Assessment Report

on

COVID-19 Case Prediction

Submitted by:

Rajat Sharma(202401100400151)

Ranjan Kumar(202401100400153)

Shashi Ranjan(202401100400171)

Shivam Kumar Singh(202401100400176)

Sunny Kumar(202401100400195)

Year & Section:

1St year / CSE-AIML / C

Submitted to:

- **Abhishek sir**

Title: Time-Series Forecasting of COVID-19 Cases in India Using Regression Techniques

Introduction:

The COVID-19 pandemic has had a profound impact on global public health and economies. Accurate forecasting of COVID-19 cases is critical for governments, healthcare systems, and the general public to plan and respond effectively. This project focuses on building a time-series prediction model to forecast the number of future COVID-19 cases using historical data.

Using publicly available datasets from Johns Hopkins University, we analyze the growth trend of cumulative confirmed COVID-19 cases in India. By applying regression techniques, particularly linear regression, we aim to model the trajectory of the outbreak and predict future case numbers. This analysis helps in understanding patterns and can support timely decision-making.

Methodology:

1. **Data Collection:**
   - The dataset used in this project is obtained from Kaggle, which compiles COVID-19 case data from Johns Hopkins University.
   - The CONVENIENT_global_confirmed_cases.csv file provides cumulative confirmed cases by country.

1. **Data Preprocessing:**
   - Extract the dataset from a ZIP file.
   - Select India's data and clean the dataset (e.g., remove redundant headers, convert date formats).
   - Convert the date into a numerical format (Day) representing days since the start of the dataset.

1. **Feature Engineering:**
   - Create a new feature representing the number of days from the first recorded case.
   - Split the data into training and testing sets (last 30 days used for testing).

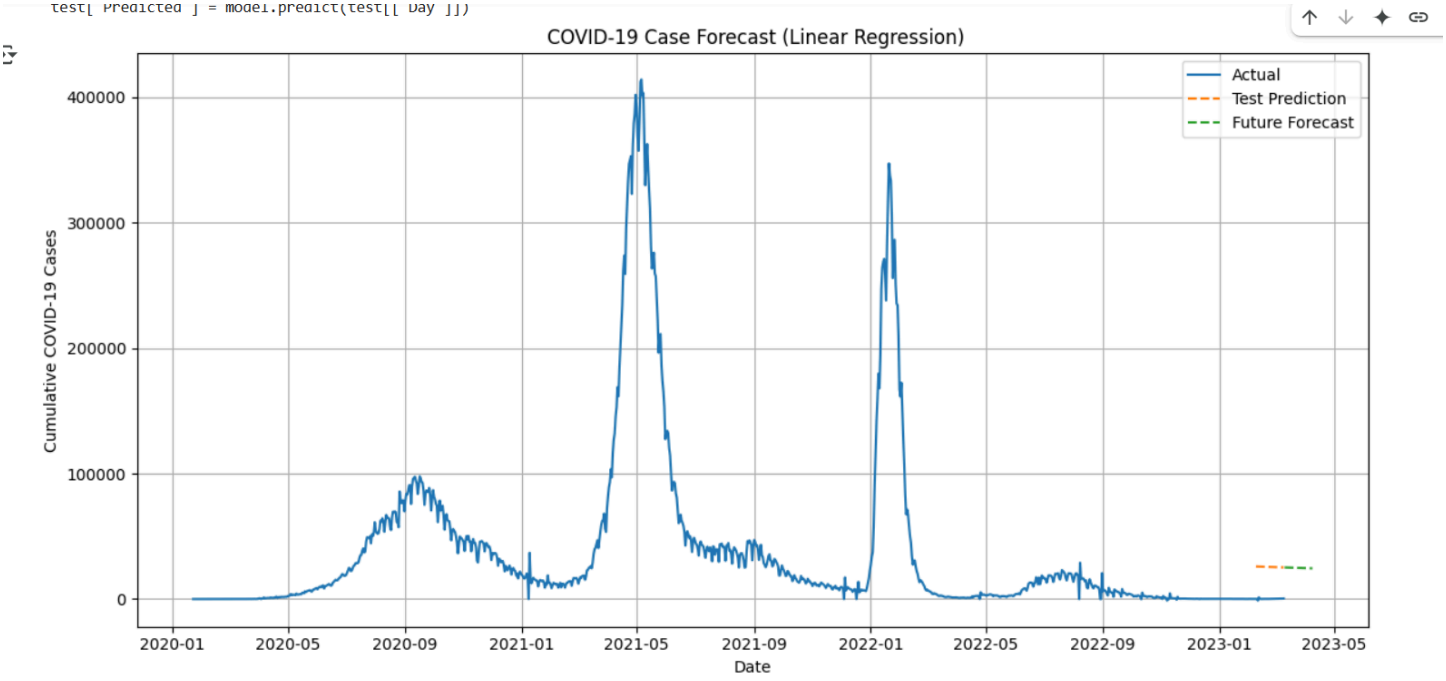
1. **Model Implementation:**
   - Use **Linear Regression** to model the relationship between Day and cumulative cases.
   - Train the model on the historical data and predict the number of cases for the next 30 days.

1. **Forecasting:**
   - Generate predictions for the testing set and an additional 30-day future period.
   - Visualize actual cases, test predictions, and forecasted values using line plots.

1. **Evaluation (Optional):**
   - Evaluate model performance using error metrics such as Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) on the test data.

1. **Visualization:**
   - Plot the actual data, test predictions, and future forecast to visualize the trend.
   - Use matplotlib for plotting time-series data clearly and interpretab

Output:



Code:

```
import zipfile

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression
```

```
import os

# STEP 1: Extract ZIP file
zip_file_path = '/content/archive (2).zip'
extract_to = 'covid_data'
with zipfile.ZipFile(zip_file_path, 'r') as zip_ref:
    zip_ref.extractall(extract_to)

# STEP 2: Load India data from the convenient dataset
file_path = os.path.join(extract_to, 'CONVENIENT_global_confirmed_cases.csv')
df = pd.read_csv(file_path)

# STEP 3: Clean data
df = df.drop(index=0)  # remove header row inside data
df = df.rename(columns={"Country/Region": "Date"})
df['Date'] = pd.to_datetime(df['Date'])  # convert to datetime
india_df = df[['Date', 'India']].copy()
india_df['India'] = india_df['India'].astype(float)

# STEP 4: Feature engineering
india_df['Day'] = (india_df['Date'] - india_df['Date'].min()).dt.days

# STEP 5: Train/test split
train = india_df[:-30]
test = india_df[-30:]

# STEP 6: Linear Regression model
model = LinearRegression()
model.fit(train[['Day']], train['India'])
```

```
test['Predicted'] = model.predict(test[['Day']])

# STEP 7: Forecast next 30 days
future_days = 30
last_day = india_df['Day'].max()
future = pd.DataFrame({'Day': range(last_day + 1, last_day + 1 + future_days)})
future['Predicted'] = model.predict(future[['Day']])
future['Date'] = india_df['Date'].max() + pd.to_timedelta(future['Day'] - last_day, unit='D')

# STEP 8: Plot the results
plt.figure(figsize=(12, 6))
plt.plot(india_df['Date'], india_df['India'], label='Actual')
plt.plot(test['Date'], test['Predicted'], label='Test Prediction', linestyle='--')
plt.plot(future['Date'], future['Predicted'], label='Future Forecast', linestyle='--')
plt.xlabel('Date')
plt.ylabel('Cumulative COVID-19 Cases (India)')
plt.title('India COVID-19 Case Forecast (Linear Regression)')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```