作业1

第一题

代码

```python
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import matplotlib.font_manager as fm
import matplotlib.dates as mdates

data_history = pd.read_excel("dataset/covid19_data.xls",
sheet_name="data_history")

fm.findfont(fm.FontProperties(fname="Microsoft Yahei"))
plt.rcParams["font.sans-serif"] = ["Microsoft Yahei"]
plt.rcParams["axes.unicode_minus"] = False

data_history['date'] = pd.to_datetime(data_history['date'])

fig, ax = plt.subplots(2, 1, figsize=(12, 12))

# Line plot
ax[0].plot(data_history['date'], data_history['confirm'], label='Confirmed',
color='blue', linestyle='-')
ax[0].plot(data_history['date'], data_history['dead'], label='Deaths',
color='red', linestyle='--')
ax[0].plot(data_history['date'], data_history['heal'], label='Recovered',
color='green', linestyle='-.')

# Scatter plot
ax[1].scatter(data_history['date'], data_history['confirm'], label='Confirmed',
color='blue')
ax[1].scatter(data_history['date'], data_history['dead'], label='Deaths',
color='red')
ax[1].scatter(data_history['date'], data_history['heal'], label='Recovered',
color='green')

ax[0].set_title('COVID-19 Data Over Time (Line Plot)', fontsize=14)
ax[1].set_title('COVID-19 Data Over Time (Scatter Plot)', fontsize=14)
for a in ax:
    a.set_xlabel('Date', fontsize=12)
    a.set_ylabel('Number of Cases', fontsize=12)
    a.legend()
    a.xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m-%d'))
    a.xaxis.set_major_locator(mdates.MonthLocator())
    a.tick_params(axis='x', rotation=45)
    a.set_xticks(data_history['date'])


plt.tight_layout()
plt.savefig("res/1-1.png")
```
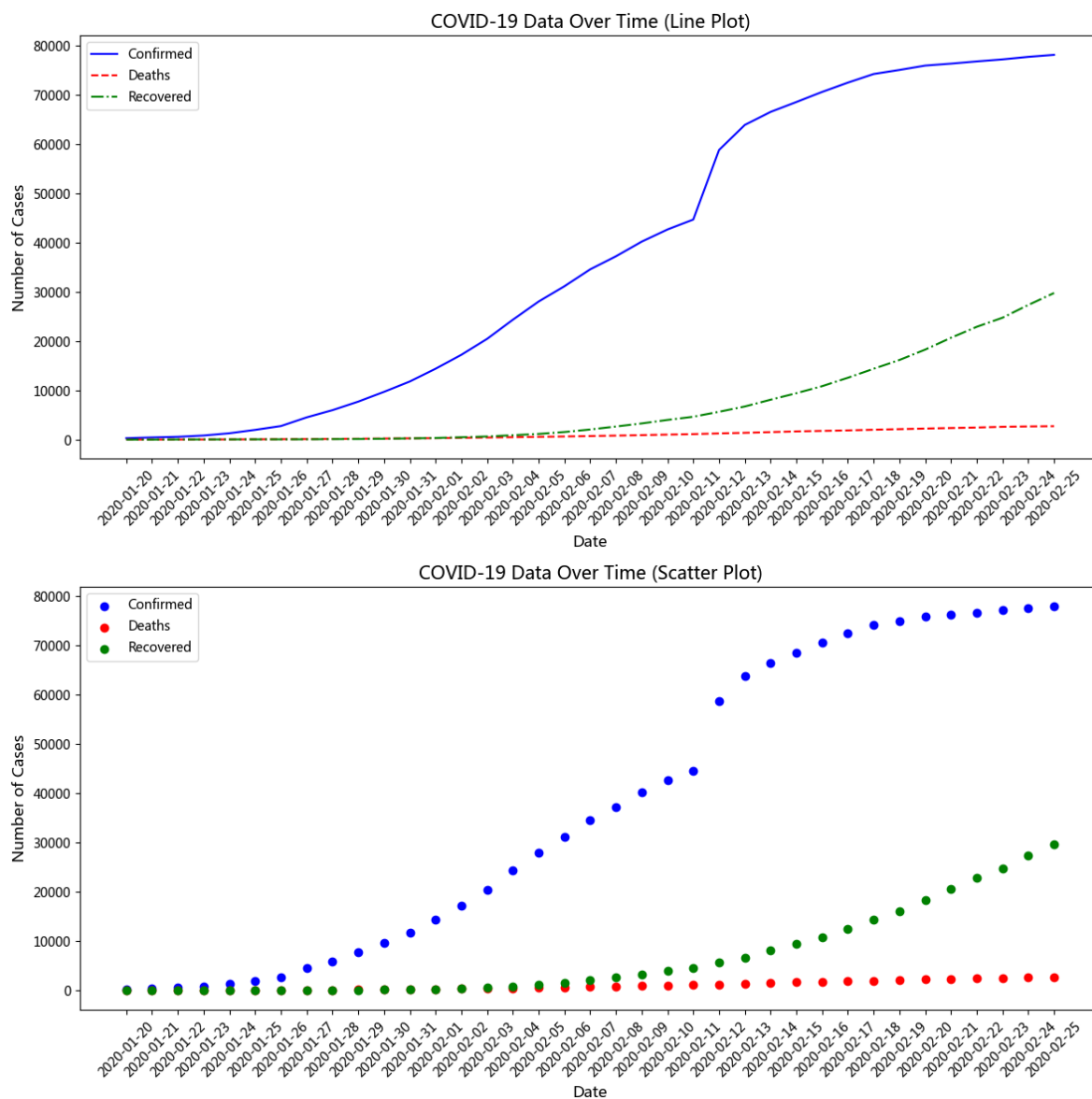
效果展示

COVID-19 Data Over Time (Line Plot)

COVID-19 Data Over Time (Scatter Plot)

思考

　　本题中使用折线图能够更好的表示确诊人数、死亡人数和治愈人数随着时间推移的变化趋势，而散点图更适合用来表示坐标系中数据的分布情况，不符合题目的客观背景。因此我认为折线图更适合。

第二题

代码

```python
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import matplotlib.font_manager as fm

data_world = pd.read_excel("dataset/covid19_data.xls", sheet_name="data_world")

fm.findfont(fm.FontProperties(fname="Microsoft Yahei"))
plt.rcParams["font.sans-serif"] = ["Microsoft Yahei"]

plt.rcParams["axes.unicode_minus"] = False
```

```
top_countries = data_world.nlargest(4, 'confirm')


fig, axes = plt.subplots(2, 2, figsize=(14, 14))
axes = axes.flatten()


for i, (index, row) in enumerate(top_countries.iterrows()):
    labels = ['Confirm', 'Dead', 'Heal', 'Suspect']
    sizes = [row['confirm'], row['dead'], row['heal'], row['suspect']]
    axes[i].pie(sizes, labels=labels, autopct='%1.1f%%', startangle=140)
    axes[i].set_title(f'COVID-19 Distribution in {row["country"]}', fontsize=14)

plt.tight_layout()
plt.savefig("res/1-2.png")
```
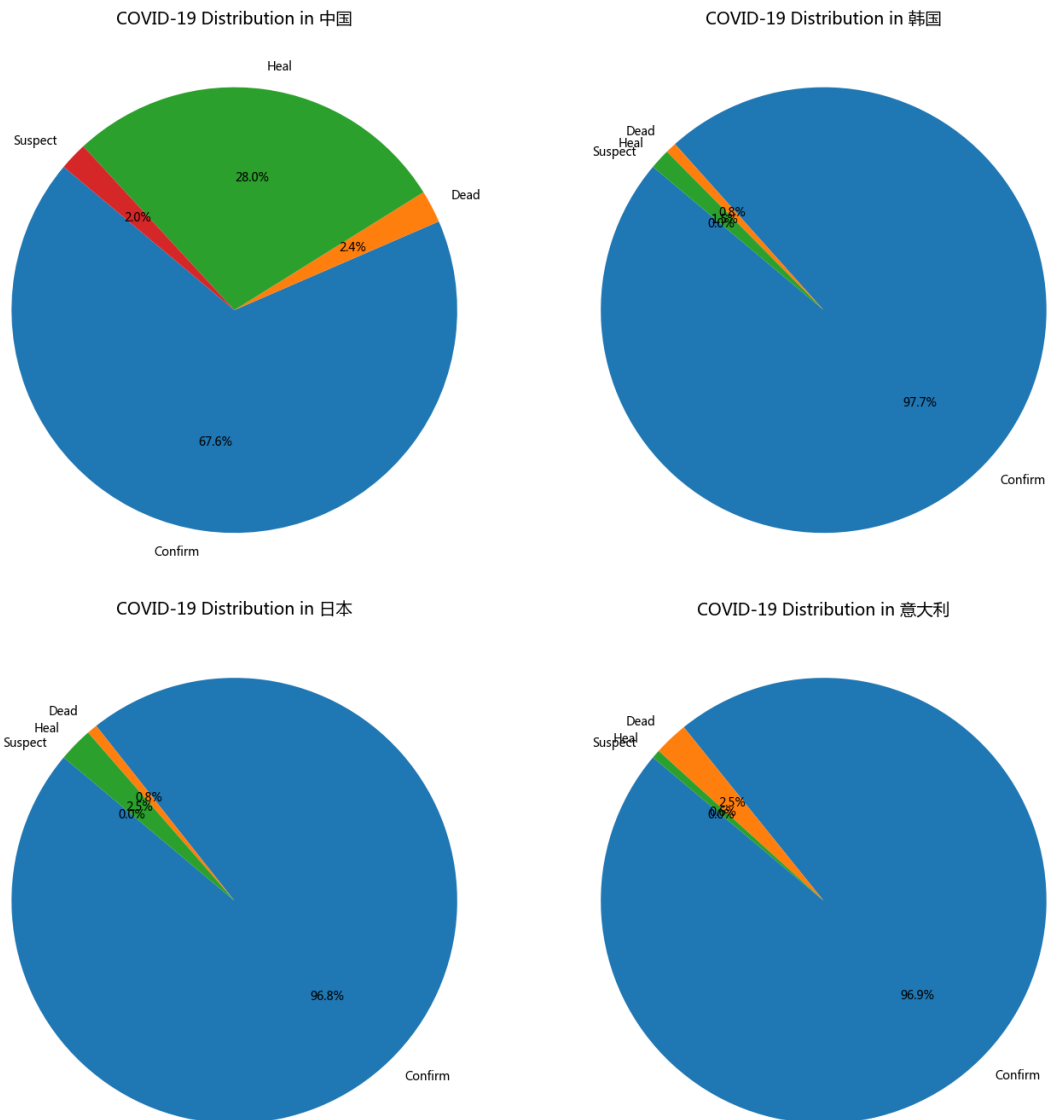
效果展示

COVID-19 Distribution in 中国

COVID-19 Distribution in 韩国

COVID-19 Distribution in 日本

COVID-19 Distribution in 意大利

第三题

```python
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import matplotlib.font_manager as fm


current_prov = pd.read_excel("dataset/covid19_data.xls",
sheet_name="current_prov")
fm.findfont(fm.FontProperties(fname="Microsoft Yahei"))
plt.rcParams["font.sans-serif"] = ["Microsoft Yahei"]

plt.rcParams["axes.unicode_minus"] = False


provinces = current_prov["province"]
positions = np.arange(len(provinces))
width = 0.25
fig, axes = plt.subplots(2, 1, figsize=(12, 10))

axes[0].bar(positions, current_prov['confirm'], width, color='blue')
axes[0].set_title('Histogram of Confirmed Cases by Province', fontsize=14)
axes[0].set_xlabel('Province', fontsize=12)
axes[0].set_ylabel('Number of Confirmed Cases', fontsize=12)
axes[0].set_xticks(positions)
axes[0].set_xticklabels(provinces, rotation=45)

axes[1].bar(positions - width, current_prov['confirm'], width, label='Confirmed',
color='blue')
axes[1].bar(positions, current_prov['dead'], width, label='Dead', color='red')
axes[1].bar(positions + width, current_prov['heal'], width, label='Healed',
color='green')
axes[1].set_title('Bar Chart of Confirmed Cases by Province', fontsize=14)
axes[1].set_xlabel('Province', fontsize=12)
axes[1].set_ylabel('Number of Confirmed Cases', fontsize=12)
axes[1].set_xticks(positions)
axes[1].set_xticklabels(provinces, rotation=45)
axes[1].legend()

plt.tight_layout()
plt.savefig("res/1-3.png")
```
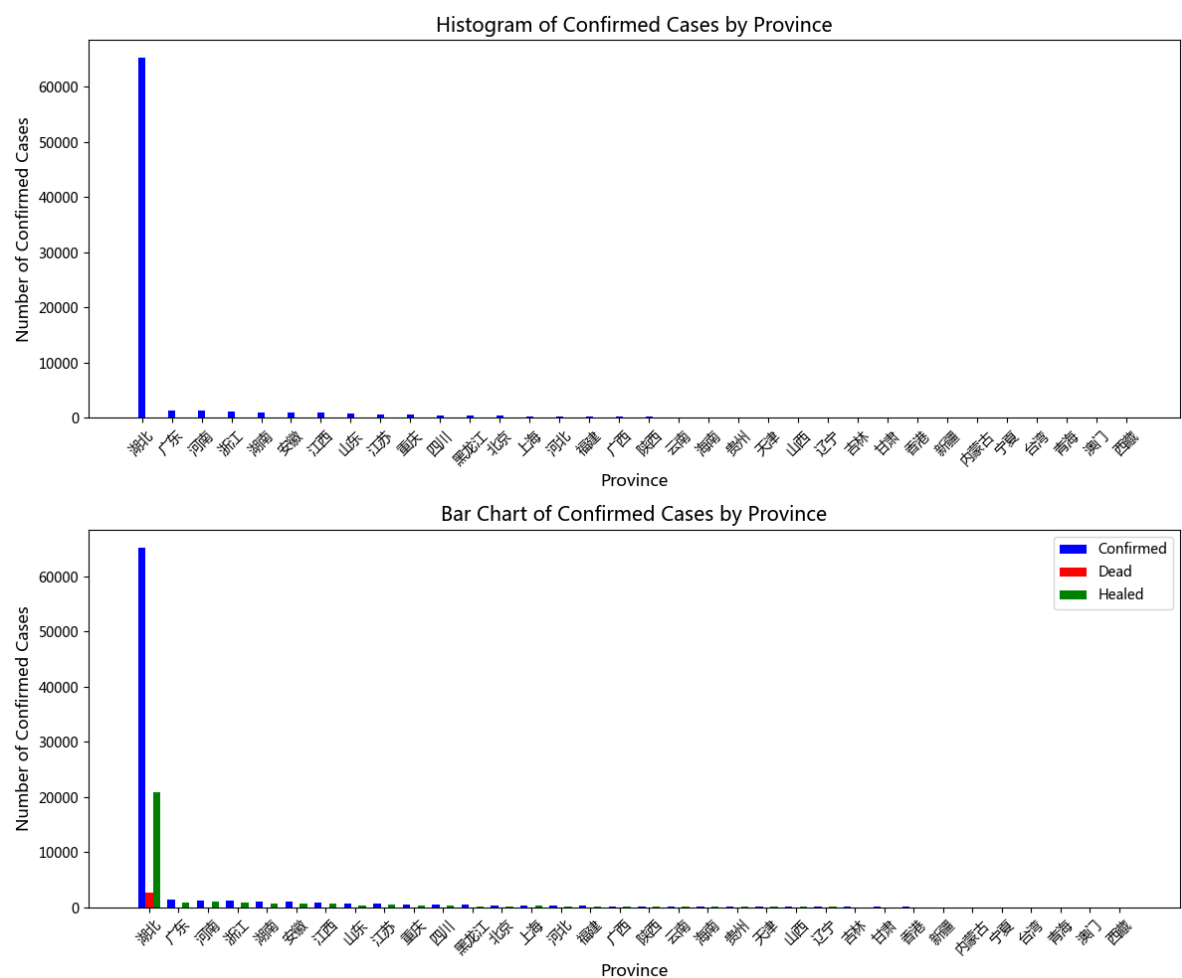
效果展示

Histogram of Confirmed Cases by Province

Bar Chart of Confirmed Cases by Province

思考

因为本题要求绘制不同省份之间确诊，死亡和治愈人数之间的关系，条形图通过长度能够很好的衡量不同省份之间相同变量之间的差异，以及同一省份不同变量之间的差异。

而直方图是用来可视化数据频率，中位数等概率特征，一般用横轴表示数据类型，纵轴表示分布情况。直方图是数值数据分布的精确图形表示。这是一个连续变量（定量变量）的概率分布的估计。在本题目中各个省份之间相互独立，使用直方图的可视化并无意义。

因此我认为应该使用条形图进行数据可视化。