

Module 4: (*Template*)

RENAME THE FILE TO INCLUDE YOUR COMPANY, GROUP NUMBER, AND LAST NAMES

E.G. KAMEN1_GROVES_MODULE_4.IPYNB

Team Members:

Julena and William

Project Title:

COVID-19 Italy 2020

Project Goal:

This project seeks to model COVID-19 (Italy 2020) with SIR to predict the future of this disease.

Disease Background:

Using your assigned disease, fill in the following bullet points.

- Prevalence & incidence
 - There was a very large increase in total mortality in Italy in 2020: from March to December, there were ~ 99,334 “excess” deaths compared to expected baselines — about +18.8% over normal. (<https://pubmed.ncbi.nlm.nih.gov/35481574/>)
 - The official COVID-19 death toll in that period under-reported the total impact: many of the excess deaths weren’t recorded in surveillance systems, indicating that the “true” disease burden was much higher. (<https://pubmed.ncbi.nlm.nih.gov/35481574/>)
 - Modeling estimates suggest that much more of the Italian population was infected than official case counts show: for example, one demographic-scaling model estimated ~1.4 million total infections in Italy by mid-2020, which was many times higher than confirmed cases (https://arxiv.org/abs/2004.12836?utm_source=chatgpt.com)
- Economic burden
 - The total estimated permanent productivity loss due to premature mortality from COVID-19 in Italy (up to 28 April 2020) was about EUR 300 million.
 - The estimated temporary productivity loss (due to work absenteeism) during that same period was about EUR 100 million. (<https://pmc.ncbi.nlm.nih.gov/articles/PMC7345321/>)
- Risk factors (genetic, lifestyle) & Societal determinants
 - Your risk of catching COVID-19 is higher when you live with someone who has it, spend time in poorly ventilated or crowded indoor spaces, or have close contact (especially for >30 minutes) with an infected person.
 - Older age (65+), babies under 6 months, and people with chronic conditions (like heart disease, lung disease, kidney/liver disease, obesity, or a weakened immune

system) are more likely to develop serious COVID-19 illness.
(<https://www.mayoclinic.org/diseases-conditions/coronavirus/symptoms-causes/syc-20479963>)

- Symptoms
 - Most common symptoms: fever, cough, tiredness, loss of taste or smell.
 - Less common symptoms: sore throat, headache, aches and pains, diarrhoea, a rash on skin, or discolouration of fingers or toes, red or irritated eyes.
 - Serious symptoms: difficulty breathing or shortness of breath, loss of speech or mobility, or confusion, chest pain.
(https://www.who.int/health-topics/coronavirus#tab=tab_3)
- Diagnosis
 - You can use at-home FDA-approved COVID-19 tests, or go through your healthcare provider, especially if you have symptoms or have been exposed.
 - If you're symptomatic, test right away. If you were exposed but have no symptoms, wait ~5 days after exposure before testing to improve accuracy.
(<https://www.mayoclinic.org/diseases-conditions/coronavirus/symptoms-causes/syc-20479963>)
- Biological mechanisms (anatomy, organ physiology, cell & molecular physiology)
 - How the virus enters the body: SARS-CoV-2 attaches to ACE2 receptors on human cells (especially in the lungs) and uses host enzymes like TMPRSS2 to enter and begin infection.
 - How it causes damage: Infection triggers an intense inflammatory response ("cytokine storm") and can injure blood vessels, leading to problems like clotting and organ damage.
 - Why multiple organs are affected: Because ACE2 receptors exist in the lungs, heart, kidneys, intestines and blood vessels, the virus can spread beyond the lungs and cause widespread symptoms.
(<https://pmc.ncbi.nlm.nih.gov/articles/PMC8592035/>)

Dataset:

- Github page: <https://github.com/CSSEGISandData/COVID-19>
- The dataset analyzed is the Italy COVID-19 cumulative case data for 2020, sourced from the COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University. It was collected as part of JHU's global COVID-19 tracking dashboard, which gathered daily reports of confirmed cases from sources like the World Health Organization (WHO), European CDC, and national news and health agencies. Our instructor cleaned up the data so every group has a 2-column data-frame: date and confirmed cases. The dates range from March 2020 to July 2020, and the range of people infected goes from 1,694 to 247,537. From these cumulative counts, we can calculate daily new cases and estimate the number of susceptible, infectious, and recovered individuals over time. The data are measured as counts of people and are organized by date, which makes it suitable for modeling the spread of COVID-19 using approaches like the SIR model.

Cumulative infections over time

```

# infections over time
import pandas as pd
import matplotlib.pyplot as plt

# Load the CSV file
df = pd.read_csv("/Users/julenapatel/Downloads/Computational
BME/Module 4/Data/covid_italy_data_march_july_2020_cumulative.csv")

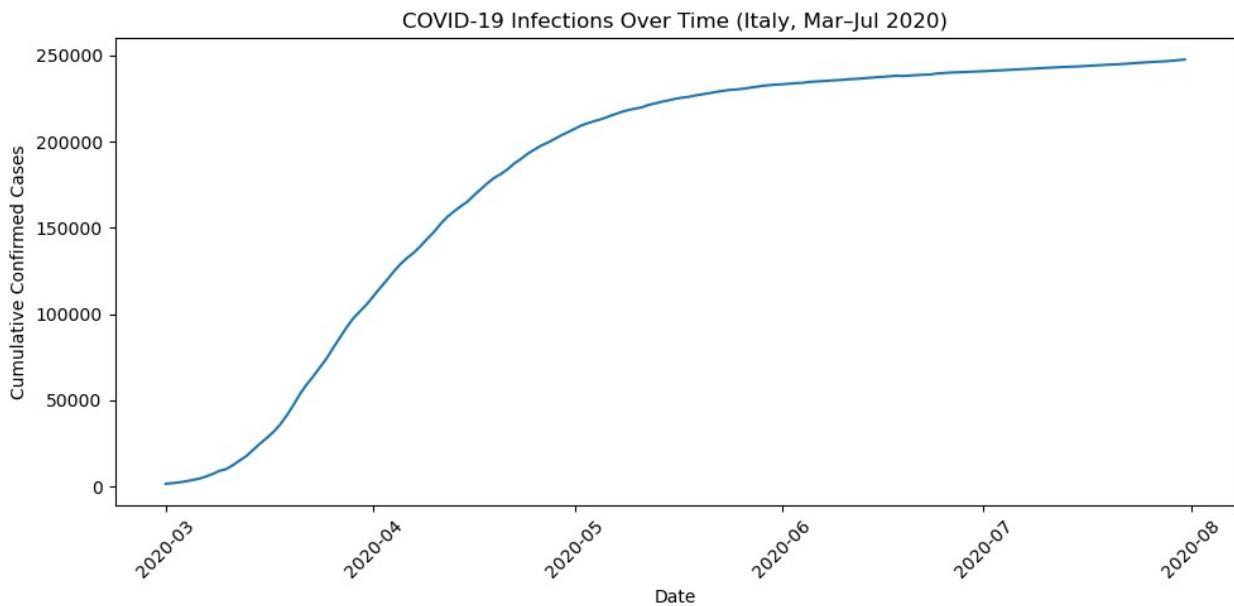
# Convert date column to datetime
df['date'] = pd.to_datetime(df['date'])

# Plot infections over time
plt.figure(figsize=(10,5))
plt.plot(df['date'], df['confirmed_cases'])

plt.xlabel("Date")
plt.ylabel("Cumulative Confirmed Cases")
plt.title("COVID-19 Infections Over Time (Italy, Mar-Jul 2020)")
plt.xticks(rotation=45)

plt.tight_layout()
plt.show()

```



Non-cumulative infections over time

```

import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

# 1. LOAD YOUR DATASET
df = pd.read_csv("/Users/julenapatel/Downloads/Computational

```

```

BME/Module 4/Data/covid_italy_data_march_july_2020_cumulative.csv")
df['date'] = pd.to_datetime(df['date'])

# 2. Convert cumulative cases to S, I, R
def convert_cumulative_to_SIR(df, date_col='date',
                               cumulative_col='cumulative_cases',
                               population=60_000_000,
                               infectious_period=8, recovered_col=None,
                               new_case_col='new_cases',
                               I_col='I_est', R_col='R_est', S_col='S_est'):
    """
        Convert cumulative reported cases into S, I, R estimates for SIR
        modeling.
        - new_cases = diff(cumulative)
        - I_est = rolling sum(new_cases, window=infectious_period)
        - R_est = cumulative cases MINUS currently infectious
        - S_est = population - I_est - R_est
    """

    Returns a copy of the dataframe with the added columns.
    """
    df = df.copy()

    # Ensure date column sorted if present
    if date_col in df.columns:
        df[date_col] = pd.to_datetime(df[date_col])
        df = df.sort_values(date_col).reset_index(drop=True)

    if cumulative_col not in df.columns:
        raise ValueError(f"Column '{cumulative_col}' not found in
dataframe.")

    # Compute new cases (incident)
    df[new_case_col] =
    df[cumulative_col].diff().fillna(df[cumulative_col].iloc[0])
    df[new_case_col] = df[new_case_col].clip(lower=0)

    # Estimate I(t) as rolling sum over infectious_period
    if infectious_period <= 0:
        raise ValueError("infectious_period must be positive
integer.")
    df[I_col] = df[new_case_col].rolling(window=infectious_period,
                                         min_periods=1).sum()

    # Estimate R(t) - Total cumulative cases MINUS currently
    # infectious
    if recovered_col and recovered_col in df.columns:
        df[R_col] = df[recovered_col].fillna(0)
    else:
        df[R_col] = df[cumulative_col] - df[I_col]

```

```

df[R_col] = df[R_col].clip(lower=0)

# Compute S(t)
df[S_col] = population - df[I_col] - df[R_col]
df[S_col] = df[S_col].clip(lower=0)

# Ensure numeric and non-negative
for col in [new_case_col, I_col, R_col, S_col]:
    df[col] = df[col].astype(float).clip(lower=0)

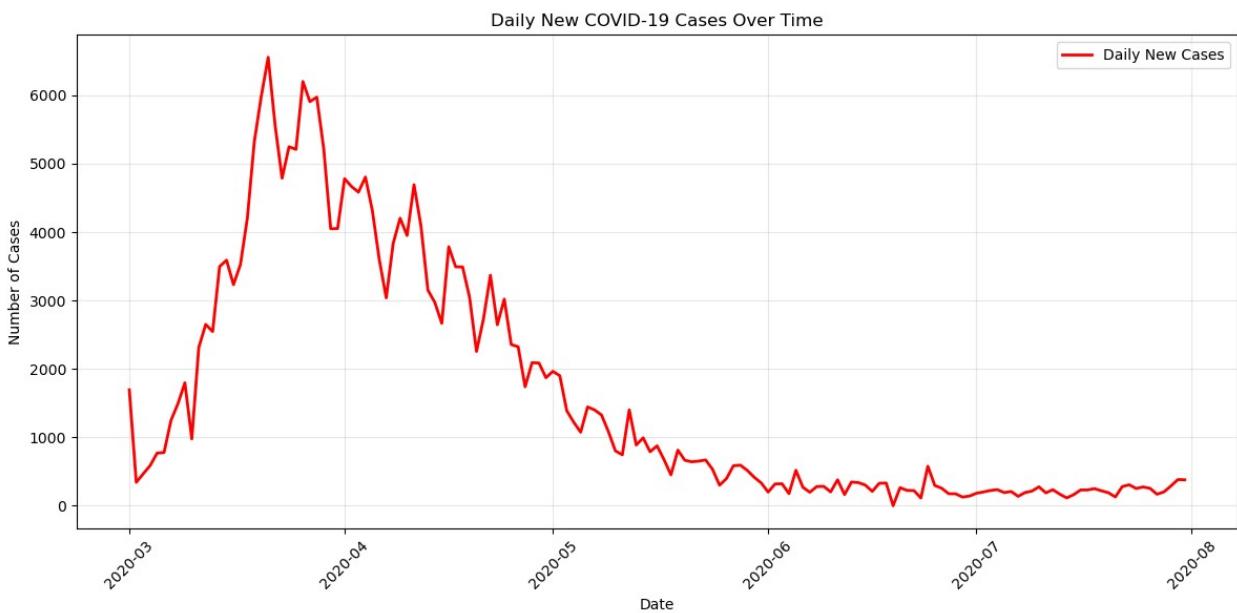
return df

# Apply the conversion
df = convert_cumulative_to_SIR(df, cumulative_col='confirmed_cases')

# Extract the new_cases (daily, non-cumulative) column
daily_cases = df['new_cases']

# 3. Plot daily new cases over time
plt.figure(figsize=(12,6))
plt.plot(df['date'], daily_cases, label='Daily New Cases',
color='red', linewidth=2)
plt.xlabel("Date")
plt.ylabel("Number of Cases")
plt.title("Daily New COVID-19 Cases Over Time")
plt.legend()
plt.xticks(rotation=45)
plt.grid(True, alpha=0.3)
plt.tight_layout()
plt.show()

```



SIR MODEL

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

# 1. LOAD YOUR DATASET
df = pd.read_csv("/Users/julenapatel/Downloads/Computational
BME/Module 4/Data/covid_italy_data_march_july_2020_cumulative.csv")
df['date'] = pd.to_datetime(df['date'])

# 2. Convert cumulative cases to S, I, R
def convert_cumulative_to_SIR(df, date_col='date',
cumulative_col='cumulative_cases',
population=60_000_000,
infectious_period=8, recovered_col=None,
new_case_col='new_cases',
I_col='I_est', R_col='R_est', S_col='S_est'):
    """
    Convert cumulative reported cases into S, I, R estimates for SIR
    modeling.
    - new_cases = diff(cumulative)
    - I_est = rolling sum(new_cases, window=infectious_period)
    - R_est = cumulative cases MINUS currently infectious
    - S_est = population - I_est - R_est
    Returns a copy of the dataframe with the added columns.
    """
    df = df.copy()

    # Ensure date column sorted if present
    if date_col in df.columns:
        df[date_col] = pd.to_datetime(df[date_col])
        df = df.sort_values(date_col).reset_index(drop=True)

    if cumulative_col not in df.columns:
        raise ValueError(f"Column '{cumulative_col}' not found in
dataframe.")

    # Compute new cases (incident)
    df[new_case_col] =
df[cumulative_col].diff().fillna(df[cumulative_col].iloc[0])
    df[new_case_col] = df[new_case_col].clip(lower=0)

    # Estimate I(t) as rolling sum over infectious_period
    if infectious_period <= 0:
        raise ValueError("infectious_period must be positive
integer.")
    df[I_col] = df[new_case_col].rolling(window=infectious_period,
min_periods=1).sum()
```

```

# Estimate R(t) - Total cumulative cases MINUS currently
infectious
    if recovered_col and recovered_col in df.columns:
        df[R_col] = df[recovered_col].fillna(0)
    else:
        df[R_col] = df[cumulative_col] - df[I_col]
        df[R_col] = df[R_col].clip(lower=0)

# Compute S(t)
df[S_col] = population - df[I_col] - df[R_col]
df[S_col] = df[S_col].clip(lower=0)

# Ensure numeric and non-negative
for col in [new_case_col, I_col, R_col, S_col]:
    df[col] = df[col].astype(float).clip(lower=0)

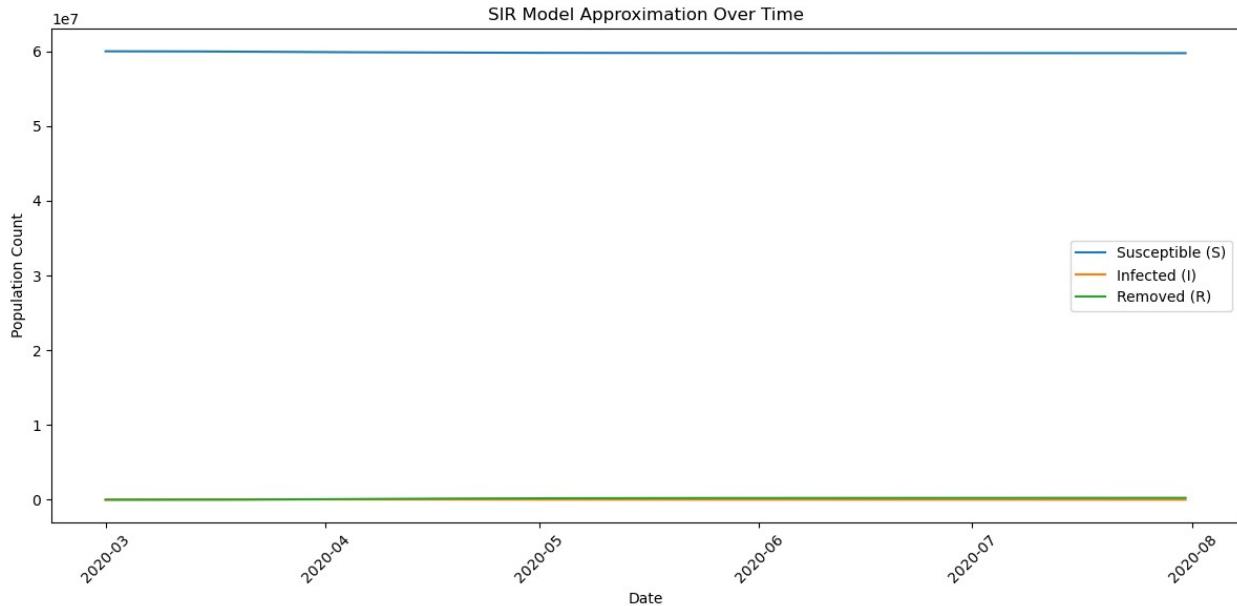
return df

# Apply the conversion
df = convert_cumulative_to_SIR(df, cumulative_col='confirmed_cases')

# Extract S, I, R columns from the returned dataframe
S = df['S_est']
I = df['I_est']
R = df['R_est']

# 3. Plot S, I, R over time
plt.figure(figsize=(12,6))
plt.plot(df['date'], S, label='Susceptible (S)')
plt.plot(df['date'], I, label='Infected (I)')
plt.plot(df['date'], R, label='Removed (R)')
plt.xlabel("Date")
plt.ylabel("Population Count")
plt.title("SIR Model Approximation Over Time")
plt.legend()
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

```



Data Analysis:

Methods

IN A SUMMARY, DESCRIBE THE METHODS YOU USED TO ANALYZE AND MODEL THE DATA.

Analysis

(Describe how you analyzed the data. This is where you should intersperse your Python code so that anyone reading this can run your code to perform the analysis that you did, generate your figures, etc.)

```
## PYTHON CODE TO BUILD AND FIT AN SIR MODEL GOES HERE. INTERSPERSE
COMMENTS (IN MARKDOWN) TO EXPLAIN EACH STEP.
```

Verify and validate your analysis:

(Describe how you checked to see that your analysis gave you an answer that you believe (verify). Describe how you determined if your analysis gave you an answer that is supported by other evidence (e.g., a published paper).

Conclusions and Ethical Implications:

(Think about the answer your analysis generated, draw conclusions related to your overarching question, and discuss the ethical implications of your conclusions.

Limitations and Future Work:

(Think about the answer your analysis generated, draw conclusions related to your overarching question, and discuss the ethical implications of your conclusions.

NOTES FROM YOUR TEAM:

This is where our team is taking notes and recording activity.

QUESTIONS FOR YOUR TA:

These are questions we have for our TA.

- In our SIR model, are infected (I) and removed (R) supposed to overlap? Did I miss a value I was supposed to enter?