# SARS-CoV-2 (COVID19) Analysis and Prediction


## Abstract

One might consider Nuclear Warfare and a Climatic catastrophe as the biggest failure that humanity could ever commit but, the emergence of an infectious disease has the potential to wipe out a major portion of human existence within no time. Despite the intense studies on the patterns of theses epidemic outbreaks, when, where and how these outbreaks trigger is out of the comprehension.A severe respiratory disease was recently reported in Wuhan, Hubei province, China. As of 25 January 2020, at least 1,975 cases had been reported since the first patient was hospitalised on 12 December 2019.After the phylogenetic analysis of the complete viral genome it was found to be closely related to SARS like virus which is related to the family **Coronaviridae**. This outbreak highlights the ongoing ability of viral spill-over from animals to cause severe disease in humans.


## Introduction - I

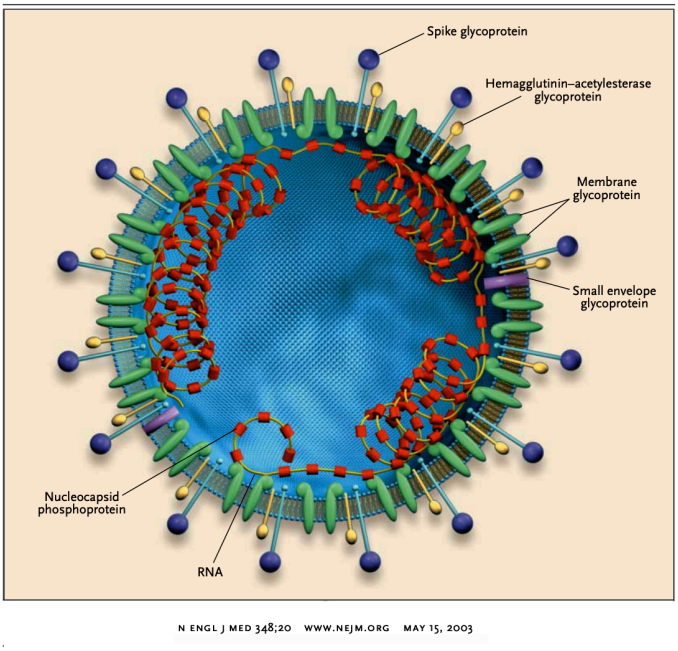
Science is very complex, in fact, we are entities made up of science itself, living in the medium of science with an idiosyncratic consciousness. The reason behind achieving the thought process employed behind this report is, how this distressing time period has made everything purely correlational. This biological event, when evaluated mathematically, we can derive patterns that could lead to potentially positive insights. We the human beings have the ability to understand and manipulate space-time continuum as quantity through imagination. All that we have to do is to deploy this thought process at the right instance to derive solutions. This report gives you, intricate Knowledge of every aspect of science integrated in this crisis period. Note a fact that this Mathematical Analysis is based on a Staple Dataset. The results derived from the analysis made known in this report will be updated over time. We presume that this report will enlighten you with the right attitude required in this crisis period.

## Origin of Corona Virus

The origin of the novel SARS-CoV-2 outbreak in China is a tad bit Controversial, but it is for a fact emerged naturally, Coronaviruses are large enveloped single-strained RNA viruses, their natural reservoirs are believed to be Horseshoe Bats, mice, birds, pigs and cattle. It is one of the viruses which is capable of **Zoonosis,** which means the infection in the any of the above non-human animals (vertebrates) can transmit to humans. In human beings, five respiratory coronaviruses have been described, causing common cold, upper respiratory tract infections, or pneumonia. In September, 2012, a novel human coronavirus, named HCoV-EMC, was identified in two patients with severe respiratory disease.The patients infected with this novel disease were observed to develop symptoms that are closely related to SARS like corona which also out broke in the year 2002 in china with 8,098 confirmed cases and 774 deaths total and the fatality rate was estimated about 50% on the basis of outbreak dynamics it was later renamed as MERS-Cov(Middle East Respiratory Syndrome) by the International committee on Taxonomy of Viruses. This proved that this virus is zoonotic it genetically mutated and became more potent.

Every virus ever originated has a possibility to genetically mutate itself to become more potent over the time provided if it has the perfect biological factors required. Talking about the mutation The Novel Corona virus, now called the COVID19 is also a genetically mutated version of SARS-CoV-1(2002). It has an estimated fatality rate of **3.3%**, Patients contracted with the virus doesn't have any Symptoms initially at the recent stage, It has an Incubation period of 14days to show any potential symptoms such as cold, cough, fatigue etc. Asymptomatic transmission is the key factor for the outbreak to cause a rapid spread across countries to further become a global ***Pandemic.***

## Structure of Corona Virus Virion




N ENGL J MED 348;20 WWW.NEJM.ORG MAY 15, 2003

A Virus is basically a vessel containing the genetic material and a few proteins arguably not even a living being. It can only duplicate by the entering a living self making it a possible host. Corona may spread via surfaces of different objects but it's still not evident how long the virus can survive. The main way of spreading seems to be droplet infection, when people cough and you come into the nearest vicinity of it then you might contract the virus through the droplets or rubbing your eyes or noes after touching the surfaces with virus.

## Journey of the Virus

The Virus starts its journey from the nose or the eyes, it then rides deeper into the body, its potential destinations for colonising are the spleen and lungs. The lungs on the other hand is the most probable destination, lungs are lined with billions of Epithelial Cells which are the border cells of the body lining the organs and mucosa which and are most vulnerable to get infected. In the figure is the structure of corona virus virion the Spike Glycoprotein acts as the key to get access to the cell lining, this Spike protein connects to a specific receptor (ACE2) and injects the genetic material the virus is carrying into the cell, the cell not knowledgeable of what happening

considers it as a ‘New-Instruction’ and ‘Executes’ it, and the instructions are simply ‘Copy and Duplicate’. After threshold is reached it makes one final order to self destruct which releases the viruses to infect more cells, this happens recursively until the immune system starts to react to it.

# Data Analysis

## Information Gathering

For every Data Analysis the first thing we need is to Collect data and organise into a 2-Dimensional Dataset. In my case Data has already been amalgamated into a .CSV file format.

I used Jupyter Notebook for the analysis which is an open source integrated environment for Python3 to apply the Data Analytics, Math, Visualisation and Machine Learning Algorithms efficiently.

## Importing the Required Modules.

### Required Modules

In [56]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.model_selection import train_test_split
from covid import Covid
import datetime
from matplotlib import style
style.use('fivethirtyeight')

```

## Reading the Data

The Datasets used in the analysis are **covid_19_data.csv, time_series_covid19_confirmed.csv, time_series_covid19_recovered.csv, time_series_covid19_deaths.csv.** The main dataset of all covid_19_data.csv is majorly used in the relational and visual analytics. Initially the data is read using the Pandas Library using **pd.read_csv()** function.

# Required Datasets

In [47]:
```
1 confirmed_df.head(10) # time_series_covid19_confirmed.csv
```

| | Province/State | Country/Region | Lat | Long | 1/22/20 | 1/23/20 | 1/24/20 | 1/25/20 | 1/26/20 | 1/27/20 | ... | 3/18/20 | 3/19/20 | 3/20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NaN | Afghanistan | 33.0000 | 65.0000 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 22 | 22 | 24 |
| 1 | NaN | Albania | 41.1533 | 20.1683 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 59 | 64 | 70 |
| 2 | NaN | Algeria | 28.0339 | 1.6596 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 74 | 87 | 90 |
| 3 | NaN | Andorra | 42.5063 | 1.5218 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 39 | 53 | 75 |
| 4 | NaN | Angola | -11.2027 | 17.8739 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 1 |
| 5 | NaN | Antigua and Barbuda | 17.0608 | -61.7964 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 1 | 1 | 1 |
| 6 | NaN | Argentina | -38.4161 | -63.6167 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 79 | 97 | 128 |
| 7 | NaN | Armenia | 40.0691 | 45.0382 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 84 | 115 | 136 |
| 8 | Australian Capital Territory | Australia | -35.4735 | 149.0124 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 3 | 4 | 6 |
| 9 | New South Wales | Australia | -33.8688 | 151.2093 | 0 | 0 | 0 | 0 | 3 | 4 | ... | 267 | 307 | 353 |

10 rows × 70 columns

In [49]:
```
1 deaths_df.head(10) # time_series_covid19_deaths.csv
```

| | Province/State | Country/Region | Lat | Long | 1/22/20 | 1/23/20 | 1/24/20 | 1/25/20 | 1/26/20 | 1/27/20 | ... | 3/18/20 | 3/19/20 | 3/20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NaN | Afghanistan | 33.0000 | 65.0000 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 1 | NaN | Albania | 41.1533 | 20.1683 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 2 | 2 | 2 |
| 2 | NaN | Algeria | 28.0339 | 1.6596 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 7 | 9 | 11 |
| 3 | NaN | Andorra | 42.5063 | 1.5218 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 4 | NaN | Angola | -11.2027 | 17.8739 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 5 | NaN | Antigua and Barbuda | 17.0608 | -61.7964 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 6 | NaN | Argentina | -38.4161 | -63.6167 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 2 | 3 | 3 |
| 7 | NaN | Armenia | 40.0691 | 45.0382 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 8 | Australian Capital Territory | Australia | -35.4735 | 149.0124 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 9 | New South Wales | Australia | -33.8688 | 151.2093 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 5 | 5 | 6 |

10 rows × 70 columns

In [48]:
```
1 recovered_df.head(10) # time_series_covid19_recovered.csv
```

| | Province/State | Country/Region | Lat | Long | 1/22/20 | 1/23/20 | 1/24/20 | 1/25/20 | 1/26/20 | 1/27/20 | ... | 3/18/20 | 3/19/20 | 3/20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NaN | Afghanistan | 33.0000 | 65.0000 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 1 | 1 | 1 |
| 1 | NaN | Albania | 41.1533 | 20.1683 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 2 | NaN | Algeria | 28.0339 | 1.6596 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 12 | 32 | 32 |
| 3 | NaN | Andorra | 42.5063 | 1.5218 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 1 | 1 | 1 |
| 4 | NaN | Angola | -11.2027 | 17.8739 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 5 | NaN | Antigua and Barbuda | 17.0608 | -61.7964 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 6 | NaN | Argentina | -38.4161 | -63.6167 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 3 | 3 | 3 |
| 7 | NaN | Armenia | 40.0691 | 45.0382 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 1 | 1 | 1 |
| 8 | Australian Capital Territory | Australia | -35.4735 | 149.0124 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| 9 | New South Wales | Australia | -33.8688 | 151.2093 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 4 | 4 | 4 |

10 rows × 70 columns

# Analysis on the Cumulative Data

## Getting to know the Dataset

In [4]:
```
data = pd.read_csv('covid_19_data.csv')
data.head(10)
```

| | SNo | ObservationDate | Province/State | Country/Region | Last Update | Confirmed | Deaths | Recovered |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 01/22/2020 | Anhui | Mainland China | 1/22/2020 17:00 | 1.0 | 0.0 | 0.0 |
| 1 | 2 | 01/22/2020 | Beijing | Mainland China | 1/22/2020 17:00 | 14.0 | 0.0 | 0.0 |
| 2 | 3 | 01/22/2020 | Chongqing | Mainland China | 1/22/2020 17:00 | 6.0 | 0.0 | 0.0 |
| 3 | 4 | 01/22/2020 | Fujian | Mainland China | 1/22/2020 17:00 | 1.0 | 0.0 | 0.0 |
| 4 | 5 | 01/22/2020 | Gansu | Mainland China | 1/22/2020 17:00 | 0.0 | 0.0 | 0.0 |
| 5 | 6 | 01/22/2020 | Guangdong | Mainland China | 1/22/2020 17:00 | 26.0 | 0.0 | 0.0 |
| 6 | 7 | 01/22/2020 | Guangxi | Mainland China | 1/22/2020 17:00 | 2.0 | 0.0 | 0.0 |
| 7 | 8 | 01/22/2020 | Guizhou | Mainland China | 1/22/2020 17:00 | 1.0 | 0.0 | 0.0 |
| 8 | 9 | 01/22/2020 | Hainan | Mainland China | 1/22/2020 17:00 | 4.0 | 0.0 | 0.0 |
| 9 | 10 | 01/22/2020 | Hebei | Mainland China | 1/22/2020 17:00 | 1.0 | 0.0 | 0.0 |

In [5]:
```
print("Number of Datapoints : {}".format(data.size))
print("Shape of the DataSet : {}".format(data.shape))
```

```
Number of Datapoints : 75392
Shape of the DataSet : (9424, 8)
```

# Initial insights and Data preprocessing

At the beginning of any analysis it is essential to know the Datatypes involved, Number of Data points recorded in the DataFrame, number of Features collected. The above dataset is organised to a shape (9424,8)
9424 defines the number of *rows* recorded and the 8 refers to the number of *features/columns* collected.

## Knowing the Datatypes of the Datapoints involved in the Dataset

In [7]:
```
# knowing the datatypes
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9424 entries, 0 to 9423
Data columns (total 8 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   SNo              9424 non-null   int64  
 1   ObservationDate  9424 non-null   object 
 2   Province/State   5164 non-null   object 
 3   Country/Region   9424 non-null   object 
 4   Last Update      9424 non-null   object 
 5   Confirmed        9424 non-null   float64
 6   Deaths           9424 non-null   float64
 7   Recovered        9424 non-null   float64
dtypes: float64(3), int64(1), object(4)
memory usage: 589.1+ KB
```

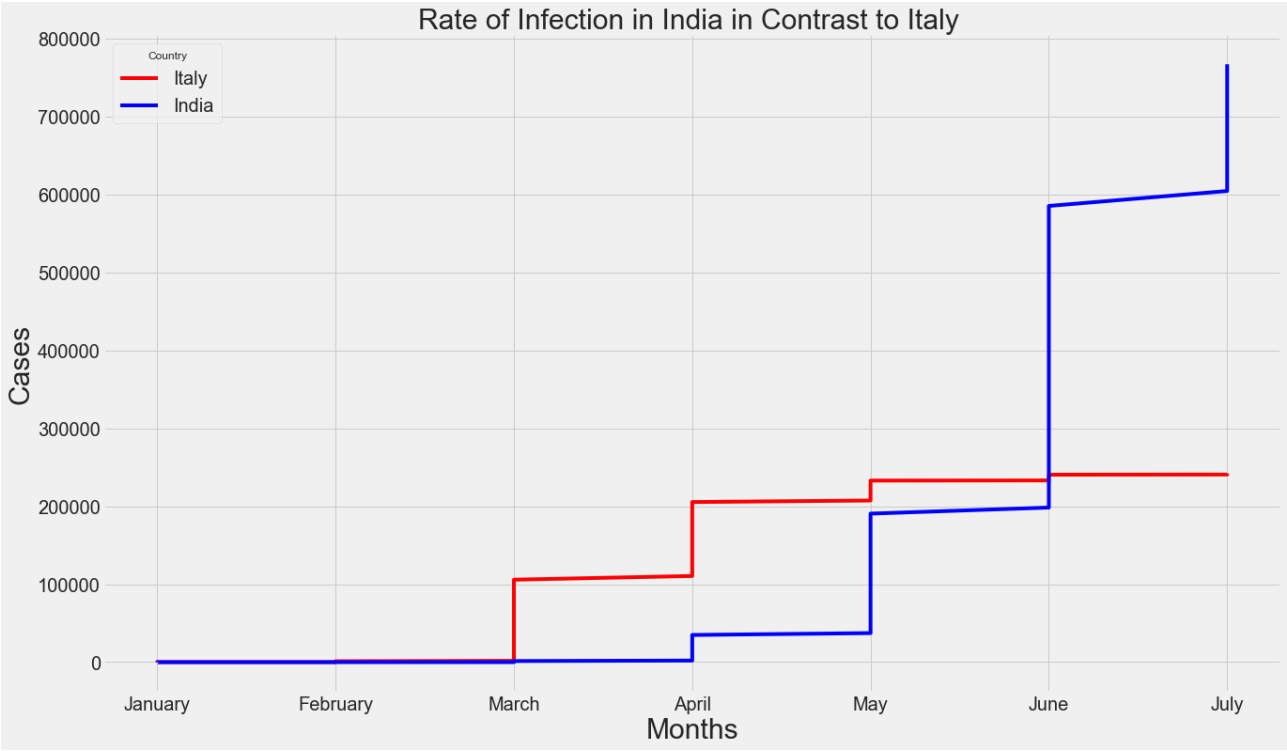
The **<DataFrameVar>.info()** method provided in the pandas library is quite efficient in checking the data types the data points involved and printing the result in an organised manner. Knowing the data types enable us to deal with missing values mathematically in few cases.

**Data Preprocessing** is an essential step in the sequential process of data analysis, Not every Dataset has perfectly recorded values, there are cases where a datapoint across a certain feature is left out un registered or even noted as special character. In the case of missing values Pandas library reads it as a **NaN** value which means **Not A Number.**

# Relational Analysis and Visualisation

The Three Numerical types, Confirmed, Recovered, Deaths in the Dataset are not correlated. So, it is unperceptive to plot graphs with two different and unrelated features . But, we do have multiple instances of a single feature with different values, meaning different countries with different values of a certain feature(Confirmed, Recovered, Deaths) we could make create a plot which gives us the Relational insights of two countries.

I chose India and Italy as the countries of desire because the contrast between two show us a major difference of the growth factor.
Graphs give us valuable insights instantly just by looking at them and so does this graph. We can clearly observe the growth differences.



# Building a Model

A Machine learning model mathematically realises and learns the patterns in a 2-dimensional dataset, so that any of the future data with no target values when passed into the model, then the model, considering the mathematical pattern previously revised, will be able to efficiently predict the target values with a certain accuracy based on the model complexity.

One can employ a particular Machine learning algorithm based on the previous conclusions made on data to classify the problem.
A predictive model could either be one of the two kinds,A Supervised learning model, or an Unsupervised learning model.In Our  case it is an Unsupervised learning model, since there are no predefined outputs given in the dataset itself.

# Predicting the Future

The collected dataset **covid_19_data.csv,** is a complex dataset with country names and dates as features. I chose my country of desire as ***India*** to predict the growth of this contagion to 30 days in the future from the last reported date in the dataset.

The 3 Different Time series datasets mentioned earlier are essential for this predictive modelling, since they are the foundation for this model.

## Reading and organising the required data.

values reported across the  country name down the reported dates, days since 1/22/2020.

Since we are building a model around one country, India, We choose to collect or separate only the values of India to a separate data frame object which is our main sample as seen in the

# Linear Regression Algorithm

The term linear refers to the relationship between the two variables, which when plotted gives us a straight line. Considering the two variables one containing the dates recorded and the other with confirmed cases, when plotted we can observe a linear growth.

The objective of a linear regression model is to find a relationship between on or more features(independent variables) and a continuous target variables(dependent variables).Basically, a linear regression model analyses the pattern in the train data, and predict the target values in the test data provided.

# Model Complexity and Dataset size Relationship

To building a perfect model which yields highest possible accuracy, there are two aspects one must consider re-evaluating they are the model complexity scenarios, Underfitting and Overfitting. Usually collecting more data points with more variety enables a scope to build more complex models, majority of the times those complex models yield a good result. We never feed a machine learning algorithm with the entire data we collected. In that case the algorithm returns only one possible result. So, in order to avoid that we split the data into *train_data* , and *test_data.*

In our scenario out dataset has good amount of data points with minimal loss during the preprocessing. So, it is possible to create a complex model to avoid overfitting.

# Polynomial Features

Our data doesn't show a perfect linear relationship, so consider fitting most data points as possible we use polynomial regression. To Increase the complexity of the features we can add powers of the original features as new features to generate a higher order equation

**General Linear Equation :**

$$Y = \theta_0 + \theta_1 x$$

**Transformed Linear Equation :**

$$Y = \theta_0 + \theta_1 x + \theta_2 x^2$$

## Polynomial Linear Regression

Polynomial Regression is a form of linear regression in which the relationship between the independent variable **x** and the dependent vairiable **y** is modeled as the **nth** degree polynomial. Polynmial Regression fits a non-linear relationship between the values of x and the corresponding conditional mean y.

In [231]:
```
poly = PolynomialFeatures(degree=5)
poly_X_train_confirmed = poly.fit_transform(days_since_1_22)
poly_future_forcast = poly.fit_transform(forcast)
```

In [232]:
```
linear_model = LinearRegression(normalize=True, fit_intercept=False)
linear_model.fit(poly_X_train_confirmed, confirmed_cases)
linear_pred = linear_model.predict(poly_future_forcast)
```

# Formatting the output

Zipping the predicted values and the 30 dates from the last reported date will organise the output as shown in Figure.

# The Final Result

```
linear_pred = linear_pred.reshape(1,-1)[0]
print('Polynomial regression future predictions in INDIA (30 days from last reported days in th
finalresult = set(zip(future_forcast_dates[-30:], np.round(linear_pred[-30:])))
print("\nDate Format : {:>11s}".format("%m/%d/%Y"))
finalresult
```

```
Polynomial regression future predictions in INDIA (30 days from last reported days in the dataset) : "Confirmed_Cases"

Date Format :    %m/%d/%Y


{('07/09/2020', 791280.0),
 ('07/10/2020', 817389.0),
 ('07/11/2020', 844234.0),
 ('07/12/2020', 871832.0),
 ('07/13/2020', 900204.0),
 ('07/14/2020', 929370.0),
 ('07/15/2020', 959348.0),
 ('07/16/2020', 990162.0),
 ('07/17/2020', 1021830.0),
 ('07/18/2020', 1054376.0),
 ('07/19/2020', 1087821.0),
 ('07/20/2020', 1122189.0),
 ('07/21/2020', 1157502.0),
 ('07/22/2020', 1193784.0),
 ('07/23/2020', 1231060.0),
 ('07/24/2020', 1269355.0),
 ('07/25/2020', 1308694.0),
 ('07/26/2020', 1349102.0),
 ('07/27/2020', 1390608.0),
 ('07/28/2020', 1433237.0),
 ('07/29/2020', 1477019.0),
 ('07/30/2020', 1521980.0),
 ('07/31/2020', 1568151.0),
 ('08/01/2020', 1615561.0),
 ('08/02/2020', 1664240.0),
 ('08/03/2020', 1714219.0),
 ('08/04/2020', 1765531.0),
 ('08/05/2020', 1818207.0),
 ('08/06/2020', 1872280.0),
 ('08/07/2020', 1927784.0)}
```

# Conclusion

Statistical analysis made in this report gives a perspective of the condition where without self-isolation, the possibility of a community spread of the infection is off the charts. Many countries as displayed above, are the major examples of the worst-case scenarios. Studying the result of the Predictive Analysis, the pattern of the growth of infection is exponential even without considering any of the cultural contexts of a country. This valuable insight to the situation tells us how the situation will worsen, if a proper protocol is not employed. The Optimal solution would be Self Isolation for any of the above numerical value to variate positively. This analysis contains a mere estimation of the future based on the past events. Everything weighs down onto us now, our actions today will reflect on the survival of mankind tomorrow. Stay Safe.

# References and Sources

[1] : Johns Hopkins University for making the data available for educational and academic research purposes

[2]: World Health Organisation (WHO): https://www.who.int/

[3]: New England Medical Journals (Picture Courtesy) : https://www.nejm.org/coronavirus

[4]: Andreas C. Müller and Sarah Guido, Introduction to Machine Learning with Python

[5]: Dynamic Data Source : https://www.worldometers.info/coronavirus/

[6]: Jupyter Notebook (Covid-19-Analysis,Prediction & Visualisation) : https://github.com/r0han99/Covid19-PredictiveAnalysis/

# Prepared by:

**Nalla Rohan Sai**, Bachelor of Technology at GITAM University, Visakhapatnam.

**Harshith Nikhil V Samudrala**, Bachelor of Technology at GITAM University, Visakhapatnam.