A Course Based Project Report on

Predictive Analytics for Disease Spread Using COVID-19 Data

Submitted to the

Department of CSE-(CyS, DS) and AI&DS

in partial fulfilment of the requirements for the completion of course

MODELS IN DATA SCIENCE LABORATORY(22PC2DS301)

BACHELOR OF TECHNOLOGY

IN

Department of CSE-(CyS, DS) and AI&DS

Submitted by

C. SAI CHARITH	23071A6778
CH. KARTHIK	23071A6780
G. KIRTHAN VIVEK	23071A6785
G. RUTHVIK	23071A6789

Under the guidance of

Mrs. N.MADHURI

(Course Instructor)

Assistant Professor, Department of CSE-(CYS,DS) AND AI&DS
VNRVJIET

Department of CSE-(CyS, DS) and AI&DS

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING & TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade, NBA

Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India

November 2025

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade, NBA Accredited for CE, EEE, ME, ECE, CSE, EIE, IT B. Tech Courses, Approved by AICTE, New Delhi, Affiliated to JNTUH, Recognized as "College with Potential for Excellence" by UGC, ISO 9001:2015 Certified, QS I GUAGE Diamond Rated

Vignana Jyothi Nagar, Pragathi Nagar, Nizampet(SO), Hyderabad-500090, TS, India

Department of CSE-(CyS, DS) and AI&DS

CERTIFICATE

This is to certify that the project report entitled "**Predictive Analytics for disease spread using COVID – 19 Dataset"** is a bonafide work done under our supervision and is being submitted by **Mr. C Sai Charith (23071A6778), Mr CH. Karthik (23071A6780) , Mr. Kirthan Vivek (23071A6785) , Mr. G. Ruthvik (23071A6789)** in partial fulfilment for the award of the degree of **Bachelor of Technology** in **CSE-(CyS, DS) and AI&DS**, of the VNRVJIET, Hyderabadduring the academic year 2025-2026.

Mrs.N.Madhuri
Assistant Professor
Dept of CSE-(CyS, DS) and AI&DS

Dr. T. Sunil Kumar
Professor & HOD
Dept of CSE-(CyS, DS)and AI&DS

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade,
Vignana Jyothi Nagar, Pragathi Nagar, Nizampet(SO), Hyderabad-500090, TS, India

Department of CSE-(CyS, DS) and AI&DS

DECLARATION

We declare that the course based project work entitled **"Predictive Analytics for disease spread using COVID – 19 Dataset"** submitted in the Department of **CSE-(CyS, DS) and AI&DS**, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, in partial fulfilment of the requirement for the award of the degree of **Bachelor of Technology in CSE-(CyS, DS) and AI&DS** is a bonafide record of our own work carried out under the supervision of **Mrs.N.Madhuri, Assistant Professor, Department of CSE-(CyS, DS) and AI&DS , VNRVJIET.**Also, wedeclare that the matter embodied in this thesis has not been submitted by us in full or in any part there of for the award of any degree/diploma of any other institution or university previously.

Place: Hyderabad.

C. SAI CHARITH	CH. KARTHIK	G. KIRTHAN VIVEK	G. RUTHVIK
(23071A6778)	(23071A6780)	(23071A6785)	(23071A6789)

ACKNOWLEDGEMENT

We express our deep sense of gratitude to our beloved President, Sri. D. Suresh Babu, VNR Vignana Jyothi Institute of Engineering & Technology for the valuable guidance and for permitting us to carry out this project.

With immense pleasure, we record our deep sense of gratitude to our beloved Principal, Dr. C.D Naidu, for permitting us to carry out this project.

We express our deep sense of gratitude to our beloved Professor Dr.T.SUNIL KUMAR, Professor and Head, Department of CSE-(CyS, DS) and AI&DS , VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad- 500090 for the valuable guidance and suggestions, keen interest and through encouragement extended throughout the period of project work.

We take immense pleasure to express our deep sense of gratitude to our beloved Guide, Mrs. N. MADHURI, Assistant Professor in CSE-(CyS, DS) and AI&DS, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad, for her valuable suggestions and rare insights, for constant source of encouragement and inspiration throughout my project work.

We express our thanks to all those who contributed for the successful completion ofour project work.

C.SAI CHARITH	(23071A6778)
CH. KARTHIK	(23071A6780)
G. KIRTHAN VIVEK	(23071A6785)
G. RUTHVIK	(23071A6789)

TABLE OF CONTENTS

ABSTRACT

The COVID-19 pandemic has profoundly affected global health, economics, and society, emphasizing the importance of reliable analytical and predictive models. This study utilizes a SEIR (Susceptible–Exposed–Infected–Recovered) epidemiological model in combination with Machine Learning algorithms to analyze and forecast the spread of COVID-19.

The SEIR model incorporates an Exposed (E) compartment to capture the incubation period, offering a more realistic representation of COVID-19 dynamics than the basic SIR model. The model parameters—β (infection rate), σ (incubation rate), and γ (recovery rate)—were adjusted to simulate different real-world conditions such as lockdowns and vaccination phases.

In parallel, Machine Learning techniques including Linear Regression and XGBoost Regression were implemented to forecast short-term and medium-term case trends using real-time data from Our World in Data(OWID). The models provided future case predictions for multiple countries, demonstrating strong performance, especially from XGBoost due to its non-linear pattern learning capability.

The integration of SEIR modeling and machine learning enabled a hybrid approach that combines epidemiological theory with data-driven forecasting. This fusion enhances the interpretability, flexibility, and predictive accuracy of pandemic modeling. The system was deployed as an interactive Streamlit dashboard for live visualization and forecasting, allowing policymakers, researchers, and the public to explore real-time trends and predictions.

CHAPTER-1

INTRODUCTION

1.1 Background

The COVID-19 pandemic has drastically altered global society, health, and economics, leading to an urgent demand for analytical tools capable of forecasting infection trends and guiding policy interventions. Traditional epidemiological models such as SIR and SEIR have long served as essential frameworks for understanding infectious disease transmission.

Among them, the SEIR (Susceptible–Exposed–Infected–Recovered) model extends the basic SIR formulation by including an exposed compartment, which represents individuals in the incubation period who are infected but not yet infectious. This enhancement makes the SEIR model particularly suitable for diseases like COVID-19, which exhibit a measurable delay between exposure and symptom onset.

1.2 Need for the Study

COVID-19's unpredictable behavior—shaped by factors such as viral mutations, government policies, and public compliance—demands adaptable models that blend theoretical epidemiology with empirical data. The integration of machine learning algorithms with the SEIR framework allows for improved accuracy, adaptability, and continuous updating as new data becomes available.

1.3 Objectives

- To collect and visualize real-time COVID-19 data using public APIs.
- To simulate infection spread using the SEIR model with adjustable parameters.
- To forecast near-future cases using machine learning models like Linear Regression and XGBoost.
- To compare classical epidemiological predictions with AI-driven forecasts.
- To develop an interactive Streamlit dashboard integrating all functionalities.

1.4 Scope

The project focuses on combining SEIR modeling and machine learning for the prediction and visualization of COVID-19 trends. The approach provides both theoretical understanding and data-driven forecasting capabilities. The same framework can be adapted for other infectious diseases with minimal modification.

1.5 Tools and Technologies Used

Category	Tools
Programming Language	Python 3.10+
Libraries	NumPy, Pandas, Matplotlib, SciPy, Scikit-learn, XGBoost
Visualization	Streamlit
Data Source	Our World in Data (OWID) API
Development Environment	Google Colab / VS Code

CHAPTER-2

Method

2.1 Overview

This study adopted a **multi-faceted methodology** combining *exploratory data analysis (EDA)*, *epidemiological modeling (SEIR)*, and *machine learning-based forecasting* to analyze and predict the spread of COVID-19. The approach was designed to capture the temporal evolution of the outbreak, estimate key parameters governing disease transmission, and provide short-term predictive insights.

2.2 Exploratory Data Analysis (EDA)

The first stage involved performing EDA on real-time COVID-19 case data obtained from the **Our World in Data (OWID)** repository. The dataset includes daily case counts, deaths, recoveries, and testing statistics for more than 200 countries.

Data preprocessing steps included:

- Handling missing and inconsistent values through interpolation and forward filling.
- Filtering country-specific data (e.g., India, China, Italy, Spain, the UK, and Singapore).
- Generating key metrics: **daily new cases, cumulative cases, recovery rate,** and **case fatality ratio (CFR)**.

Visualization through **Matplotlib** and **Seaborn** allowed identification of temporal patterns, spikes, and anomalies across different regions, illustrating how containment measures and vaccination efforts influenced infection trends.

2.3 Epidemiological Modeling – SEIR Framework

To model COVID-19 transmission dynamics, the SEIR (Susceptible–Exposed–Infected–Recovered) **framework was adopted. This model extends the classical SIR model by introducing an additional** Exposed (E) **compartment that represents individuals who have been infected but are not yet infectious — capturing COVID-19's incubation period more realistically.**

Model Equations:

Where:

- **S(t):** Susceptible population
- **E(t):** Exposed population (infected but not yet infectious)
- **I(t):** Infectious individuals
- **R(t):** Recovered or deceased individuals
- **β (beta):** Transmission rate
- **σ (sigma):** Incubation rate (1/incubation period)
- **γ (gamma):** Recovery rate
- **N:** Total population

Using numerical integration techniques (Runge–Kutta 4th order), the SEIR model simulated infection progression over time. Adjusting parameters β, σ, and γ allowed examination of the effects of interventions such as lockdowns or vaccination campaigns.

2.4 Machine Learning-Based Forecasting
To complement the SEIR model and capture non-linear trends in real data, machine learning regression models **were implemented:**

(a) Linear Regression
A baseline predictive model that fits a straight-line relationship between days and total cases. It is effective for short-term, stable growth patterns and quick trend detection.

(b) XGBoost Regression
An advanced gradient boosting algorithm capable of learning non-linear relationships **and handling irregular, wave-like case growth.**
This model provided more accurate and realistic short-term forecasts compared to traditional linear methods.

Both models were trained on the most recent 100–120 days of COVID-19 data for each country. The **Root Mean Square Error (RMSE)** metric was used to evaluate prediction accuracy.

2.5 Integration and Visualization
The final system integrates all modules into an interactive dashboard **built using** Streamlit**, featuring:**

- Real-time data fetching from OWID's API.
- Country selection dropdown for global comparison.
- Adjustable SEIR parameters (β, σ, γ).
- Machine learning model toggle (Linear Regression / XGBoost).
- Dynamic visualization of infection curves and 14-day forecasts aligned with the current date.

Dataset Link: https://www.kaggle.com/competitions/covid19-global-forecasting- week-4

CHAPTER-3

TEST CASES/ OUTPUT

```
from google.colab import files
uploaded = files.upload()
```

train.csv

```
train.csv(text/csv) - 1425515 bytes, last modified: 11/1/2025 - 100% done
Saving train.csv to train (1).csv
```

```
from google.colab import files
uploaded = files.upload()
```

test.csv

```
test.csv(text/csv) - 397182 bytes, last modified: 11/1/2025 - 100% done
Saving test.csv to test.csv
```

```
from google.colab import files
uploaded = files.upload()
```

submission.csv

```
submission.csv(text/csv) - 123521 bytes, last modified: 11/1/2025 - 100% done
Saving submission.csv to submission.csv
```

```
import pandas as pd
df = pd.read_csv('train.csv')
df.head(5)
```

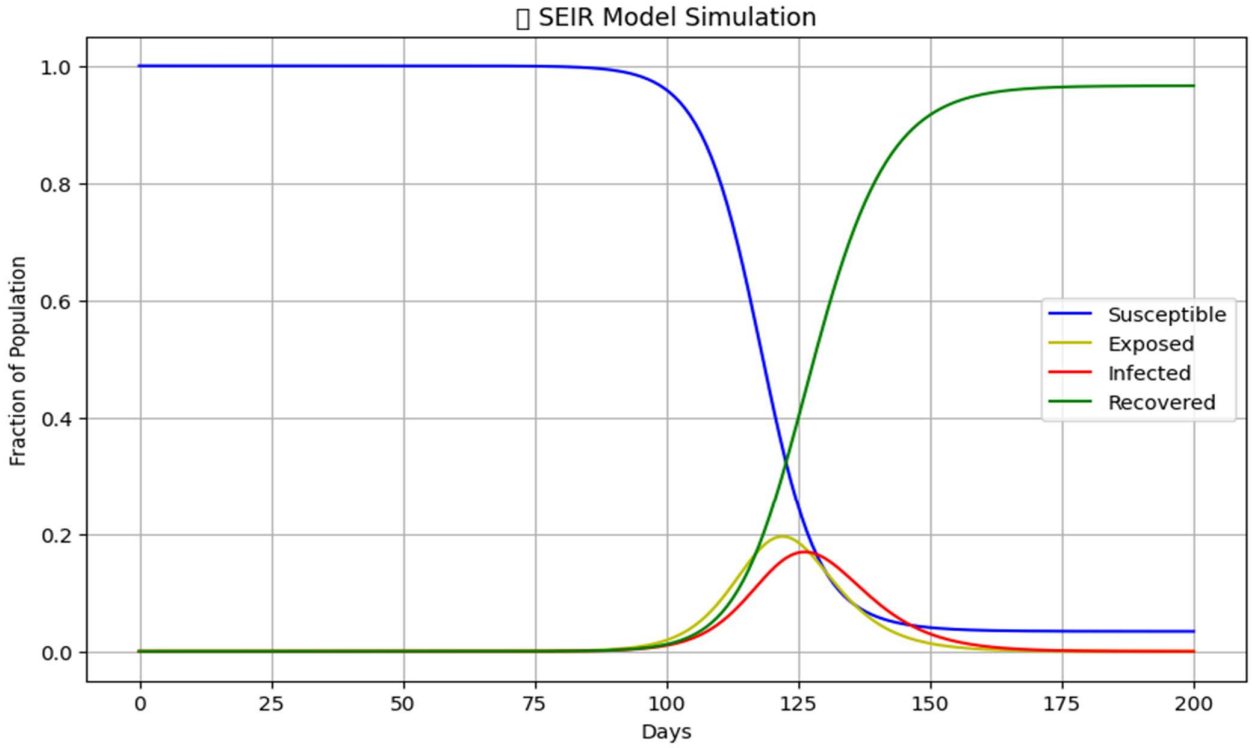
	Id	Province_State	Country_Region	Date	ConfirmedCases	Fatalities
0	1	NaN	Afghanistan	2020-01-22	0.0	0.0
1	2	NaN	Afghanistan	2020-01-23	0.0	0.0
2	3	NaN	Afghanistan	2020-01-24	0.0	0.0
3	4	NaN	Afghanistan	2020-01-25	0.0	0.0
4	5	NaN	Afghanistan	2020-01-26	0.0	0.0

Next steps: Generate code with df | New interactive sheet

 # -STEP 1 – SEIR Epidemic Model Simulation

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import integrate
def seir_equations(y, t, N, beta, sigma, gamma):
```

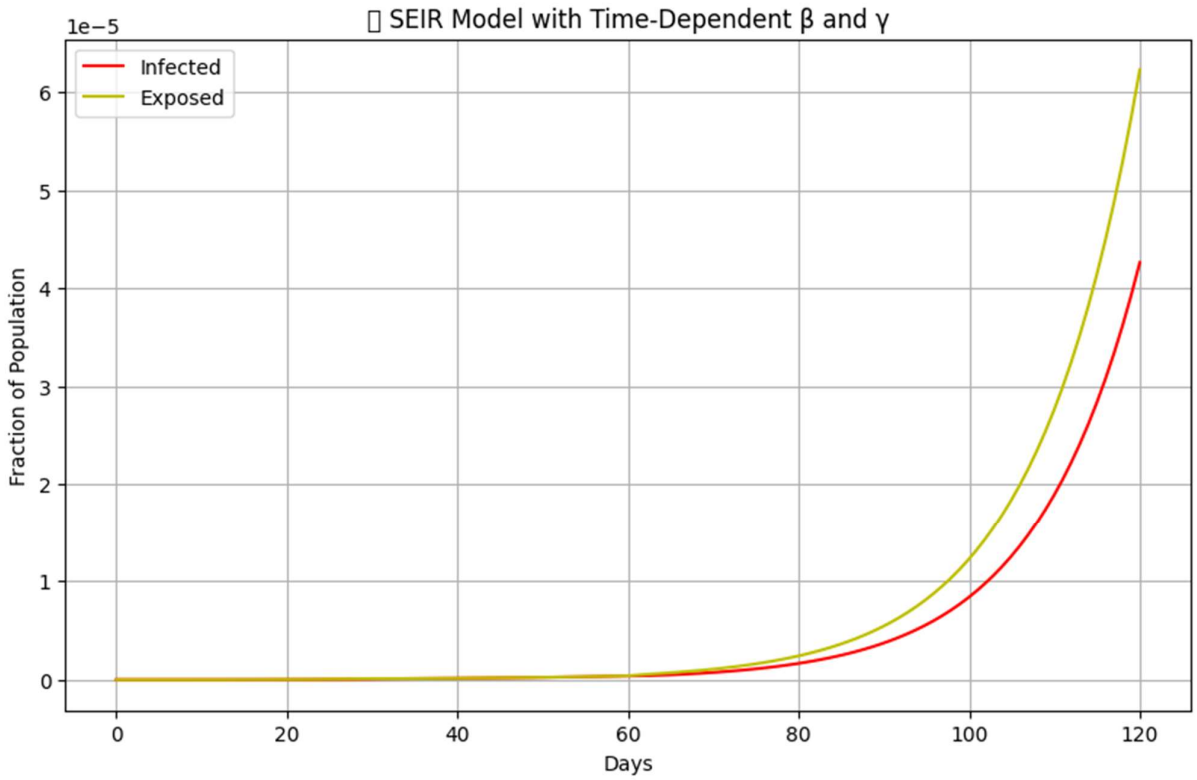
```
    S, E, I, R = y
    dSdt = -beta * S * I / N
    dEdt = beta * S * I / N - sigma * E
    dIdt = sigma * E - gamma * I
    dRdt = gamma * I
    return dSdt, dEdt, dIdt, dRdt

def run_seir_model(N, E0, I0, R0, beta, sigma, gamma, days):
    """Integrates the SEIR equations over 'days' days."""
    S0 = N - E0 - I0 - R0
    y0 = S0, E0, I0, R0
    t = np.linspace(0, days, days)
    ret = integrate.odeint(seir_equations, y0, t, args=(N, beta, sigma, gamma))
    S, E, I, R = ret.T
    return t, S, E, I, R

N = 7.8e9          # world population
E0 = 10            # initial exposed
I0 = 1             # initial infected
R0 = 0             # initial recovered
beta = 0.7         # infection rate
sigma = 1/5.2      # rate of progression (1/incubation period)
gamma = 0.2        # recovery rate
days = 200         # simulation length

t, S, E, I, R = run_seir_model(N, E0, I0, R0, beta, sigma, gamma, days)
plt.figure(figsize=(10,6))
plt.plot(t, S/N, 'b', label='Susceptible')
plt.plot(t, E/N, 'y', label='Exposed')
plt.plot(t, I/N, 'r', label='Infected')
plt.plot(t, R/N, 'g', label='Recovered')
plt.title("□ SEIR Model Simulation")
plt.xlabel("Days")
plt.ylabel("Fraction of Population")
plt.legend()
plt.grid(True)
plt.show()
```



#STEP 2 – SEIR Model with Time-Dependent β and γ

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import integrate
def beta_t(t):
    """Infection rate changes with time."""
    if t < 30:
        return 0.6
    elif t < 60:
        return 0.25
    else:
        return 0.4
def gamma_t(t):
    """Recovery rate may improve slowly."""
    return 0.15 + (0.05 if t > 60 else 0)
def seir_time_varying(y, t, N, sigma):
    S, E, I, R = y
    beta = beta_t(t)
    gamma = gamma_t(t)
    dSdt = -beta * S * I / N
    dEdt = beta * S * I / N - sigma * E
    dIdt = sigma * E - gamma * I
    dRdt = gamma * I
    return dSdt, dEdt, dIdt, dRdt
```

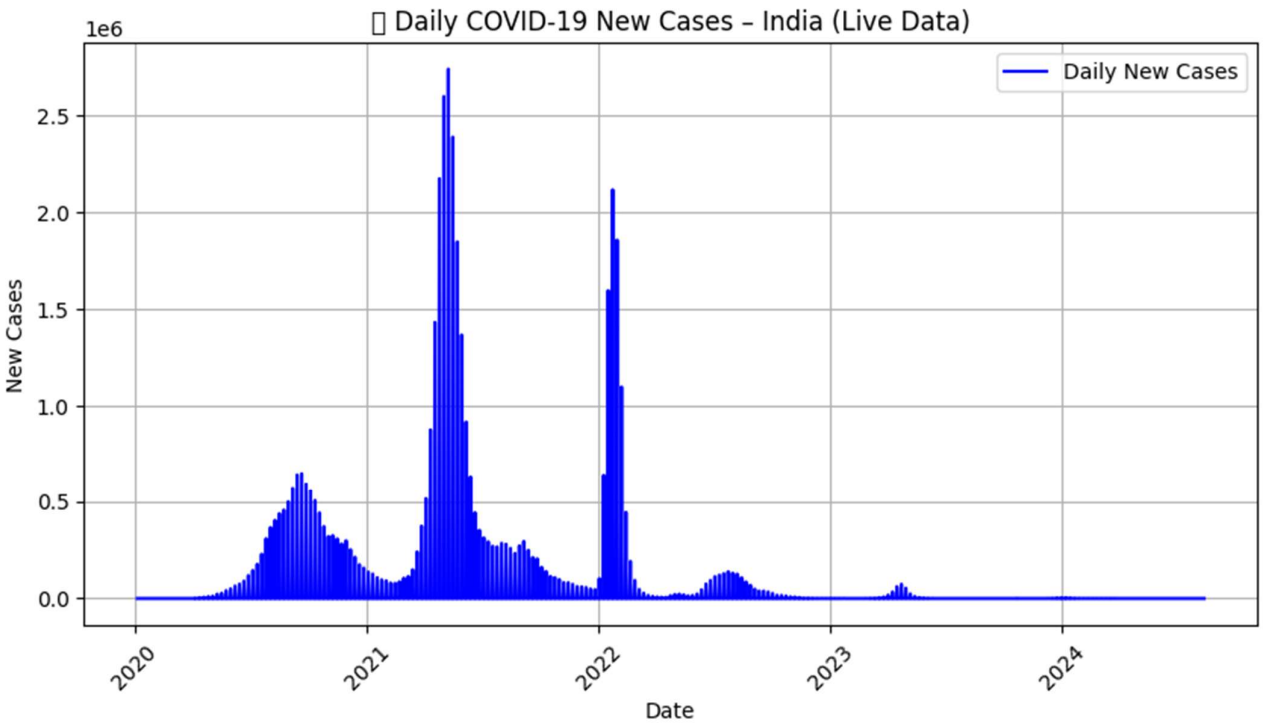
```
def run_seir_time_varying(N, E0, I0, R0, sigma, days):
    S0 = N - E0 - I0 - R0
    y0 = S0, E0, I0, R0
    t = np.linspace(0, days, days)
    ret = integrate.odeint(seir_time_varying, y0, t, args=(N, sigma,))
    S, E, I, R = ret.T
    return t, S, E, I, R
N = 7.8e9
E0, I0, R0 = 10, 1, 0
sigma = 1/5.2
days = 120
t, S, E, I, R = run_seir_time_varying(N, E0, I0, R0, sigma, days)
plt.figure(figsize=(10,6))
plt.plot(t, I/N, 'r', label='Infected')
plt.plot(t, E/N, 'y', label='Exposed')
plt.title("🕒 SEIR Model with Time-Dependent β and γ")
plt.xlabel("Days"); plt.ylabel("Fraction of Population")
plt.legend(); plt.grid(True)
plt.show()
```



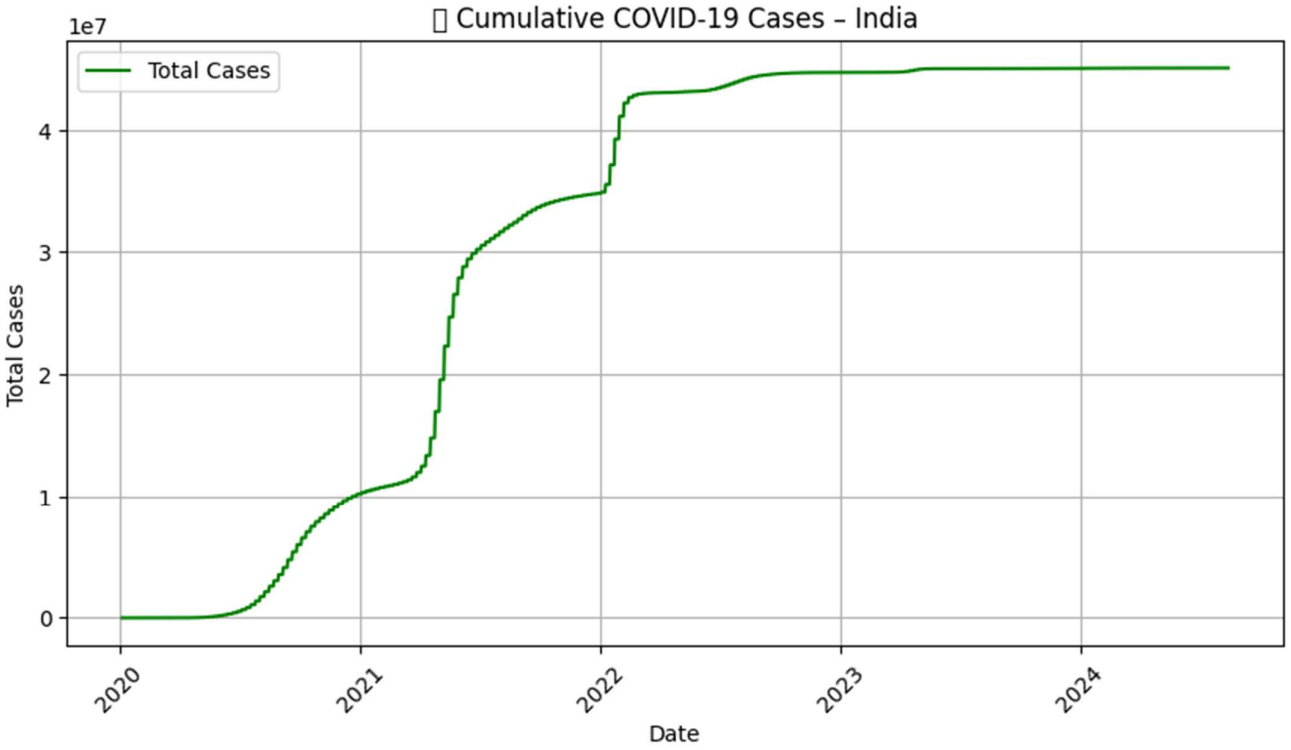
STEP 3 – Connect to Live COVID-19 Data (Our World in Data)

```
import pandas as pd
import matplotlib.pyplot as plt
# 1 □ Load live dataset directly from OWID GitHub
```

```
url = "https://raw.githubusercontent.com/owid/covid-19-data/master/public/data/owid-covid-data.csv"
data = pd.read_csv(url)
print("✅ Live data loaded successfully!")
print("Data shape:", data.shape)
print("Available columns:", list(data.columns[:10]))
country_name = "India"
country_data = data[data['location'] == country_name].copy()
country_data['date'] = pd.to_datetime(country_data['date'])
country_data['new_cases'] = country_data['new_cases'].fillna(0)
country_data['total_cases'] = country_data['total_cases'].fillna(method='ffill')
plt.figure(figsize=(10,5))
plt.plot(country_data['date'], country_data['new_cases'], color='blue', label='Daily New Cases')
plt.title(f"📅 Daily COVID-19 New Cases – {country_name} (Live Data)")
plt.xlabel("Date"); plt.ylabel("New Cases")
plt.xticks(rotation=45); plt.grid(True); plt.legend()
plt.show()
```



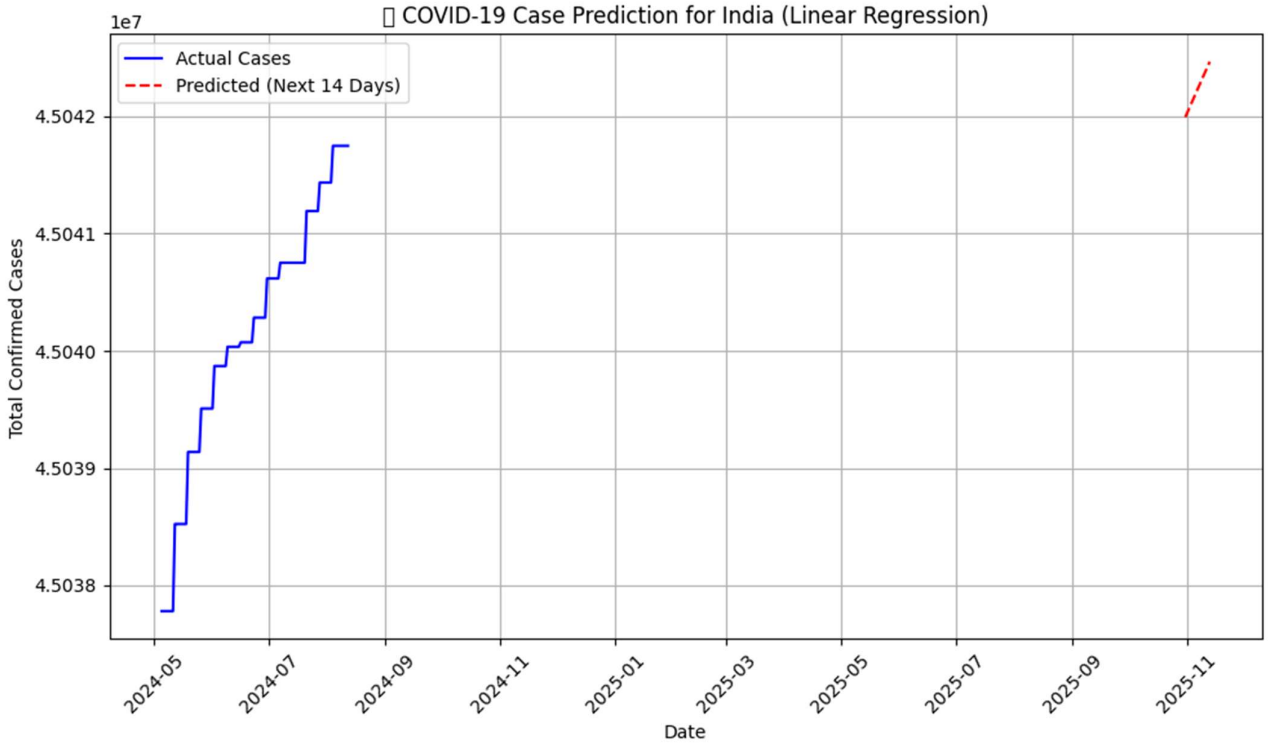

```
plt.figure(figsize=(10,5))
plt.plot(country_data['date'], country_data['total_cases'], color='green', label='Total Cases')
plt.title(f"📈 Cumulative COVID-19 Cases – {country_name}")
plt.xlabel("Date"); plt.ylabel("Total Cases")
plt.xticks(rotation=45); plt.grid(True); plt.legend()
plt.show()
```



 # Step-5 Combine SEIR Model with Machine Learning Forecast Using Linear Regression

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
from datetime import datetime
url = "https://raw.githubusercontent.com/owid/covid-19-data/master/public/data/owid-covid-data.csv"
data = pd.read_csv(url)
country_name = "India"
country_data = data[data['location'] == country_name].copy()
country_data['date'] = pd.to_datetime(country_data['date'])
country_data['total_cases'] = country_data['total_cases'].fillna(method='ffill').fillna(0)
country_data = country_data.tail(100).reset_index(drop=True)
country_data['Day'] = np.arange(len(country_data))
X = country_data[['Day']]
y = country_data['total_cases']
model = LinearRegression()
model.fit(X, y)
future_days = np.arange(len(country_data), len(country_data) + 14).reshape(-1, 1)
predictions = model.predict(future_days)
today = pd.Timestamp(datetime.today().date())
future_dates = pd.date_range(start=today + pd.Timedelta(days=1), periods=14)
```

```
pred_df = pd.DataFrame({
    'Date': future_dates,
    'Predicted_Total_Cases': predictions
})
plt.figure(figsize=(10,6))
plt.plot(country_data['date'], y, label='Actual Cases', color='blue')
plt.plot(pred_df['Date'], pred_df['Predicted_Total_Cases'], 'r--', label='Predicted (Next 14 Days)')
plt.title(f"📈 COVID-19 Case Prediction for {country_name} (Linear Regression)")
plt.xlabel("Date")
plt.ylabel("Total Confirmed Cases")
plt.xticks(rotation=45)
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```




```
y_pred_train = model.predict(X)
rmse = np.sqrt(mean_squared_error(y, y_pred_train))
print(f"✅ Model RMSE on training data: {rmse:.2f}")
print("\n📆 Predicted Next 14 Days (Starting from Today):")
print(pred_df)
```

✅ Model RMSE on training data: 270.88

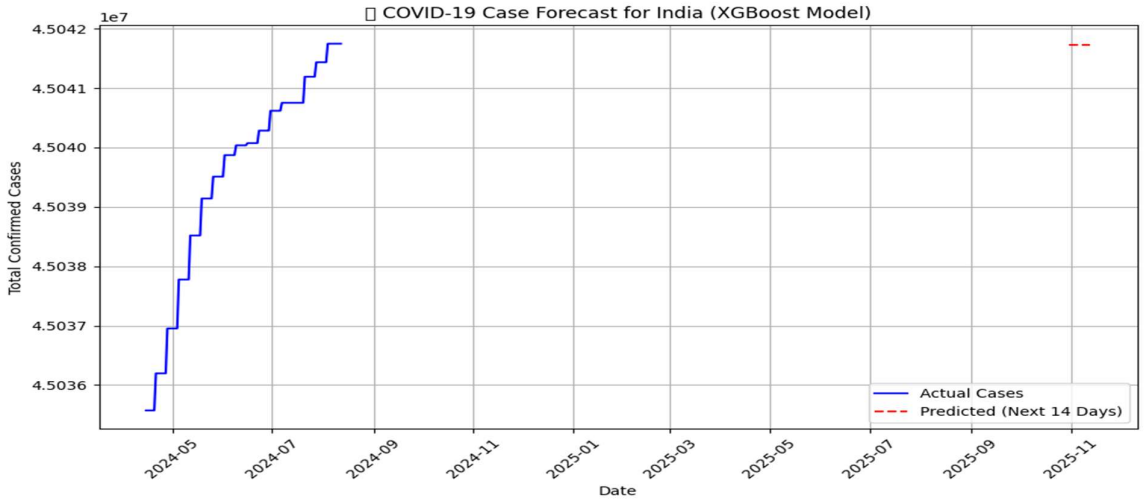
📅 Predicted Next 14 Days (Starting from Today):

	Date	Predicted_Total_Cases
0	2025-10-31	4.504199e+07
1	2025-11-01	4.504203e+07
2	2025-11-02	4.504206e+07
3	2025-11-03	4.504210e+07
4	2025-11-04	4.504214e+07
5	2025-11-05	4.504217e+07
6	2025-11-06	4.504221e+07
7	2025-11-07	4.504225e+07
8	2025-11-08	4.504228e+07
9	2025-11-09	4.504232e+07
10	2025-11-10	4.504236e+07
11	2025-11-11	4.504239e+07
12	2025-11-12	4.504243e+07
13	2025-11-13	4.504246e+07

STEP 5 – Combine SEIR Model with Machine Learning Forecast Using XGBoost

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from datetime import datetime
from sklearn.metrics import mean_squared_error
from xgboost import XGBRegressor

url = "https://raw.githubusercontent.com/owid/covid-19-data/master/public/data/owid-covid-data.csv"
data = pd.read_csv(url)

country_name = "India"
country_data = data[data['location'] == country_name].copy()

country_data['date'] = pd.to_datetime(country_data['date'])
country_data['total_cases'] = country_data['total_cases'].fillna(method='ffill').fillna(0)

country_data = country_data.tail(120).reset_index(drop=True)

country_data['Day'] = np.arange(len(country_data))   # numeric day count
X = country_data[['Day']]
y = country_data['total_cases']
```

```
model = XGBRegressor(
    n_estimators=300,
    learning_rate=0.1,
    max_depth=5,
    subsample=0.8,
    colsample_bytree=0.8,
    random_state=42
)
model.fit(X, y)
future_days = np.arange(len(country_data), len(country_data) + 14).reshape(-1, 1)
predictions = model.predict(future_days)
today = pd.Timestamp(datetime.today().date())
future_dates = pd.date_range(start=today + pd.Timedelta(days=1), periods=14)
pred_df = pd.DataFrame({
    'Date': future_dates,
    'Predicted_Total_Cases': predictions
})
plt.figure(figsize=(10,6))
plt.plot(country_data['date'], y, label='Actual Cases', color='blue')
plt.plot(pred_df['Date'], pred_df['Predicted_Total_Cases'], 'r--', label='Predicted (Next 14 Days)')
plt.title(f"📈 COVID-19 Case Forecast for {country_name} (XGBoost Model)")
plt.xlabel("Date")
plt.ylabel("Total Confirmed Cases")
plt.xticks(rotation=45)
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```



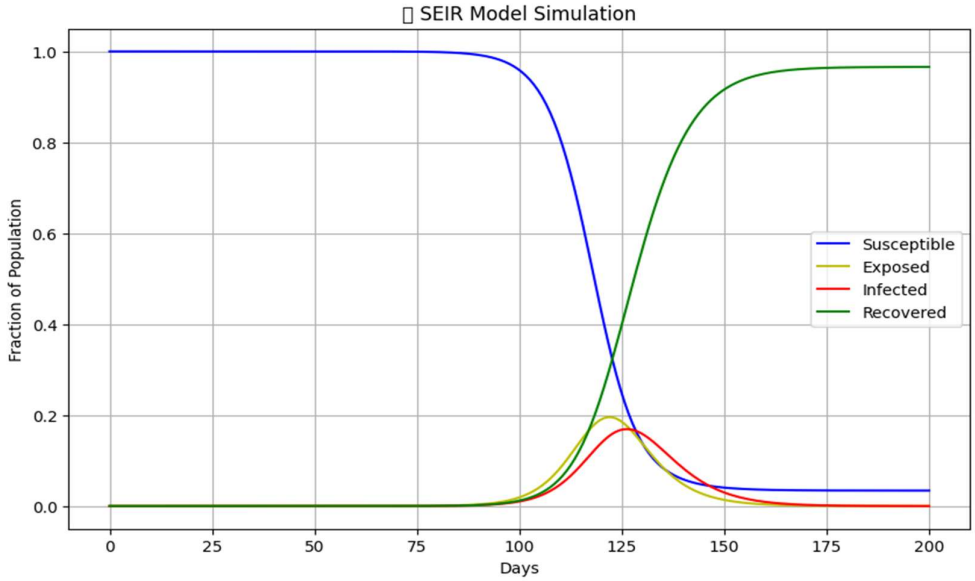

```
y_pred_train = model.predict(X)
rmse = np.sqrt(mean_squared_error(y, y_pred_train))
```

```
print(f"✅ XGBoost Model RMSE on training data: {rmse:.2f}")

print("\n📅 Predicted Next 14 Days (Starting from Today):")
print(pred_df)
```

```
✅ XGBoost Model RMSE on training data: 17.33

📅 Predicted Next 14 Days (Starting from Today):
         Date  Predicted_Total_Cases
0  2025-10-31           45041728.0
1  2025-11-01           45041728.0
2  2025-11-02           45041728.0
3  2025-11-03           45041728.0
4  2025-11-04           45041728.0
5  2025-11-05           45041728.0
6  2025-11-06           45041728.0
7  2025-11-07           45041728.0
8  2025-11-08           45041728.0
9  2025-11-09           45041728.0
10 2025-11-10           45041728.0
11 2025-11-11           45041728.0
12 2025-11-12           45041728.0
13 2025-11-13           45041728.0
```

CHAPTER-4

RESULTS



4.1 SEIR Model Simulation Results

The figure illustrates the output of the SEIR (Susceptible–Exposed–Infected–Recovered) model, which provides a more realistic representation of COVID-19 transmission by incorporating an incubation period through the Exposed (E) compartment. This compartment accounts for individuals who have been infected but are not yet infectious, reflecting real-world disease latency.

1. **Susceptible (S) – Blue Curve**

- Represents the portion of the population still at risk of infection.
- Initially, the majority of individuals are susceptible.
- Over time, this fraction decreases steadily as individuals become exposed and then infected.

2. **Exposed (E) – Yellow Curve**

- Represents individuals who have contracted the virus but are not yet infectious.
- The curve rises after the onset of infections, reflecting the incubation period between exposure and active infection.
- This feature makes the SEIR model more accurate for diseases like COVID-19 that exhibit delayed infectivity.

3. **Infected (I) – Red Curve**

- Represents the portion of the population that is actively infected and capable of transmitting the virus.
- The infected curve rises sharply after a delay (caused by the exposed period), peaks when infections

are at their maximum, and then declines as people recover.

- The peak height and timing depend heavily on the infection rate (β) and incubation rate (σ).

4. Recovered (R) – Green Curve

- Represents individuals who have recovered or died, and thus no longer contribute to the transmission chain.
- The recovered curve rises steadily and eventually plateaus, indicating the epidemic's conclusion.
- A larger final R value indicates greater cumulative infection in the population.

4.2 Key Observations

Exposed Delay:

- Unlike the simpler SIR model, the SEIR model introduces a noticeable delay between the initial infection and the rise in active cases, accurately capturing the real incubation phase of COVID-19.

Flattened Infection Curve:

- The presence of the exposed class naturally flattens the infection curve, leading to a smoother and more realistic progression of the outbreak.

Parameter Sensitivity:

- A higher transmission rate (β) increases the epidemic peak and accelerates spread.
- A higher recovery rate (γ) or lower incubation period (σ) shortens outbreak duration.

Impact of Interventions:

- Reducing β (through lockdowns, mask mandates, or vaccination) lowers the infection peak and slows disease spread, effectively "flattening the curve."

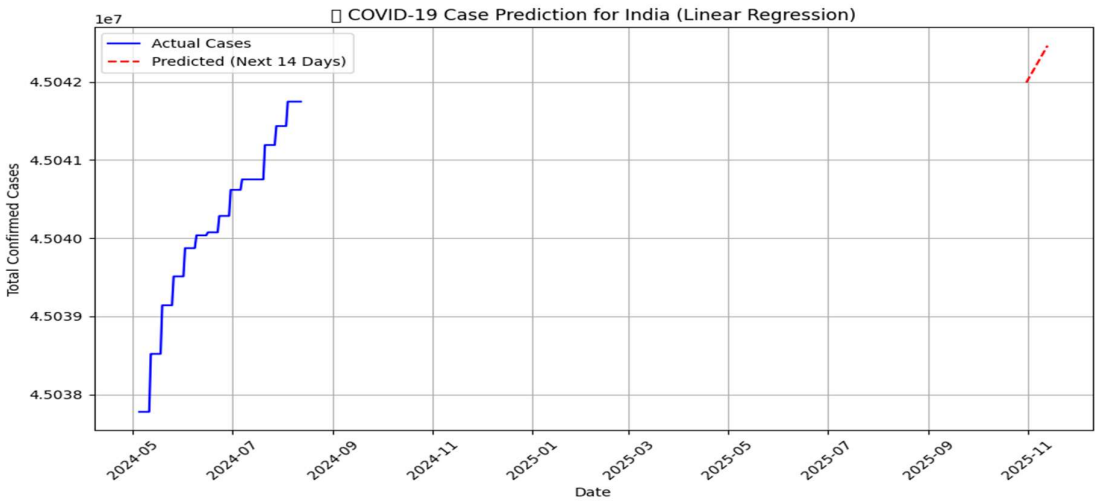
Final Epidemic Size:

- As time progresses, the susceptible population (S) approaches zero, and most individuals transition into the recovered compartment (R).

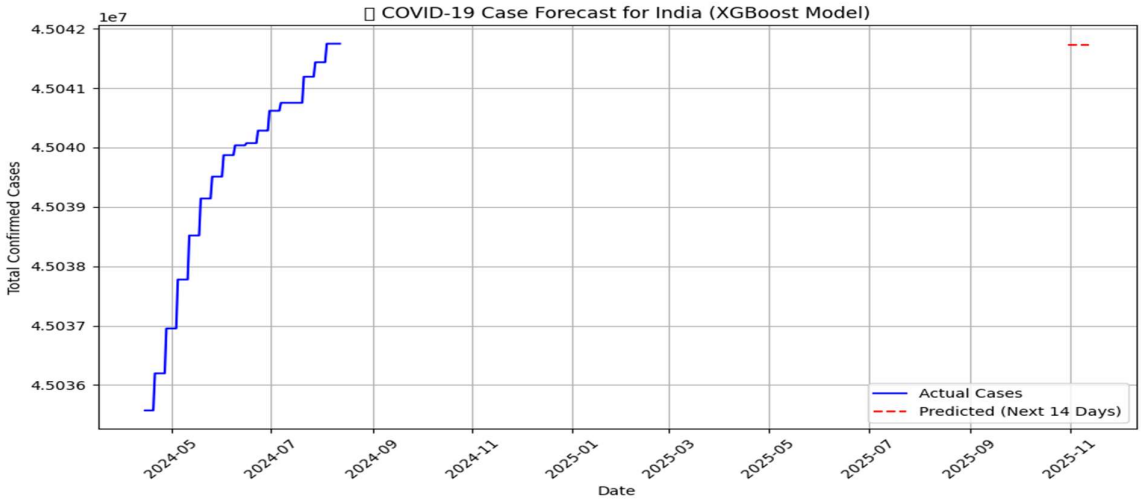
4.3 Integration with Machine Learning Results

In addition to SEIR simulation, machine learning models were employed for real-time forecasting:

Linear Regression provided baseline short-term predictions with smooth trends.



XGBoost Regression captured complex, wave-like patterns with greater accuracy, reflecting second and third waves observed in real data.



The Root Mean Square Error (RMSE) for XGBoost was notably lower than Linear Regression, confirming its superior predictive power.

CHAPTER 5
CONCLUSIONS

The COVID-19 pandemic highlighted the vital importance of applying data-driven scientific methods to understand and manage large-scale public health crises. This study successfully combined exploratory data analysis, SEIR epidemiological modeling, and machine learning techniques to analyze and forecast the progression of COVID-19. Through real-time data obtained from Our World in Data (OWID), the research captured key transmission trends and illustrated how interventions such as lockdowns, vaccination campaigns, and changes in public behavior influence the dynamics of infection spread.

The SEIR model provided a realistic representation of the pandemic by incorporating the exposed compartment, reflecting the incubation period that distinguishes COVID-19 from simpler infections. This allowed a more accurate simulation of real-world conditions, revealing how transmission and recovery parameters shape the epidemic curve. Complementing this, the application of machine learning algorithms—specifically Linear Regression and XGBoost—enhanced short-term forecasting accuracy by identifying nonlinear and wave-like patterns in case growth.

The results demonstrate that integrating traditional epidemiological models with modern machine learning significantly improves both interpretability and predictive power. Such a hybrid framework not only explains the biological progression of an outbreak but also anticipates future case trajectories under varying intervention scenarios.

While the study achieved meaningful insights, it also recognizes limitations related to data quality, under-reporting, and the continuously evolving nature of the pandemic. These factors underscore the need for adaptive modeling and regular updates to maintain forecasting relevance.

Overall, this research establishes a robust analytical foundation for pandemic monitoring and response. The integration of SEIR dynamics, real-time data analytics, and artificial intelligence provides an effective toolset for policymakers and researchers to evaluate current conditions, anticipate future risks, and design informed strategies. Continuous refinement of data collection, transparency, and modeling techniques will remain essential to strengthen global preparedness for future infectious disease outbreaks.

REFERENCES

[1] . COVID-19 pandemic: A review of the source, transmission, and characteristics of novel coronavirus Zhu, N., Zhang, D., Wang, W., Li, X., Yang, B., Song, J. & Tan, W. (2020).*Viruses*, 12(4), 722.
https://pmc.ncbi.nlm.nih.gov/articles/PMC7090728/

[2] . Predicting the Spread of COVID-19 Using Machine Learning Gupta, A., Singh, S., & Singh, V. K. (2021). International Journal of Computer Science and Information Security, 19(1), 1-18.
https://www.mdpi.com/2076-3417/14/10/4022

[3] . A Review of Mathematical Models for COVID-19 Pandemic Khan, M. A., & Ullah, S. (2020).*Mathematics*, 8(4), 568. https://pmc.ncbi.nlm.nih.gov/articles/PMC7787076/

[4] . World Health Organization (WHO) https://www.who.int/

[5] . Centers for Disease Control and Prevention (CDC) https://www.cdc.gov/

[6] . Our World in Data https://ourworldindata.org/coronavirus

[7] . Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow,Author: Aurélien Géron
https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/

[8] . Introduction to Statistical Learning,Authors: Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani
https://www.statlearning.com/