的时候, 需要根据 CSV 文件存储时的编码格式, 进行读取。

## 2.3 数据包含字段

```
In [33]: df.info()
...:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3720 entries, 0 to 3719
Data columns (total 11 columns):
dateRep          3720 non-null object
day              3720 non-null int64
month            3720 non-null int64
year             3720 non-null int64
cases            3720 non-null int64
deaths           3720 non-null int64
countriesAndTerritories 3720 non-null object
geoId            3720 non-null object
countryterritoryCode 3720 non-null object
popData2020      3720 non-null int64
continentExp     3720 non-null object
dtypes: int64(6), object(5)
memory usage: 319.8+ KB
```

根据以上结果，对数据进行基本了解，11 个字段中有 6 个字段是数字类型，这 6 个字段在计算时是不需要转换类型的，其他字段的数据都是 object 类型。

在获取数据时要注意数据的类型，特别是日期字段的数据，整理数据时可以将其转换成时间格式，以方便后续的数据处理。

同时发现该数据不存在缺失值，故不用进行缺失值的处理。

## 2.4 数据清理

### 2.4.1 数据清理的描述

数据清洗是数据分析的基础，也是非常重要的一步，因为数据清洗在提高数据质量的同时也可以避免脏数据影响分析结果。

所谓数据清洗，实际上就是对缺失值、异常值的删除处理或填充处理，以及为了方便数据的获取和分析，对列名的重命名、列数据的类型转换或者是排序等操作。

但是并不是所有的数据都需要将上述的所有操作都执行一遍，具体操作的选择可根据实际的数据和需求进行选定

### 2.4.2 查看是否含有缺失值

通过 info 函数了解到在数据集中不含有缺失值。结果返回的是所有字段不为空的数据个数或者使用 isna().any() 方法，此方法会返回一个仅含 True 和 False 这两种值的 Series，这个方法主要是用来判断所有列中是否含有空值。

方法：df.info()

### 2.4.3 查看是否含有异常值

在查看数据的缺失值之后还需要检查一下数据中是否含有异常值，Pandas 的 `describe()` 可以用来统计数据集的集中趋势，分析各行列的分布情况，因此在查看异常值时会经常用到。`describe()` 函数会对数值型数据进行统计，输出结果指标包括 `count`、`mean`、`std`、`min`、`max` 及下四分位数，中位数和上四分位数。通过观察发现该数据的死亡人数中出现了一个负数，为异常值，需要对其进行处理，把死亡人数为负数的值去掉。

方法：`df.describe()`

```
df1=df.query('deaths<0')
```

```
lst=list(df1['deaths'])
```

```
df=df[df.deaths.isin(lst)]
```

```
In [36]: print(df.describe())
```

	day	month	...	deaths	popData2020
count	3720.000000	3720.000000	...	3720.000000	3.720000e+03
mean	15.531183	4.530914	...	199.021505	1.510301e+07
std	8.919291	1.153277	...	2892.827344	2.121733e+07
min	1.000000	2.000000	...	-3.000000	3.874700e+04
25%	8.000000	3.000000	...	1.000000	2.095861e+06
50%	16.000000	5.000000	...	10.000000	6.387122e+06
75%	23.000000	6.000000	...	51.000000	1.152244e+07
max	31.000000	7.000000	...	97699.000000	8.316671e+07

[8 rows x 6 columns]

### 2.4.3 数据整理

由于很多分析的维度都是建立在时间基础上的，通过数据类型的结果发现数据中的时间是字符串类型的，所以需要处理时间的类型，将其修改成 `datetime` 类型。

方法：`df['dateRep']=pd.to_datetime(df['dateRep'])`

```
print(df.dtypes)
```

## 3 数据分析内容

### 3.1 欧洲不同国家累计确诊人数、累计死亡人数、死亡率分析

#### 3.1.1 该项分析的介绍

确诊人数是诊断过并且已经确认患上了新冠肺炎的人数，死亡人数主要核算的是因为患上该新冠肺炎而逝世的人数。核算不同国家的确诊人数、死亡人数以及死亡率能够很好的衡量该国的疫情状况以及该国的疫情防控状况，也能够给该国应该做出什么样的政策指令提供一定的提示性作用。该项数据越大，则说明该国的疫情比较严重，说明该国政府必须要对此加以重视。

#### 3.1.2 死亡率公式

某国死亡率=该国死亡人数/该国确诊人数

## 3.2 法国、意大利、波兰、德国、西班牙确诊人数月增长率对比

### 3.2.1 该项分析介绍

确诊人数月增长率是本月确诊人数增长额同上月累计确诊人数之比。本月累计确诊人数增长额为本月确诊人数减去上月确诊人数的差额,它是衡量一个国家或地区疫情防控能力的重要指标,该项指标越小,表明该国或该地区疫情防控能力越好,政府越有作为。通过第一项的分析已经知道法国、意大利、波兰、德国、西班牙是欧洲疫情较为严重的国家,因此,我们挑选出了这五国对其进行分析。对这五国的月增长率进行分析可以让我们能够更为直观地看出这五国地疫情防控情况,以便给出政策性地建议。

### 3.2.2 确诊人数月增长率的计算公式

某国确诊人数月增长率= (本月确诊人数-上月确诊人数) /上月确诊人数\*100%

## 3.3 法国、意大利、波兰、德国、西班牙死亡人数月增长率对比

### 3.3.1 该项分析介绍

死亡人数月增长率是本月死亡人数增长额同上月累计死亡人数的比率。本月累计死亡人数增长额为本月死亡人数减去上月死亡人数的差额,它不仅仅是衡量一个国家或者地区疫情防控能力的重要指标,还是衡量一个国家或者地区医疗卫生水平的重要指标,也是衡量一个地区或者国家应对突发公共卫生事件时医疗应急能力的重要指标。该项指标越低,则说明该国家或者该地区医疗水平比较高,也可以说明该地区应对突发公共卫生事件时有较好的医疗应急能力,政府在医疗方面的措施也很有效,该地疫情防控取得一定成效。对死亡人数月增长率进行分析,我们可以从分析中给出相关性的政策建议,医疗方面的建议等。

### 3.3.2 死亡人数月增长率的计算公式

某国死亡人数月增长率= (本月死亡人数-上月死亡人数) /上月死亡人数\*100%

## 4 数据分析图表

### 4.1 欧洲不同国家确诊人数以及死亡人数图表

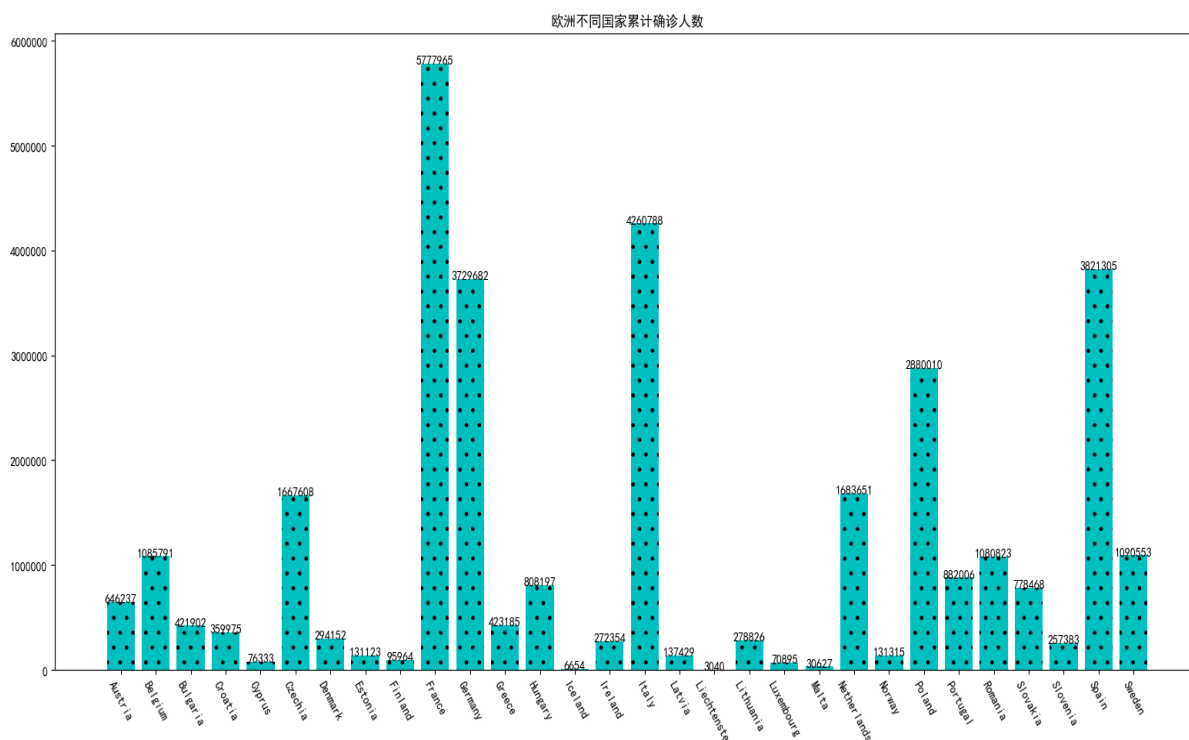


图 1-1

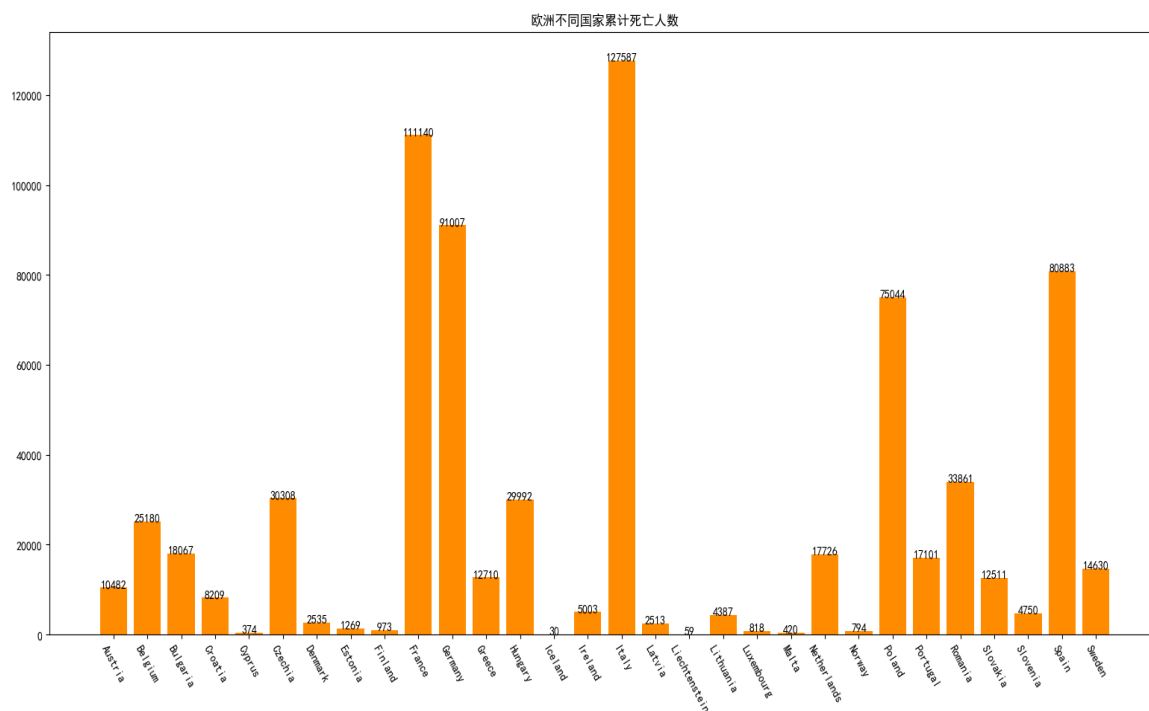


图 1-2

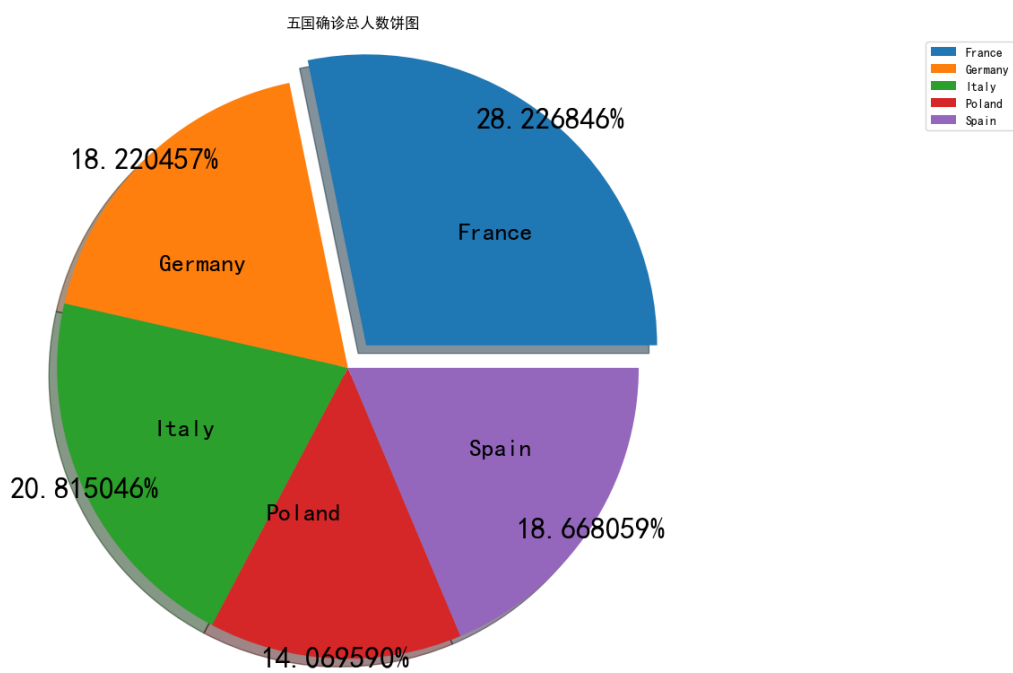


图 1-3

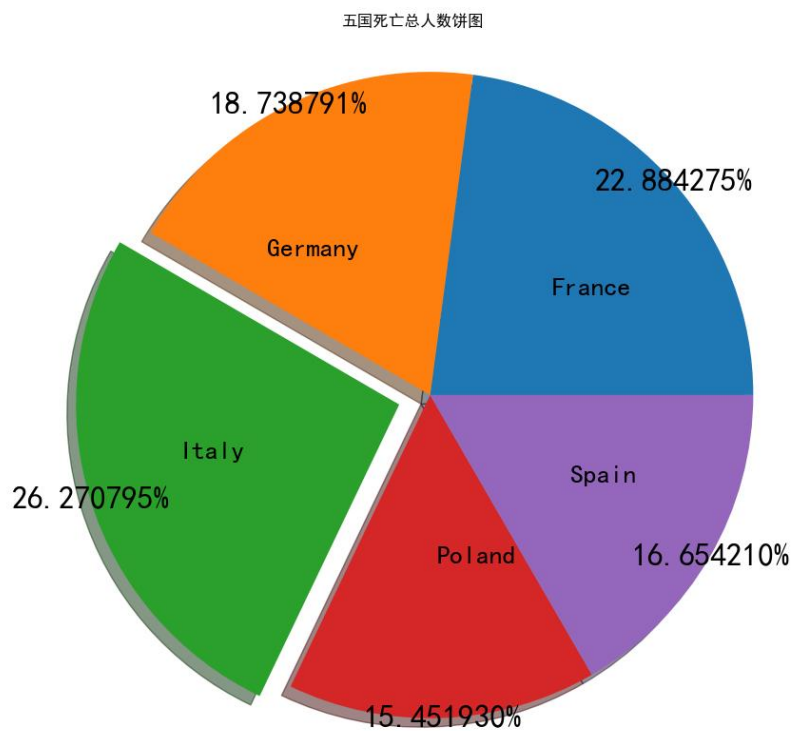


图 1-4



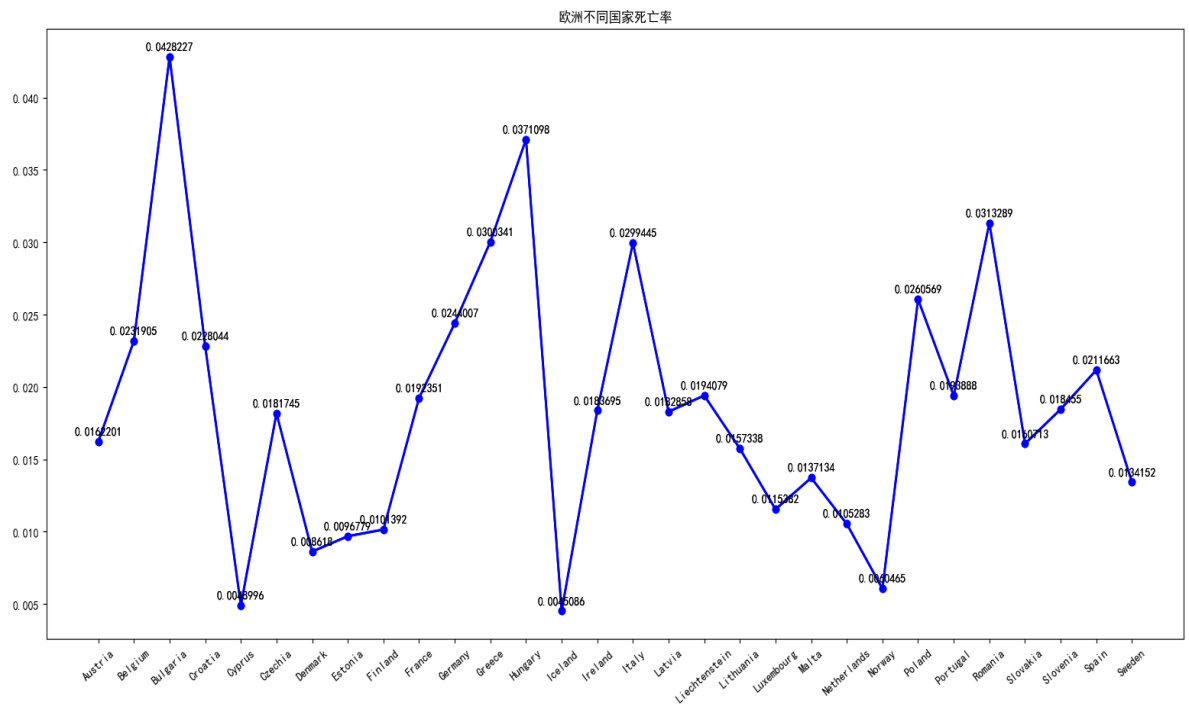


图 1-5

## 4.2 法国、意大利、波兰、德国、西班牙确诊人数月增长率图表

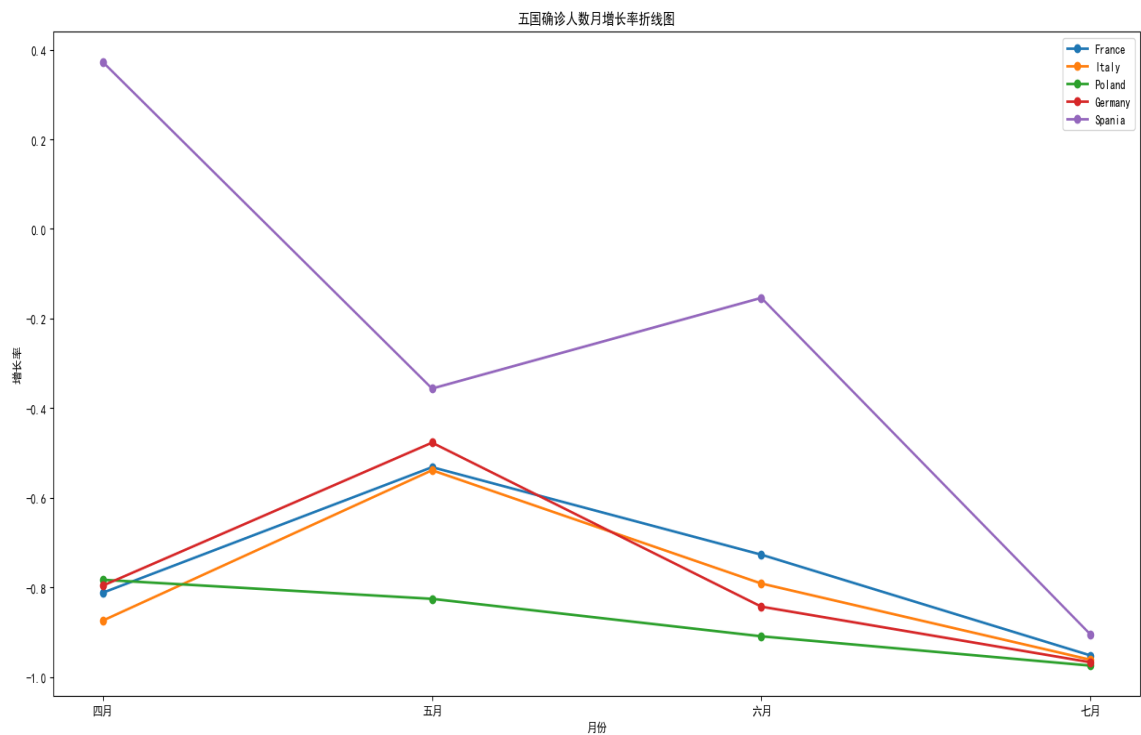


图 1-6

### 4.3 法国、意大利、波兰、德国、西班牙确诊人数月增长率图表

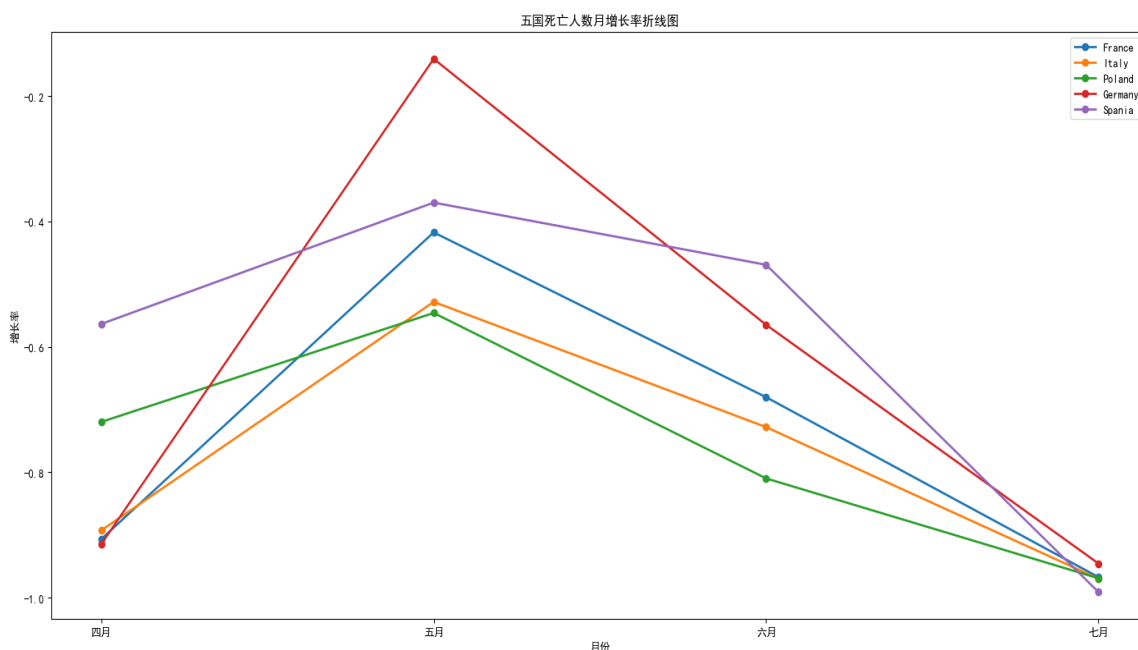


图 1-7

## 5 数据分析结果

文字阐述数据分析的结果以及启示

### 5.1 对欧洲不同国家确诊人数以及死亡人数分析结果

从图表 1-1，欧洲不同国家累计确诊人数中可以看出，法国、意大利、西班牙、德国、波兰是欧洲国家中确诊人数最多的，从图 1-2 可以看出意大利、法国、德国、西班牙、波兰也是欧洲这些国家中累计死亡人数最多的。从图 1-3 中可以看出，欧洲国家中死亡率最高的是保加利亚、匈牙利、意大利、罗马尼亚、波兰。从中可见，波兰和意大利不仅仅确诊人数多，而且死亡率也比较高，这两个国家的政府应该要高度重视起来，政府应该要采取相应的措施，以免疫情在欧洲和全世界再进一步扩散开来。同时，从图 1-4 中也可以看出，法国在这五个疫情比较严重的欧洲国家中，确认人数占比百分之二十八，意大利在这五个国家中死亡率占比达到百分之二十六，这两个国家的政府应该要高度重视起来，法国应该要采取更加有力的措施，加大对疫情的防控，呼吁全员少出门，出门应该要带口罩。意大利可以采取增加医院床位等的措施，或者呼吁市民打疫苗，降低死亡率。

### 5.2 对法国、意大利、波兰、德国、西班牙确诊人数月增长率分析结果

从图 1-4 五国增长率折线图中可以看出，法国、意大利、西班牙、波兰、德国这五个国家的确诊人数月增长率大体都在呈现降低的趋势，尤其是西班牙，降低的最快，并且西班牙在四月到五月的增长率是大幅降低的，这与政府颁布的相关性政策措施不无关系，西班牙首相佩德罗·桑切斯在三月底时曾表示，在国家紧急状态下，西班牙能最大限度地调动资源以防控

新冠肺炎疫情。他号召西班牙民众“以负责任的态度团结起来遏制病毒传播”。当前，西班牙国家和地方警察等安全力量在内政部统筹下实施系列应急管理举措。除特定情况外，全国民众不能离开住所。这些情况包括购买生活必需品、就医、上下班、照顾老年人和未成年人、去金融机构或保险机构等。西班牙全国所有教育机构停止现场教学，除供应食品等生活必需品、报纸杂志、燃料等的商店外，其他零售商店全部停止营业。可见，西班牙政府的这一系列相关性举措是非常有效的，因此，西班牙政府可以继续采取这类相关性举措。与此同时，可以注意到德国、法国、意大利这三个地区疫情在五月份的时候反弹，增长率增加。通过相关性的新闻我们可以注意到在四月份时这三个国家部分地区放松管控，允许夜店等人群聚集场所营业；部分人不遵守防疫规定；境外度假回国人数大幅增加。因此，这三个国家的政府必须要采取相应的强制性的措施，比如严格禁止人群聚集场所的营业，对境外回国的居民加强管控，做好隔离工作等措施。

### 5.3 对法国、意大利、波兰、德国、西班牙死亡人数月增长率分析结果

从图表 1-5，五国死亡人数月增长率折线图我们可以看出，从四月份到五月份，各个国家的死亡率都呈现出了上升的趋势，并都达到了各自的死亡率的最高点，尤其是德国，死亡增长率增加。但在五月之后，各国的死亡人数增长率都呈现出了下降的趋势。通过相关性新闻可知，由于各国接种疫苗的人数增加，死亡增长率开始呈现下降趋势，可见新冠疫苗对于疫情防控是十分有效果的，各国政府应该加大各自地区新冠疫苗的接种率。

## 6 总结

在整个分析过程中，综合运用到了 python 的基础知识，除此之外还有一些课外的扩展知识。在整个分析的过程中，我认识到了数据分析的重要性，它可以为人们的决策提供依据，让政府的决策更具有科学性，把国家新冠疫情的状况可视化，这对于管理一个国家、制定决策具有十分重要的意义。在数据分析的过程中，我明白了在对一个数据进行分析时，不仅仅要横向对比数据，还需要纵向对比数据，要观察数据的整体趋势，还要观察数据与数据之间的共性与差异，从而才可以得出更加准确和有效的结论。

## 附录-数据分析代码

```
import pandas as pd
import matplotlib.pyplot as plt
from decimal import Decimal

plt.rcParams['font.sans-serif']=['SimHei']
plt.rcParams['axes.unicode_minus']=False

#获取数据
df=pd.read_csv(r'd:\data.csv',encoding='cp936')

#查看是否存在缺失值
print(df.info()) #没有缺失值
```

```
#查看是否存在异常值
print(df.describe())
#对异常值进行处理
df1=df.query('deaths<0')
lst=list(df1['deaths'])
df=df[-df.deaths.isin(lst)]

#把时间数据修改成 datetime 类型

df['dateRep']=pd.to_datetime(df['dateRep'])
print(df.dtypes)

#数据分析
#欧洲不同国家的确诊人数以及死亡人数分析
g=df.groupby('countriesAndTerritories')
g1=g.sum()
lstindex=list(g1.index)
lstcases=list(g1.cases)
lstdeaths=list(g1.deaths)

#绘制累计确诊人数柱状图
plt.figure(figsize=(15,15))
plt.bar(lstindex,lstcases,color='c',hatch='.')
plt.xticks(rotation=300)
for x,y in enumerate(lstcases):
    plt.text(x,y+0.03,'{}'.format(y),ha='center',fontsize=10)

plt.title('欧洲不同国家累计确诊人数')
plt.show()

#绘制累计死亡人数柱状图
plt.figure(figsize=(15,15))
plt.bar(lstindex,lstdeaths,color='darkorange')
plt.xticks(rotation=300)
for x,y in enumerate(lstdeaths):
    plt.text(x,y+0.03,'{}'.format(y),ha='center',fontsize=10)
plt.title('欧洲不同国家累计死亡人数')
plt.show()

#死亡率图表
plt.figure(figsize=(15,15))
lst_death_rate=[]
for k,v in zip(lstdeaths,lstcases):
```

```

i=k/v
i1=Decimal(i).quantize(Decimal('0.0000000'))
i1=float(i1)
lst_death_rate.append(i1)
plt.plot(lstindex,lst_death_rate,ls='-',color='b',linewidth=2,marker='o')
plt.xticks(rotation=400)
for x,y in enumerate(lst_death_rate):
    plt.text(x,y+0.0005,'{}'.format(y),ha='center',fontsize=10)
plt.title('欧洲不同国家死亡率')
plt.show()

#绘制法国、西班牙、波兰、意大利、德国确诊总人数的饼图
g2=g1.loc[['France','Germany','Italy','Poland','Spain']]
lstcases=list(g2['cases'])
lstd=list(g2['deaths'])
explode=[0.1,0,0,0,0]
plt.figure(figsize=(15,15))
patches,l_text,p_text=plt.pie(lstcases=explode,labels=['France','Germany','Italy','Poland','Spain'],au
topct='%1f%%',pctdistance=1,shadow=True,labeldistance=0.5)
for x in l_text:
    x.set_size(20)
for x in p_text:
    x.set_size(24)
plt.axis('equal')
plt.legend()
plt.title('五国确诊总人数饼图')
plt.show()

#绘制法国、西班牙、波兰、意大利、德国死亡总人数的饼图
explode1=[0,0,0.1,0,0]
plt.figure(figsize=(15,15))
patches,l_text,p_text=plt.pie(lstd=explode1,labels=['France','Germany','Italy','Poland','Spain'],autop
ct='%1f%%',pctdistance=1,shadow=True,labeldistance=0.5)
for x in l_text:
    x.set_size(20)
for x in p_text:
    x.set_size(24)
plt.axis('equal')
plt.legend()
plt.title('五国死亡总人数饼图')
plt.show()

#法国、意大利、波兰、德国、西班牙确诊人数月增长率对比
#数据准备

```

```
dff=df.set_index('countriesAndTerritories')
dfffrance=dff.loc['France']
dffitaly=dff.loc['Italy']
dffpoland=dff.loc['Poland']
dffgermany=dff.loc['Germany']
dffspania=dff.loc['Spain']

dffrance=dfffrance.groupby('month')
dffitaly=dffitaly.groupby('month')
dffpoland=dffpoland.groupby('month')
dffgermany=dffgermany.groupby('month')
dffspania=dffspania.groupby('month')

gfrance=dffrance.sum()
gitaly=dffitaly.sum()
gpoland=dffpoland.sum()
ggermany=dffgermany.sum()
gspania=dffspania.sum()

#法国的月增长率计算
france_rate_4=gfrance.loc[4,'cases']/gfrance.loc[3,'cases']-1
france_rate_5=gfrance.loc[5,'cases']/gfrance.loc[4,'cases']-1
france_rate_6=gfrance.loc[6,'cases']/gfrance.loc[5,'cases']-1
france_rate_7=gfrance.loc[7,'cases']/gfrance.loc[6,'cases']-1
lstfrance=[france_rate_4,france_rate_5,france_rate_6,france_rate_7]

#意大利的月增长率计算
italy_rate_4=gitaly.loc[4,'cases']/gitaly.loc[3,'cases']-1
italy_rate_5=gitaly.loc[5,'cases']/gitaly.loc[4,'cases']-1
italy_rate_6=gitaly.loc[6,'cases']/gitaly.loc[5,'cases']-1
italy_rate_7=gitaly.loc[7,'cases']/gitaly.loc[6,'cases']-1
lstitaly=[italy_rate_4,italy_rate_5,italy_rate_6,italy_rate_7]

#波兰的月增长率
poland_rate_4=gpoland.loc[4,'cases']/gpoland.loc[3,'cases']-1
poland_rate_5=gpoland.loc[5,'cases']/gpoland.loc[4,'cases']-1
poland_rate_6=gpoland.loc[6,'cases']/gpoland.loc[5,'cases']-1
poland_rate_7=gpoland.loc[7,'cases']/gpoland.loc[6,'cases']-1
lstpoland=[poland_rate_4,poland_rate_5,poland_rate_6,poland_rate_7]

#德国的月增长率
germany_rate_4=ggermany.loc[4,'cases']/ggermany.loc[3,'cases']-1
germany_rate_5=ggermany.loc[5,'cases']/ggermany.loc[4,'cases']-1
germany_rate_6=ggermany.loc[6,'cases']/ggermany.loc[5,'cases']-1
```

```
germany_rate_7=ggermany.loc[7,'cases']/ggermany.loc[6,'cases']-1
lstgermany=[germany_rate_4,germany_rate_5,germany_rate_6,germany_rate_7]
```

#西班牙的月增长率

```
spania_rate_4=gspania.loc[4,'cases']/gspania.loc[3,'cases']-1
spania_rate_5=gspania.loc[5,'cases']/gspania.loc[4,'cases']-1
spania_rate_6=gspania.loc[6,'cases']/gspania.loc[5,'cases']-1
spania_rate_7=gspania.loc[7,'cases']/gspania.loc[6,'cases']-1
lstspania=[spania_rate_4,spania_rate_5,spania_rate_6,spania_rate_7]
```

```
x=['四月','五月','六月','七月']
```

```
plt.figure(figsize=(15,15))
plt.plot(x,lstfrance,label='France',linewidth=2,marker='o')
plt.plot(x,lstitaly,label='Italy',linewidth=2,marker='o')
plt.plot(x,lstpoland,label='Poland',linewidth=2,marker='o')
plt.plot(x,lstgermany,label='Germany',linewidth=2,marker='o')
plt.plot(x,lstspania,label='Spania',linewidth=2,marker='o')
plt.legend()
plt.xlabel('月份')
plt.ylabel('增长率')
plt.title('五国确诊人数月增长率折线图')
plt.show()
```

#法国、意大利、波兰、德国、西班牙死亡人数月增长率对比

#法国的死亡人数月增长率计算

```
francedeath_rate_4=gfrance.loc[4,'deaths']/gfrance.loc[3,'deaths']-1
francedeath_rate_5=gfrance.loc[5,'deaths']/gfrance.loc[4,'deaths']-1
francedeath_rate_6=gfrance.loc[6,'deaths']/gfrance.loc[5,'deaths']-1
francedeath_rate_7=gfrance.loc[7,'deaths']/gfrance.loc[6,'deaths']-1
lstfrancedeath=[francedeath_rate_4,francedeath_rate_5,francedeath_rate_6,francedeath_rate_7]
```

#意大利的死亡人数月增长率计算

```
italydeath_rate_4=gitaly.loc[4,'deaths']/gitaly.loc[3,'deaths']-1
italydeath_rate_5=gitaly.loc[5,'deaths']/gitaly.loc[4,'deaths']-1
italydeath_rate_6=gitaly.loc[6,'deaths']/gitaly.loc[5,'deaths']-1
italydeath_rate_7=gitaly.loc[7,'deaths']/gitaly.loc[6,'deaths']-1
lstitalydeath=[italydeath_rate_4,italydeath_rate_5,italydeath_rate_6,italydeath_rate_7]
```

#波兰的死亡人数月增长率

```
polanddeath_rate_4=gpoland.loc[4,'deaths']/gpoland.loc[3,'deaths']-1
polanddeath_rate_5=gpoland.loc[5,'deaths']/gpoland.loc[4,'deaths']-1
polanddeath_rate_6=gpoland.loc[6,'deaths']/gpoland.loc[5,'deaths']-1
polanddeath_rate_7=gpoland.loc[7,'deaths']/gpoland.loc[6,'deaths']-1
lstpolanddeath=[polanddeath_rate_4,polanddeath_rate_5,polanddeath_rate_6,polanddeath_rate_7]
```

#德国的月增长率

```
germanydeath_rate_4=ggermany.loc[4,'deaths']/ggermany.loc[3,'deaths']-1
germanydeath_rate_5=ggermany.loc[5,'deaths']/ggermany.loc[4,'deaths']-1
germanydeath_rate_6=ggermany.loc[6,'deaths']/ggermany.loc[5,'deaths']-1
germanydeath_rate_7=ggermany.loc[7,'deaths']/ggermany.loc[6,'deaths']-1
lstgermanydeath=[germanydeath_rate_4,germanydeath_rate_5,germanydeath_rate_6,germanydeath_rate_7]
```

#西班牙的月增长率

```
spaniadeath_rate_4=gspania.loc[4,'deaths']/gspania.loc[3,'deaths']-1
spaniadeath_rate_5=gspania.loc[5,'deaths']/gspania.loc[4,'deaths']-1
spaniadeath_rate_6=gspania.loc[6,'deaths']/gspania.loc[5,'deaths']-1
spaniadeath_rate_7=gspania.loc[7,'deaths']/gspania.loc[6,'deaths']-1
lstspaniadeath=[spaniadeath_rate_4,spaniadeath_rate_5,spaniadeath_rate_6,spaniadeath_rate_7]
```

```
plt.figure(figsize=(15,15))
plt.plot(x,lstfrancedeath,label='France',linewidth=2,marker='o')
plt.plot(x,lstitalydeath,label='Italy',linewidth=2,marker='o')
plt.plot(x,lstpolanddeath,label='Poland',linewidth=2,marker='o')
plt.plot(x,lstgermanydeath,label='Germany',linewidth=2,marker='o')
plt.plot(x,lstspaniadeath,label='Spania',linewidth=2,marker='o')
plt.legend()
plt.xlabel('月份')
plt.ylabel('增长率')
plt.title('五国死亡人数月增长率折线图')
plt.show()
```