

TIME SERIES FORECASTING OF COVID- 19

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Engineering degree in Computer Science and Engineering
by

PATURU ANUHYA (37110553)

PENTELA NEELIMA (37110563)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAA

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119**

APRIL - 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Paturu Anuhya (37110553)** and **Pentela Neelima (37110563)** who carried out the project entitled as **Time Series Forecasting Of COVID- 19** under my supervision from November 2020 to April 2021.

Internal Guide

Dr. G. MATHIVANAN, M.E., (Ph.D.,)

Head of the Department

Dr. S. VIGNESHWARI, M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We **Paturu Anuhya (37110553)** and **Pentela Neelima (37110563)** hereby declare that the Project Report entitled “**Time Series Forecasting Of COVID- 19**” done by me under the guidance of **Dr. G. Mathivanan, M.E., (Ph.D.,)** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge our sincere thanks to **Board of Management of Sathyabama Institute Of Science And Technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean, School of Computing, Dr. S. Vigneshwari, M.E., Ph.D., and Dr. L. Lakshmanan, M.E., Ph.D., Heads** of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. G. Mathivanan, M.E., (Ph.D.,)** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Covid-19 is a highly contagious virus which almost freezes the world along with its economy. Its ability of human-to-human and surface-to-human transmission turns the world into catastrophic phase. Coronaviruses are enveloped RNA viruses from the Coronaviridae family affecting neurological, gastrointestinal, hepatic and respiratory systems. In late 2019 a new member of this family belonging to the Betacoronavirus genera (referred to as COVID-19) originated and spread quickly across the world calling for strict containment plans and policies. In most countries in the world, the outbreak of the disease has been serious and the number of confirmed COVID-19 cases has increased daily, while, fortunately the recovered COVID-19 cases have also increased. Clearly, forecasting the "confirmed" and "recovered" COVID-19 cases helps planning to control the disease and plan for utilization of health care resources. Time series models based on statistical methodology are useful to model time-indexed data and for forecasting. Time-series analysis was performed based on the total confirmed cases and deaths of COVID-19 in the world, Brazil, Peru, Canada and Chile between 27 February 2020 and 30 June 2020.

TABLE OF CONTENT

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	V
	LIST OF FIGURES	IX
	LIST OF ABBREVIATION	X
1	INTRODUCTION	1
	1.1 INTRODUCTION	1
	1.2 AIM AND OBJECTIVE	2
	1.3 SCOPE	2
	1.4 ADVANTAGES	2
	1.5 TASKS TO BE PERFORMED	2
2	LITERATURE SURVEY	3
	2.1 COVID- 19 PANDEMIC PREDICTION	3
	2.2 TREND ANALYSIS AND FORECATING OF COVID-19	3
	2.3 ANALYSIS OF SPATIAL SPREAD RELATIONSHIPS	4
	2.4 SEIR AND REGRESSION MODEL	4
3	METHODOLOGY	5
	3.1 SYSTEM ARCHITECTURE	6
	3.2 SYSTEM REQUIREMENTS	6
	3.3 MODULE DESCRIPTION	6
	3.3.1 ANALYZING PRESENT CONDISTION	6
	3.3.2 TREND SIMLAR TO ITALY/WUHAN	6
	3.3.3 EXPLORING WORLD WIDE DATA	7

	3.3.4 FORECASTING TOTAL NO OF CASES	7
	3.4 DATA COLLECTION	7
	3.5 CLEANING DATA	7
	3.6 ANALYSING DATA	8
	3.7 MODEL SELECTION	8
	3.7.1 ADVANTAGES OF PROPHET	10
	3.8 DATA VISUALIZATION	10
4	RESULT AND DISCUSSION	12
5	CONCLUSION AND FRAMEWORK	13
	REFERENCE	14
	APPENDIX	15
	SOURCE CODE	15
	SCREENSHOTS	22
	PLAGIARISM REPORT	25

CERTIFICATION



LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
3.1	SYSTEM ARCHITECTURE	5
3.2	PREDICTION PROCEDURE METHOD	9
3.3	VISUALIZATION OF COVID CASES IN INDIA	11
3.4	VISUALIZATION IN DIFFERENT COUNTRIES	11
3.5	VISUALISING THE SPREAD GEOGRAPHICALLY	11
4.1	RESULT OF CONFIRMED, DEATHS AND RECOVERED CASES	12

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Coronavirus disease (COVID-19) is deemed one of the most significant provocative challenges facing humanity in the twenty-first century in terms of disease progression, fast spread, infection ways, and the increasing mortality rate over the world. coronavirus disease (COVID-19) is regarded as one of the most imminent disease outbreaks which threaten public health on various levels worldwide. Because of the unpredictable outbreak nature and the virus's pandemic intensity, people are experiencing depression, anxiety, and other strain reactions. The response to prevent and control the new coronavirus pneumonia has reached a crucial point. Therefore, it is essential—for safety and prevention purposes—to promptly predict and forecast the virus outbreak in the course of this troublesome time to have control over its mortality.

Coronaviruses are enveloped RNA viruses from the Coronaviridae family affecting neurological, gastrointestinal, hepatic and respiratory systems. In late 2019 a new member of this family belonging to the Betacoronavirus genera (referred to as COVID-19) originated and spread quickly across the world calling for strict containment plans and policies. In most countries in the world, the outbreak of the disease has been serious and the number of confirmed COVID-19 cases has increased daily, while, fortunately the recovered COVID-19 cases have also increased. Clearly, forecasting the "confirmed" and "recovered" COVID-19 cases helps planning to control the disease and plan for utilization of health care resources. Time series models based on statistical methodology are useful to model time-indexed data and for forecasting.

COVID-19 has now had a huge impact in the world, and more than 8 million people in more than 100 countries are infected. To contain its spread, a number of countries published control measures. However, it's not known when the epidemic will end in global and various countries. Predicting the trend of COVID-19 is an extremely important challenge. We integrate the most updated COVID-19 epidemiological data before June 16, 2020 into the Logistic model to fit the

cap of epidemic trend, and then feed the cap value into FbProphet model, a machine learning based time series prediction model to derive the epidemic curve and predict the trend of the epidemic.

1.2 AIM AND OBJECTIVE

The main aim of this project is that we need a strong modal that predicts how the virus could spread across different countries and regions. The goal of this task is to build a model that predicts the spread of the virus. This project provides features like

- To provide an evaluative study of prediction model using COVID- 19 cases, and forecasting the impact of virus.
- To develop a Time Series Forecasting model using FBProphet

1.3 SCOPE

This model helps us to analyze the transmission of COVID- 19 Virus across the globe and to get the projection of Corona virus cases in world wide. It helps in analysing the present condition in India.

1.4 ADAVANTAGES

- Open Source algorithm
- Accurate
- Faster Data fitting

1.5 TASKS TO BE PERFORMED

- Analyse the condition in India
- Compare the trend with Italy/S. Korea/Wuhan for similarity
- Explore the World Wide data
- Forecast the World Wide COVID- 19 cases using Prohet

CHAPTER 2

LITERATURE SURVEY

2.1 SURVEY 1

2.1.1 Publication: COVID-19 Pandemic Prediction using Time Series Forecasting Models

2.1.2 Alabi , Paikrshit

- Albai adopted the Facebook Prophet model to forecast spread of COVID- 19.They have performed prediction for confirmed and death cases .Their forecasting accuracy of Prophet was 79.6% for the data from WHO between 7th April to 3rd May 2020.
- Parikshit have presented medical perspective of COVID- 19, and prediction using Prophet model. They have recommended the Prophet for prediction due to open source algorithm, accuracy, and faster data fitting.
- Using the Prophet model, they have predicted infected cases worldwide as 1.6 million by the end of May 2020, and 2.3 million by the end of June 2020.

2.2 SURVEY 2

2.2.1 Publication: Trend analysis and forecasting of covid19 break in India

2.2.2 Author: Gupta, Pal and Kumar

- used exploratory data analysis to report the situation in the time period of January to March in India. They use time series forecasting methods to predict the future trends.
- A very famous machine learning model- Arima model prediction was used and inferred that a huge surge in the number of likely covid-19 positive cases was predicted in April and May.
- The average that was forecasted was a detection of approximately 7000 patients in a total span of 30 days in April. However in reality the figures were higher.

2.3 SURVEY 3

2.3.1 Publication: Analysis of Spatial Spread Relationships of Coronavirus

2.3.2 Author: Julio , Monica , Sanchez, Castillo

- Uses clustering methods to analyze countries on the basis of most affected patients and how they are reacting to it.
- This paper shows the world scenario first and thereby drilling down to the country of Mexico. Speaking of a country like India, where 65.97% (Trading Economics) belong to rural population, accessibility and utilization of primary health care during the current pandemic situation is a big question

2.4 SURVEY 4

2.4.1 Publication: SEIR and Regression Model based COVID-19 outbreak predictions in India

2.4.2 Author: Pandey and Chowdhury

- In collaboration with Defence Research and Development Organization (DRDO) India also covered data from January 30 to March 30, 2020. They used regression models for forecasting. According to them, expected cases may rise to about 5000 in a two week time period.
- This was far more accurate than the model predicted by Pal and Gupta however actual scenario showed a bigger upsurge.
- The findings of the paper may be relevant to several other sectors or other branches of healthcare as immunity power is strongly related to fighting with Covid-19. Healthcare experts also maintain that people with a less developed immunity system are more likely to be a victim of Covid-19.

CHAPTER 3

METHODOLOGY

3.1 SYSTEM ARCHITECTURE

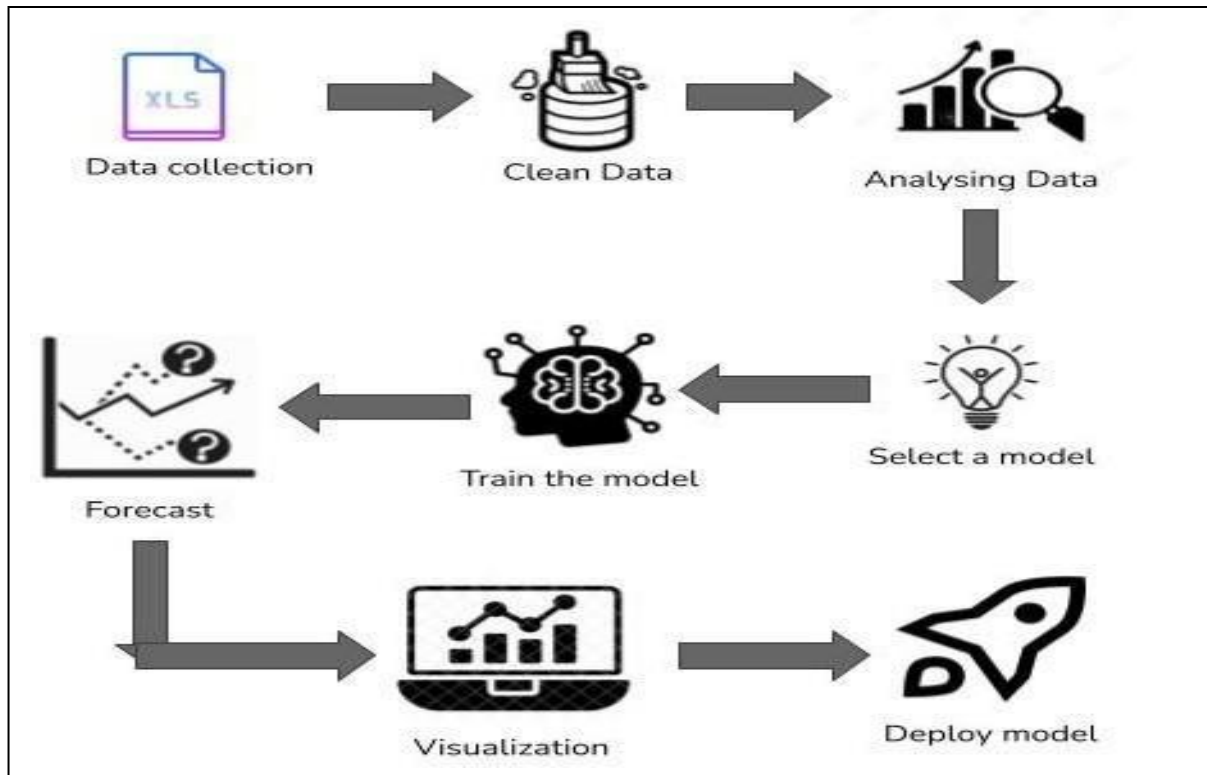


Fig:3.1 System Architecture

3.2 System Requirements

- Operating System : Windows 7 , 8, 10 (64 bit)
- Tools : Google Colabs (or) Jupyter Notebook
- Language : Python
- Libraries Needed : Pandas, matplotlib, seaborn, folium, Plotly,

FBProphet

- Hard Disk : 500GB and Above
- RAM : 4GB and Above
- Processor : I3 and Above

3.3 MODULE DESCRIPTION

- Analyzing Present condition in India
- Is the trend similar to Italy, South Korea and Wuhan?
- Exploring World Wide data
- Forecasting total number of cases

3.3.1 Analyzing Present condition in India

- In this module, we read the datasets that we have collected.
- Then we calculate the total no. of confirmed cases across India.
- Then analyze the no. of Active Covid-19 cases in affected State/Union Territories by highlighting the dataframe
- We visualize the spread geographically using folium to create a zoomable map.
- We also visualize the Confirmed vs. Recovered Cases using Seaborn
- Finally, we visualize how the coronavirus cases are rising using plotly.

3.3.2 Is the trend similar to Italy, South Korea and Wuhan?

- Visualize the cumulative cases in India, Italy, South Korea and Wuhan.
- Comparison between the rise of cases in the countries.
- Visualize the trend after crossing 100 cases

3.3.3 Exploring World Wide data

- We read the world wide data of Covid-19 and visualize the data by scatter plot differentiating between confirmed, recovered and deaths.

3.3.4 Forecasting total number of cases

- Generating a forecast graph of confirmed cases, deaths and recovered cases of NCOVID-19 using Prophet by creating a base model with no tweaking of seasonality-related parameters and additional regressors.

3.4 DATA COLLECTION

The most recent daily COVID-19 outbreak epidemical data in global are collected form John Hopkins University, which released a dashboard at the country level. Data are available at an ArcGIS platform . Therefore, we collected the time series at County level for the periodical analysis from January 22, 2020 to June 16, 2020.

3.5 CLEANING DATA

Data cleaning or cleansing is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data. Dealing with missing data/value is one of the most tricky but common parts of data cleaning. While many models can live with other problems of the data, most models don't accept missing data.

Data cleaning impacts efficiency of rest of your data modeling and decision making process. Hence professional data scientists treat this step as critical as the algorithm building step. Efficiency gains of data cleaning results from:

- 1) Lesser processing time
- 2) More accurate predictions

- 3) Simpler algorithms needed
- 4) Better learning ability of the model

3.6 ANALYSING DATA

Data Analysis is the process of systematically applying statistical and/or logical techniques to describe and illustrate, condense and recap, and evaluate data. According to Shamoo and Resnik (2003) various analytic procedures “provide a way of drawing inductive inferences from data and distinguishing the signal (the phenomenon of interest) from the noise (statistical fluctuations) present in the data”..

While data analysis in qualitative research can include statistical procedures, many times analysis becomes an ongoing iterative process where data is continuously collected and analyzed almost simultaneously. Indeed, researchers generally analyze for patterns in observations through the entire data collection phase (Savenye, Robinson, 2004). The form of the analysis is determined by the specific qualitative approach taken (field study, ethnography content analysis, oral history, biography, **unobtrusive** research) and the form of the data (field notes, documents, audiotape, videotape).

An essential component of ensuring data integrity is the accurate and appropriate analysis of research findings. Improper statistical analyses distort scientific findings, mislead casual readers (Shepard, 2002), and may negatively influence the public perception of research. Integrity issues are just as relevant to analysis of non-statistical data as well.

3.7 MODEL SELECTION

The framework Facebook team know as FbProphet is that the approach adopted for forecasting. Daily collect historical statistic records .Since the dataset is formed of daily records, they have to be converted so far rather than a string format. supported the dataset, the model are going to be fitted by the framework of FBprophet. consistent with the forecasting model fitted an appropriate algorithm generated to predict for the defined period. counting on

the number and quality of the dataset injected the results of Confirmed cases forecasted. The above mentioned procedure method is the followed to predict the outcome.

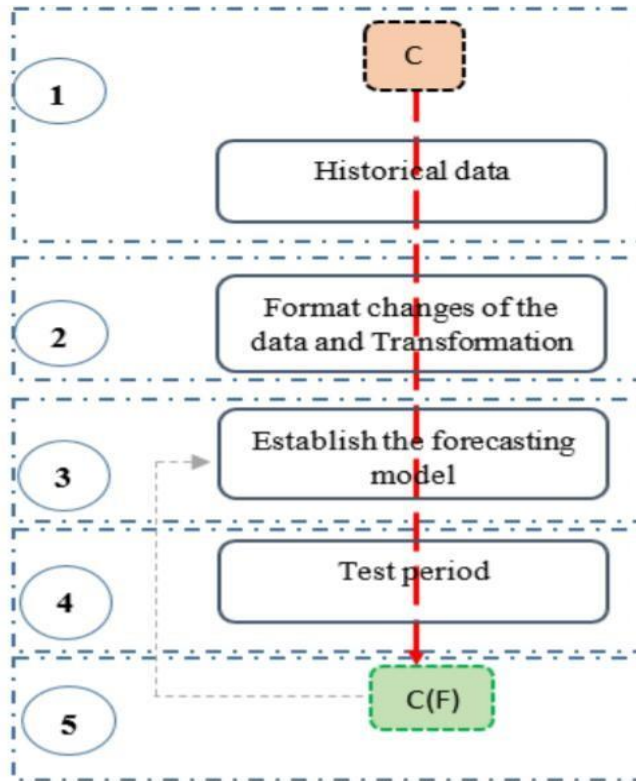


Fig:3.2 Prediction Procedure Method

Prophet is an open source time series forecasting algorithm designed by Facebook for ease of use without any expert knowledge in statistics or time series forecasting. Prophet is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well.

Machine learning techniques for forecasting algorithms are a branch of computer science that is trained from historical data, such as artificial neural networks, deep learning, decision trees and Bayesian networks. The idea of the algorithm is to select a suitable training model according to the characteristics of historical data and use it to predict the future observation results. We applied this technique to COVID-19 forecasting in the real world. Prophet is an open source framework of Facebook for time series forecasting based on additive model which

is opened up to the public in 2017 . The non-linear trends of Prophet are fitted with yearly, weekly, and daily seasonality, plus holiday effects. The perfect Prophet function can not only predict the future, but also fill in missing values and detect anomalies.

Prophet, the prediction model consists of superpositions $y(t)=g(t)+s(t)+h(t)+\epsilon_t$

Where,

- $g(t)$ is a trend function used to analyze the non-periodic changes of time series.
- $s(t)$ a periodic term, reflecting the periodic change, such as the periodicity of a week or year.
- $h(t)$ is the influence of an occasional day or days, such as a holiday.
- ϵ_t is an error term, on behalf of the failed to consider the effect of the error of the model.

3.7.1 Advantages of Prophet

- It is open source, accurate and fast.
- Allow for an outsized number of individuals to form forecasts, without training in statistic methods.
- It has tunable parameters.
- It is out there for both the Python and R programming languages.
- For missing data prophet is strong and shifts within the trend, and handles outliers typically well.

3.8 DATA VISUALIZATION

Data visualization is a technique that uses an array of static and interactive visuals within a specific context to help people understand and make sense of large amounts of data. The data is often displayed in a story format that visualizes patterns, trends and correlations that may otherwise go unnoticed.

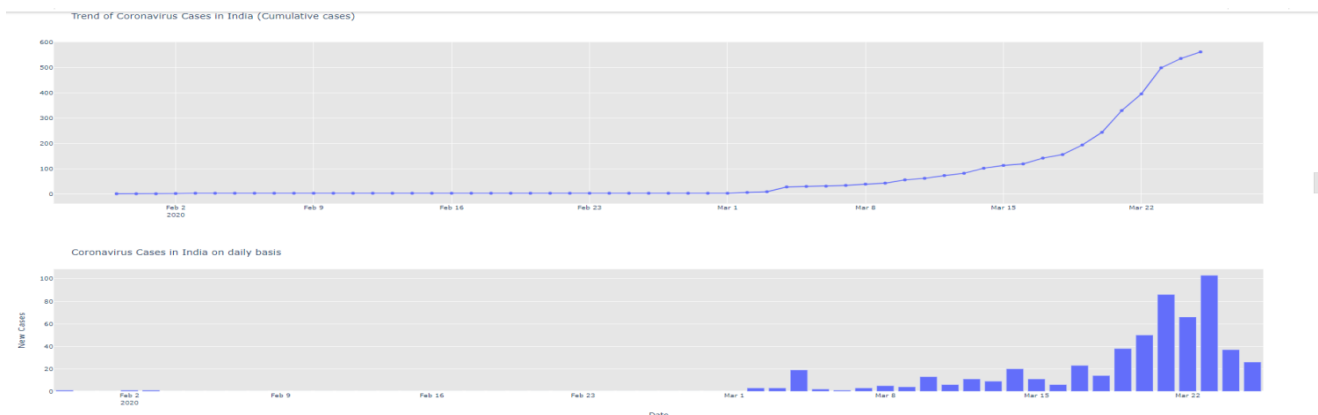


Fig: 3.3 Visualization of Covid Cases in India

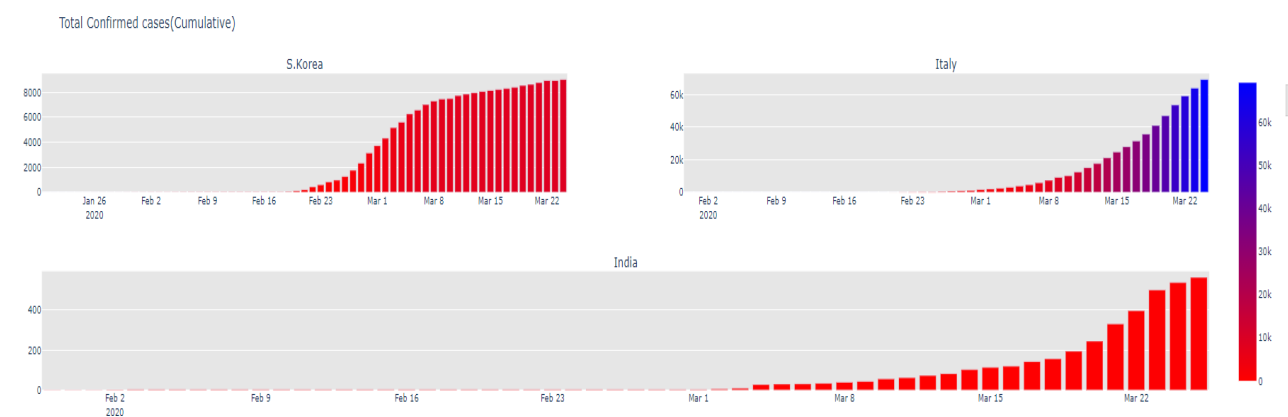


Fig: 3.4 Visualization in different countries

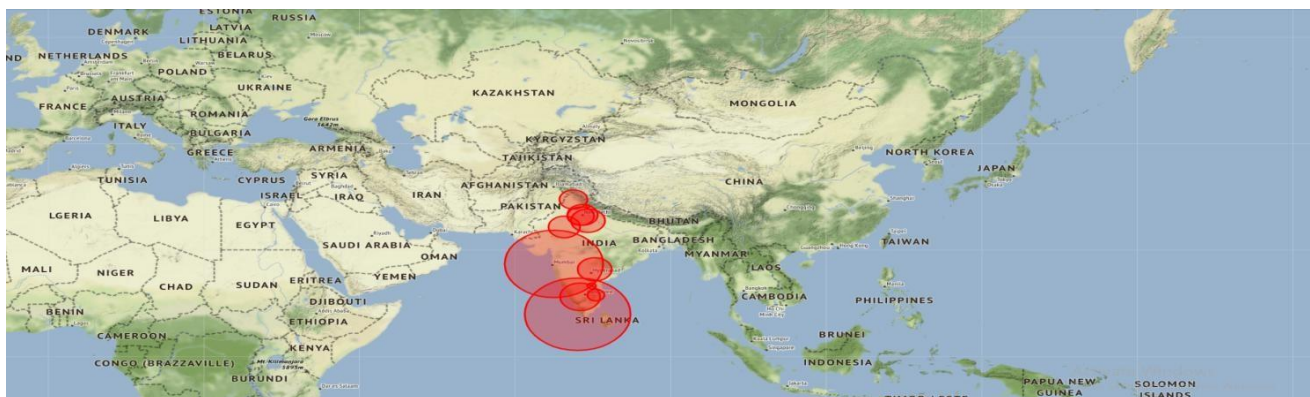


Fig: 3.5 Visualising the spread geographically

CHAPTER 4

RESULTS AND DISCUSSION

The predicted fit model based on the transformed number of confirmed cases by the $\log_{10}(C)$, has generated a quite well plot prediction for the coming days. The below chart and graph clearly represents that the rate of increase of cases are getting worse day by day. Moreover, the framework of FBprophet will create a new dataset assigned to the forecast variable that contains the forecasted values for future dates under the column $yhat$ which represent forecasting based on linear regression, as well as an interval in both ways', a sigmoid ($yhat_lower$) curve as the second and desirable scenario if the citizens respect the instructions of the Indian government orders, otherwise, an exponential ($yhat_upper$) curve will be followed as the third no desirable scenario. This means that many confirmed cases can vary from lowest to highest. The forecasting model and component visualizations show the underlying trend in the predicted trend as a linear trend, while also accurately modeling weekly seasonality (e.g. upper confirmed cases on Saturday and Sunday). This trend analysis can be used to effectively shows where to led resources up to weeks and months in the future on the pandemic. The same process can be done for forecasting worldwide deaths and forecasting worldwide recovered cases using Prophet. Generating a specific period ahead forecast of deaths and recovered cases of NCOVID-19 using Prophet, with 95% prediction interval by creating a base model with no tweaking of seasonality-related parameters and additional regressors.

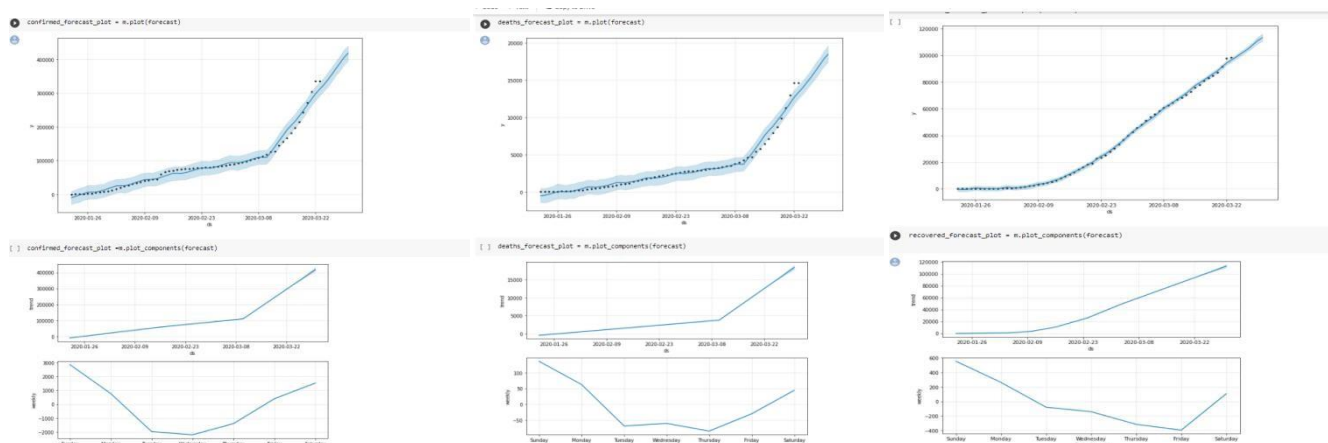


Fig: 4.1 Result of Confirmed, Deaths and Recovered cases

CHAPTER 5

CONCLUSION AND FUTURE WORK

This study provides a trend mechanism as seen by FBprophet of pandemics of coronavirus (COVID-19) spread, which can provide valuable insight for the national competent authorities; In future research, we pair this research with Deep Learning models and compare it with the FBprophet result's to refine the prediction models to further develop the prediction of the spread of this disease. WHO has declared COVID-19 as pandemic because it has infected most of the countries, and it is a major threat to human race. In this paper, we have analyzed and predicted the study of the disease using widely accepted forecasting model; FBProphet. We have collected COVID-19 data of highly affected countries Italy, South Korea, Wuhan India worldwide. For the most of the countries data, ARIMA has better performed compared to Prophet on scale of MAE, RMSE, RRSE, and MAPE error matrices. The trend analysis shows rapid growth in the infected cases, and prediction study shows great rise in the expected confirmed, recovered, and death cases worldwide. However, lockdowns and containment policies may affect the prediction results. The adopted models have performed well but it limits our study to the effectiveness of the models, which can be further improved using ensemble of multiple prediction models. The obtained forecasting results further can be improved by taking various variables into account like population density, weather, health system, patient history etc. using deep learning techniques, and artificial intelligence.

REFERENCE

- [1] Parikshit N Mahalle, Nilesh P Sable, Namita P Mahalle, and Gitanjali R Shinde. Data analytics: Covid-19 prediction using multimodal data. Preprints, 2020.
- [2] IHME COVID, Christopher JL Murray, et al. Forecasting covid-19 impact on hospital bed-days, icu-days, ventilator-days and deaths by us state within the next 4 months. MedRxiv, 2020.
- [3] Ensheng Dong, Hongru Du, and Lauren Gardner. An interactive webbased dashboard to trace covid-19 in real time. The Lancet infectious diseases, 20(5):533– 534, 2020.
- [4] Rasheed Omobolaji Alabi, Akpojoto Siemuri, and Mohammed Elmusrati. Covid-19: Easing the corona virus lockdown swith caution. medRxiv, 2020.
- [5] Du Toit, A., Outbreak of a completely unique coronavirus. Nature Reviews Microbiology, 2020. 18(3): p. 123-123. Bogoch, I.I., et al., Pneumonia of Unknown Etiology in Wuhan, China: Potential for International Spread Via. 2020.
- [6] Matheus Henrique Dal Molin Ribeiro, Ramon Gomes da Silva, Viviana Cocco Mariani, and Leandro dos Santos Coelho. Short-term forecasting covid-19 cumulative confirmed cases: Perspectives for brazil. Chaos, Solitons & Fractals, page 109853, 2020.
- [7] Huang, C., et al., Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. The Lancet, 2020. 395(10223): p. 497-506.
- [8] Wang, W., J. Tang, and F. Wei, Updated understanding of the outbreak of 2019 novel coronavirus (2019-nCoV) in Wuhan, China. Journal of medical virology, 2020. 92(4): p. 441-447.
- [9] Features, Evaluation and Treatment Coronavirus (COVID-19) by Marco Cascella; Michael Rajnik; Arturo Cuomo; Scott C. Dulebohn; Raffaella Di Napoli

APPENDIX

A) SOURCE CODE

Importing the required libraries

```
# importing the required libraries
import pandas as pd

# Visualisation libraries
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import plotly.express as px
import plotly.graph_objects as go
import folium
from folium import plugins

# Manipulating the default plot size
plt.rcParams['figure.figsize'] = 10, 12

# Disable warnings
import warnings
warnings.filterwarnings('ignore')
```

Part 1: Analysing the present condition in India

Reading the Datasets

```
#Learn how to read a .xls file by creating a dataframe using pandas
# Reading the datasets
df= pd.read_excel('/content/Covid cases in India.xlsx')
df_india = df.copy()
df

# Coordinates of India States and Union Territories
India_coord = pd.read_excel('/content/Indian Coordinates.xlsx')

#Day by day data of India, Korea, Italy and Wuhan
dbd_India = pd.read_excel('/content/per_day_cases.xlsx',parse_dates=True, sheet_name='India')
dbd_Italy = pd.read_excel('/content/per_day_cases.xlsx',parse_dates=True, sheet_name="Italy")
```

```
dbd_Korea = pd.read_excel('/content/per_day_cases.xlsx', parse_dates=True, sheet_name="Korea")
dbd_Wuhan = pd.read_excel('/content/per_day_cases.xlsx', parse_dates=True, sheet_name="Wuhan")
```

Analysing COVID19 Cases in India

```
#Learn how to play around with the dataframe and create a new attribute of 'Total Case'
#Total case is the total number of confirmed cases (Indian National + Foreign National)

df.drop(['S. No.'], axis=1, inplace=True)
df['Total cases'] = df['Total Confirmed cases (Indian National)'] + df['Total Confirmed cases ( Foreign National )']
total_cases = df['Total cases'].sum()
print('Total number of confirmed COVID 2019 cases across India till date (22nd March, 2020):', total_cases)
#Learn how to highlight your dataframe
df.style.background_gradient(cmap='Reds')
```

Number of Active COVID-19 cases in affected State/Union Territories

```
#Total Active is the Total cases - (Number of death + Cured)
df['Total Active'] = df['Total cases'] - (df['Death'] + df['Cured'])
total_active = df['Total Active'].sum()
print('Total number of active COVID 2019 cases across India:', total_active)
Tot_Cases = df.groupby('Name of State / UT')['Total Active'].sum().sort_values(ascending=False).to_frame()
Tot_Cases.style.background_gradient(cmap='Reds')
```

Visualising the spread geographically

```
# Learn how to use folium to create a zoomable map
df_full = pd.merge(India_coord, df, on='Name of State / UT')
map = folium.Map(location=[20, 70], zoom_start=4, tiles='Stamenterrain')

for lat, lon, value, name in zip(df_full['Latitude'], df_full['Longitude'], df_full['Total cases'], df_full['Name of State / UT']):
    folium.CircleMarker([lat, lon], radius=value*0.8, popup = ('<strong>State</strong>: ' + str(name).capitalize() + '<br>'<strong>Total Cases</strong>: ' + str(value) + '<br>'), color='red', fill_color='red', fill_opacity=0.3 ).add_to(map)
map
```

Confirmed vs Recovered figures

```
#Learn how to use Seaborn for visualization
f, ax = plt.subplots(figsize=(12, 8))
data = df_full[['Name of State / UT', 'Total cases', 'Cured', 'Death']]
```

```

data.sort_values('Total cases',ascending=False,inplace=True)
sns.set_color_codes("pastel")
sns.barplot(x="Total cases", y="Name of State / UT", data=data,label="Total", color="r")

sns.set_color_codes("muted")
sns.barplot(x="Cured", y="Name of State / UT", data=data, label="Cured", color="g")

# Add a legend and informative axis label
ax.legend(ncol=2, loc="lower right", frameon=True)
ax.set(xlim=(0, 35), ylabel="", xlabel="Cases")
sns.despine(left=True, bottom=True)

```

How the Coronavirus cases are rising?

```

#This cell's code is required when you are working with plotly on colab
import plotly
plotly.io.renderers.default = 'colab'
#Learn how to create interactive graphs using plotly
# import plotly.graph_objects as go
# Rise of COVID-19 cases in India
fig = go.Figure()
fig.add_trace(go.Scatter(x=dbd_India['Date'], y = dbd_India['Total Cases'], mode='lines+
markers',name='Total Cases'))
fig.update_layout(title_text='Trend of Coronavirus Cases in India (Cumulative cases)',pl
ot_bgcolor='rgb(230, 230, 230)')
fig.show()

# New COVID-19 cases reported daily in India

import plotly.express as px
fig = px.bar(dbd_India, x="Date", y="New Cases", barmode='group', height=400)
fig.update_layout(title_text='Coronavirus Cases in India on daily basis',plot_bgcolor='r
gb(230, 230, 230)')

fig.show()

```

Part 2: Is the trend similar to Italy/ S.Korea/ Wuhan?

Cumulative cases in India, Italy, S.Korea, and Wuhan

```

# import plotly.express as px
fig = px.bar(dbd_India, x="Date", y="Total Cases", color='Total Cases', orientation='v',
height=600,
title='Confirmed Cases in India', color_discrete_sequence = px.colors.cycli
cal.IceFire)

```

```

'''Colour Scale for plotly
https://plot.ly/python/builtin-colorscales/
'''

fig.update_layout(plot_bgcolor='rgb(230, 230, 230)')
fig.show()

fig = px.bar(dbd_Italy, x="Date", y="Total Cases", color='Total Cases', orientation='v',
             height=600,
             title='Confirmed Cases in Italy', color_discrete_sequence = px.colors.cycli
cal.IceFire)

fig.update_layout(plot_bgcolor='rgb(230, 230, 230)')
fig.show()

fig = px.bar(dbd_Korea, x="Date", y="Total Cases", color='Total Cases', orientation='v',
             height=600,
             title='Confirmed Cases in South Korea', color_discrete_sequence = px.colors
.cyclical.IceFire)

fig.update_layout(plot_bgcolor='rgb(230, 230, 230)')
fig.show()

fig = px.bar(dbd_Wuhan, x="Date", y="Total Cases", color='Total Cases', orientation='v',
             height=600,
             title='Confirmed Cases in Wuhan', color_discrete_sequence = px.colors.cycli
cal.IceFire)

fig.update_layout(plot_bgcolor='rgb(230, 230, 230)')
fig.show()

```

Comparison between the rise of cases in Wuhan, S.Korea, Italy and India

```

#Learn how to create subplots using plotly
# import plotly.graph_objects as go
from plotly.subplots import make_subplots

fig = make_subplots(
    rows=2, cols=2,
    specs=[[{}], {}],
    [{"colspan": 2}, None],
    subplot_titles=("S.Korea", "Italy", "India", "Wuhan"))

fig.add_trace(go.Bar(x=dbd_Korea['Date'], y=dbd_Korea['Total Cases'],

```

```

        marker=dict(color=dbd_Korea['Total Cases'], coloraxis="coloraxis")),
1, 1)

fig.add_trace(go.Bar(x=dbd_Italy['Date'], y=dbd_Italy['Total Cases'],
                    marker=dict(color=dbd_Italy['Total Cases'], coloraxis="coloraxis")),
1, 2)

fig.add_trace(go.Bar(x=dbd_India['Date'], y=dbd_India['Total Cases'],
                    marker=dict(color=dbd_India['Total Cases'], coloraxis="coloraxis")),
2, 1)

# fig.add_trace(go.Bar(x=dbd_Wuhan['Date'], y=dbd_Wuhan['Total Cases'],
#                     marker=dict(color=dbd_Wuhan['Total Cases'], coloraxis="coloraxis")
# ),2, 2)

fig.update_layout(coloraxis=dict(colorscale='Bluered_r'), showlegend=False, title_text="Total Confirmed cases (Cumulative)")

fig.update_layout(plot_bgcolor='rgb(230, 230, 230)')
fig.show()

```

Trend after crossing 100 cases

```

# import plotly.graph_objects as go

title = 'Main Source for News'
labels = ['S.Korea', 'Italy', 'India']
colors = ['rgb(122,128,0)', 'rgb(255,0,0)', 'rgb(49,130,189)']

mode_size = [10, 10, 12]
line_size = [1, 1, 8]

fig = go.Figure()

fig.add_trace(go.Scatter(x=dbd_Korea['Days after surpassing 100 cases'],
                        y=dbd_Korea['Total Cases'], mode='lines',
                        name=labels[0],
                        line=dict(color=colors[0], width=line_size[0]),
                        connectgaps=True))

fig.add_trace(go.Scatter(x=dbd_Italy['Days after surpassing 100 cases'],
                        y=dbd_Italy['Total Cases'], mode='lines',
                        name=labels[1],
                        line=dict(color=colors[1], width=line_size[1]),
                        connectgaps=True))

```

```

fig.add_trace(go.Scatter(x=dbd_India['Days after surpassing 100 cases'],
                        y=dbd_India['Total Cases'],mode='lines',
                        name=labels[2],
                        line=dict(color=colors[2], width=line_size[2]),
                        connectgaps=True))

annotations = []

annotations.append(dict(xref='paper', yref='paper', x=0.5, y=-0.1,
                        xanchor='center', yanchor='top',
                        text='Days after crossing 100 cases ',
                        font=dict(family='Arial',
                                size=12,
                                color='rgb(150,150,150)'),
                        showarrow=False))

fig.update_layout(annotations=annotations,plot_bgcolor='white',yaxis_title='Cumulative c
ases')
fig.show()

```

Part 3: Exploring World wide data

```

df = pd.read_csv('/content/covid_19_clean_complete.csv',parse_dates=['Date'])
df.rename(columns={'ObservationDate':'Date', 'Country/Region':'Country'}, inplace=True)

df_confirmed = pd.read_csv("/content/time_series_covid19_confirmed_global.csv")
df_recovered = pd.read_csv("/content/time_series_covid19_recovered_global.csv")
df_deaths = pd.read_csv("/content/time_series_covid19_deaths_global.csv")

df_confirmed.rename(columns={'Country/Region':'Country'}, inplace=True)
df_recovered.rename(columns={'Country/Region':'Country'}, inplace=True)
df_deaths.rename(columns={'Country/Region':'Country'}, inplace=True)
df_deaths.head()
df.head()
df2 = df.groupby(["Date", "Country", "Province/State"])[['Date', 'Province/State', 'Country', 'Confirmed', 'Deaths', 'Recovered']].sum().reset_index()
df2.head()
# Check for India's data
df.query('Country=="India"').groupby("Date")[['Confirmed', 'Deaths', 'Recovered']].sum().reset_index()
#Overall worldwide Confirmed/ Deaths/ Recovered cases
df.groupby('Date').sum().head()

```

Visualizing: Worldwide NCOVID-19 cases

```
confirmed = df.groupby('Date').sum()['Confirmed'].reset_index()
deaths = df.groupby('Date').sum()['Deaths'].reset_index()
recovered = df.groupby('Date').sum()['Recovered'].reset_index()
fig = go.Figure()
#Plotting datewise confirmed cases
fig.add_trace(go.Scatter(x=confirmed['Date'], y=confirmed['Confirmed'], mode='lines+markers', name='Confirmed', line=dict(color='blue', width=2)))
fig.add_trace(go.Scatter(x=deaths['Date'], y=deaths['Deaths'], mode='lines+markers', name='Deaths', line=dict(color='Red', width=2)))
fig.add_trace(go.Scatter(x=recovered['Date'], y=recovered['Recovered'], mode='lines+markers', name='Recovered', line=dict(color='Green', width=2)))
fig.update_layout(title='Worldwide NCOVID-19 Cases', xaxis_tickfont_size=14, yaxis=dict(title='Number of Cases'))

fig.show()
```

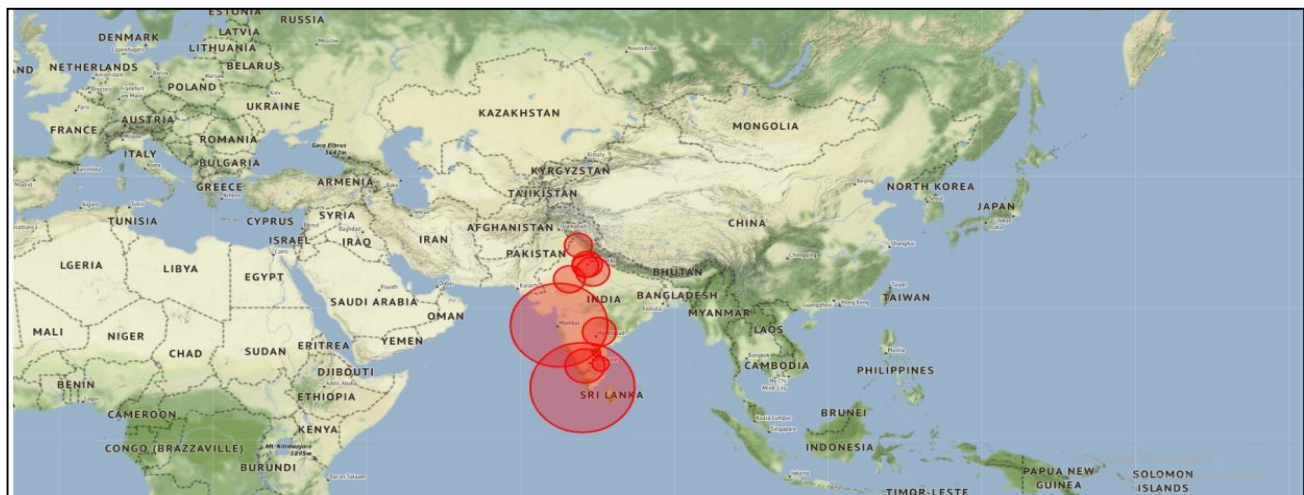
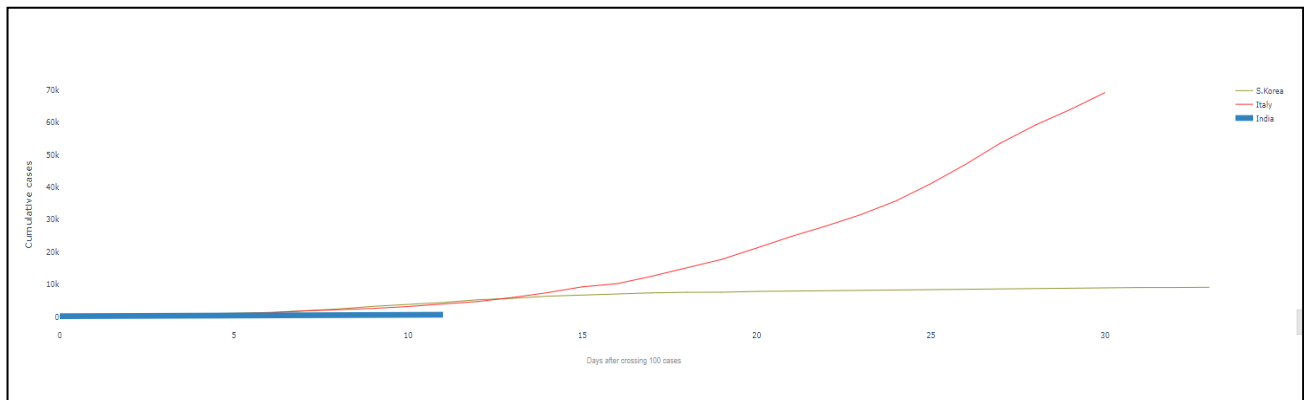
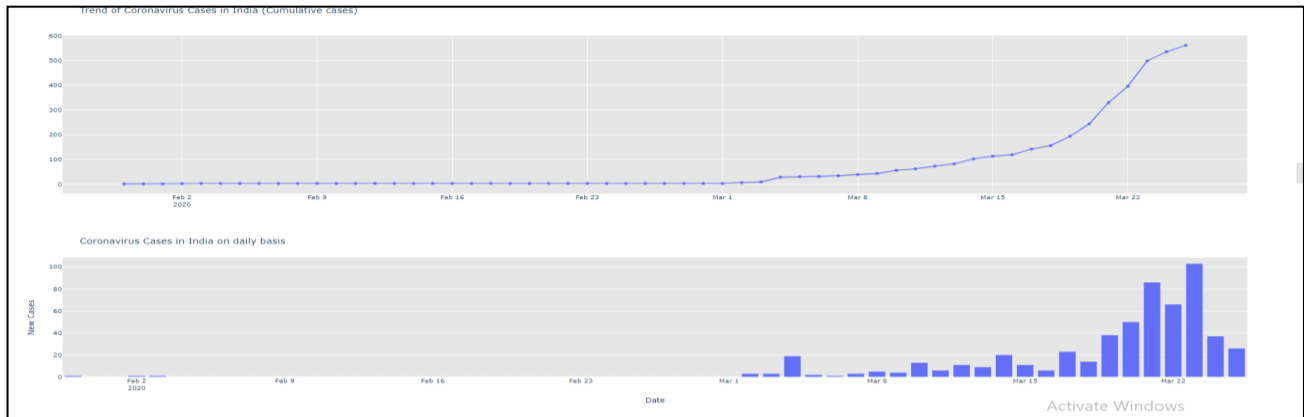
Part 4: Forecasting Total Number of Cases Worldwide

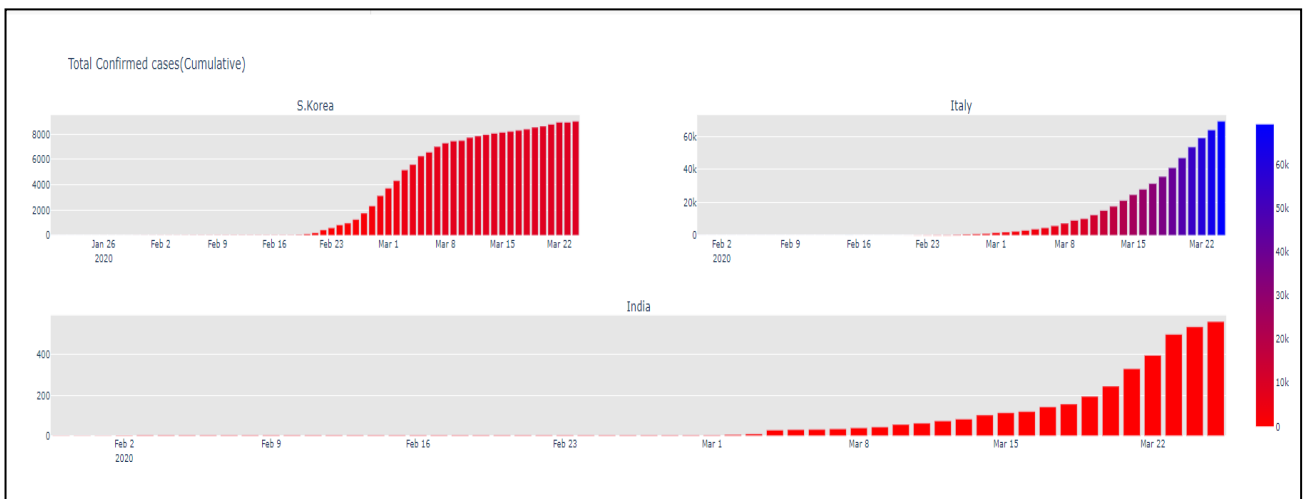
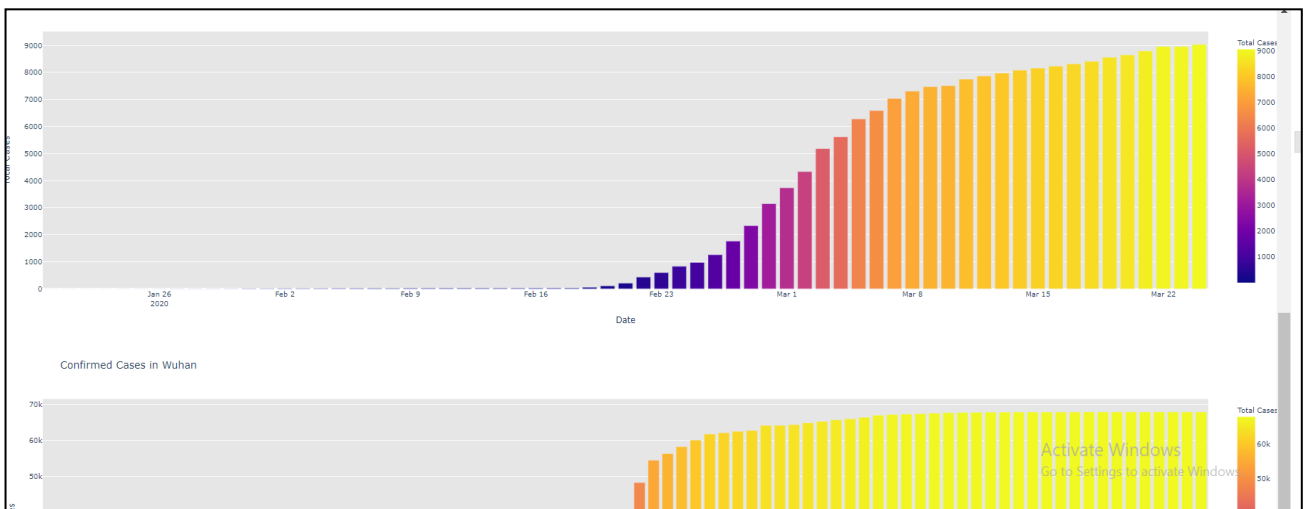
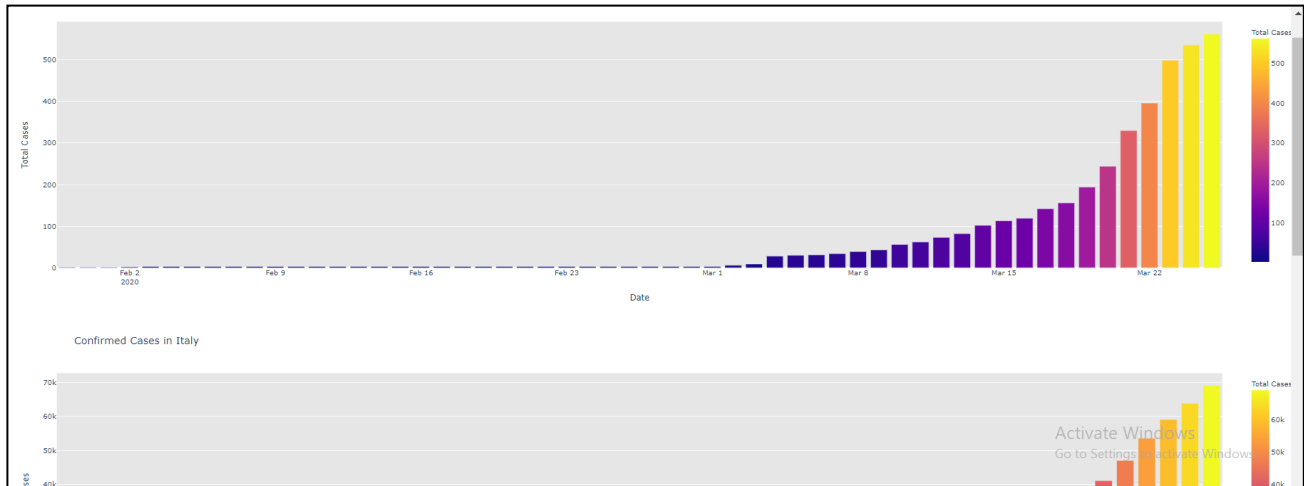
```
from fbprophet import Prophet
confirmed = df.groupby('Date').sum()['Confirmed'].reset_index()
deaths = df.groupby('Date').sum()['Deaths'].reset_index()
recovered = df.groupby('Date').sum()['Recovered'].reset_index()
confirmed.columns = ['ds', 'y']
#confirmed['ds'] = confirmed['ds'].dt.date
confirmed['ds'] = pd.to_datetime(confirmed['ds'])
confirmed.tail()
```

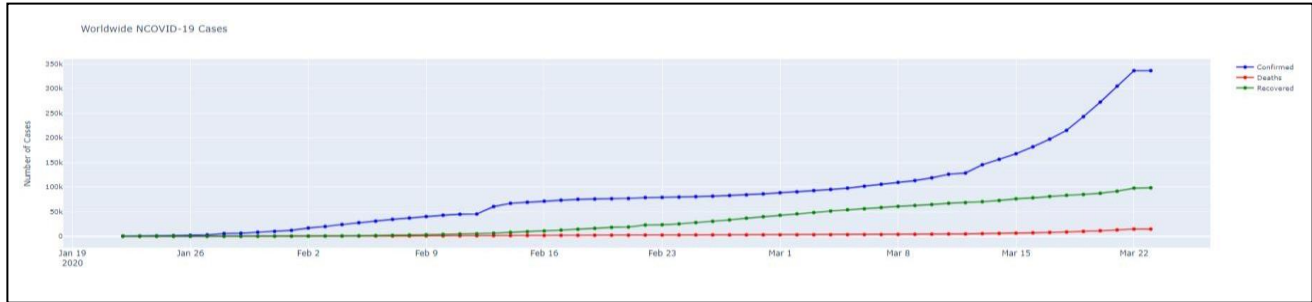
Forecasting Confirmed NCOVID-19 Cases Worldwide with Prophet (Base model)

```
m = Prophet(interval_width=0.95)
m.fit(confirmed)
future = m.make_future_dataframe(periods=7)
future.tail()
#predicting the future with date, and upper and lower limit of y value
forecast = m.predict(future)
forecast[['ds', 'yhat', 'yhat_lower', 'yhat_upper']].tail()
confirmed_forecast_plot = m.plot(forecast)
confirmed_forecast_plot = m.plot_components(forecast)
```

B) SCREENSHOTS







C) PLAGIARISM REPORT

plag word (2).docx			
ORIGINALITY REPORT			
6%	5%	4%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
PRIMARY SOURCES			
1	pypi.org Internet Source		1%
2	M.S. Sanaj, P.M. Joe Prathap. "An efficient approach to the map-reduce framework and genetic algorithm based whale optimization algorithm for task scheduling in cloud computing environment", Materials Today: Proceedings, 2020 Publication		1%
3	plus.mcmaster.ca Internet Source		1%
4	www.warse.org Internet Source		1%
5	Submitted to Delhi Technological University Student Paper		1%
6	arxiv.org Internet Source		1%
7	github.com Internet Source		<1%
8	www.ijert.org Internet Source		<1%