SARS-CoV-2 (COVID19) Analysis and Prediction

Prepared for

The Humanity

Prepared By

N.Rohan Sai

March 31,2020

Writers Note :-

Science is very complex, in fact, we are entities made up of science itself living in the medium of science with an individualistic consciousness. The reason behind achieving the thought process employed in this report was, how this intriguing time period has everything purely correlational. This biological event when evaluated mathematically, we 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 the Reader, Intermediate Knowledge of every aspect science integrated in this crisis period. Keep note of a fact that this Mathematical Analysis is based on a Staple Dataset, The results derived from the analysis produced in this report will be updated over time. I presume that this report will subconsciously feed you the right mindset required in this crisis period

~ Rohan;

Table of Contents:

Page

- List of DataFrames

- List of Graphs

- List of Figures

- Section - I

- Section - II

- Section - III

- Section - IV

– x – x – x – x –

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.

SARS-CoV-2 (COVID19) Analysis and Prediction

Section-I

Introduction

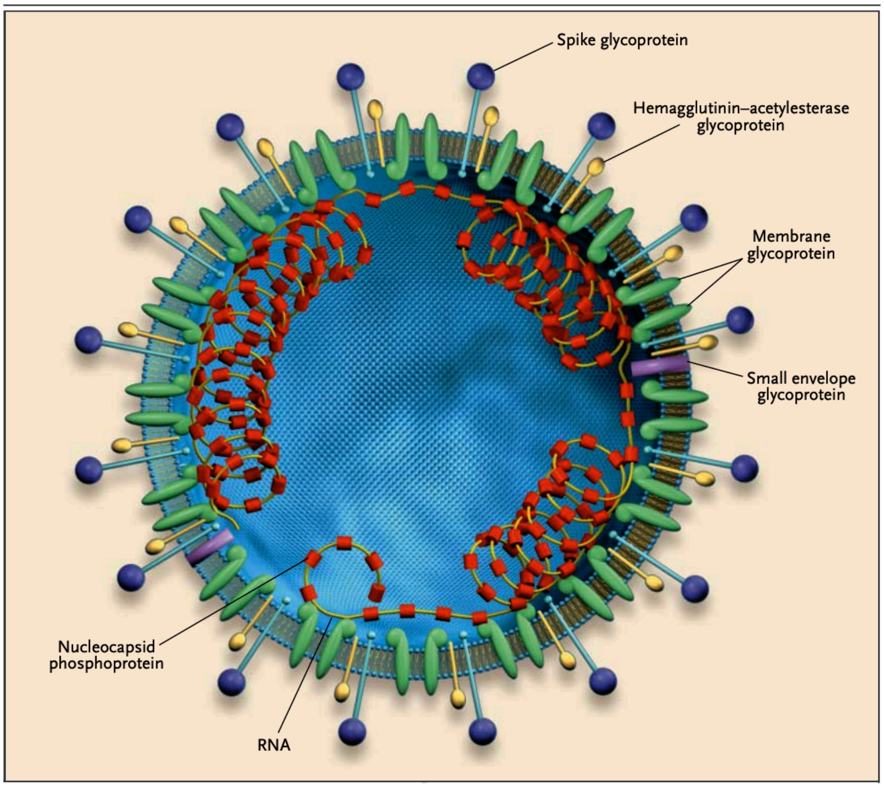
Origin of Corona Virus

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

Fig-1.0


Spike glycoprotein
Hemagglutinin–acetylesterase glycoprotein
Membrane glycoprotein
Small envelope glycoprotein
Nucleocapsid phosphoprotein
RNA

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

Causes of Contracting COVID19

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 1.0, 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.

Section-II

Take on Conspiracy Theories

A Conspiracy theory is ‘ Creating belief not fully acknowledging it publicly, that an influential organisation is responsible for the uncertain event’, but if we trace the origin of the thought behind the creation of this theory to the individual who thought of it and examine, its just, raw panic and anxiety which makes the person to re-evaluate the origin of that event and make baseless theories quite convincing to one’s self, rather than the other important things he has to attain because of creating one such made up theory on his/her own can relieve that one mind killer, *Anxiety*.

In context to the above explanation of a Conspiracy theory there are many statements made on this current COVID19 situation. Considering the biggest and most convincing one of all, there’s an idiotic belief which seems way more contagious than the virus itself

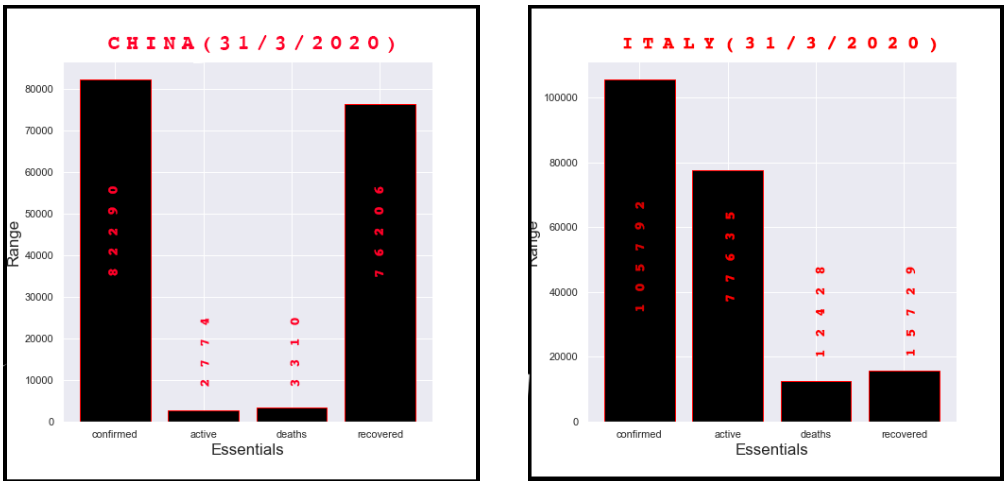
There were articles from the ***The Washington Times*** claiming that ‘The COVID19 Virus was artificially created in Wuhan’s Institute of Virology while conducting a Biological warfare program, and China is successfully covering it up’ and they later debunked it.

Why is it so Convincing after all ?

This theory is mostly bought by everyone even the United States President Mr.Donald Trump considered it a fact, he started calling it ‘The Chinese Virus’. Coming to the question why is it so convincing to people?

The general reasoning behind it would be the ‘*NUMBERS*’, people generally tend to mentally digest to accurate numbers than the perfect allegations itself. Considering numerical’s on the date this report was drafted.

Fig-1.1, Status of Spread of Infection in China and Italy



Evaluating the differences in both the graphs, we can observe that china has had a maximum recovery rate with 76206 people recovered form the disease on the other hand, looking at Italy's situation, its disturbing enough to believe any possible assumption ever made. In this situation if we put some basic reasoning and some research from true sources before considering numbers for a fact, the answer will be evident. A simple reasoning could be, China is medically well equipped country in contrast to Italy Now there are assumptions that a BCG vaccine which originated late 19 hundreds has potential to resist the Covid19 virus. It could be a possibility that china had a consistent vaccine programs in the past so that now that became a possible resistant for them.

Scientific Explanation to prove the theory wrong

Its true that we can create viruses and genetically modify the existing ones, but based on the phylogenetic analysis on the whole virus genome it was found that the RBD portion of COVID-19 has a the spike proteins(fig **1.0**) that have evolved to effectively target molecular features on the protective lining of a human cells called ACE2, a receptor involved in regulating the blood pressure. This feature is so effective at binding the Human cells that it is beyond the current technology related to Genetic engineering and most likely is a product of natural mutation or selection. Even if someone is seeking to engineer the pathogen, they would have to construct it from the backbone of the virus known to cause the illness. But it was found that SARS-CoV-2 backbone differed from those of previously acknowledged coronaviruses and mostly resembled to the virus found in their natural reservoirs, Bats and pangolins

Section-III

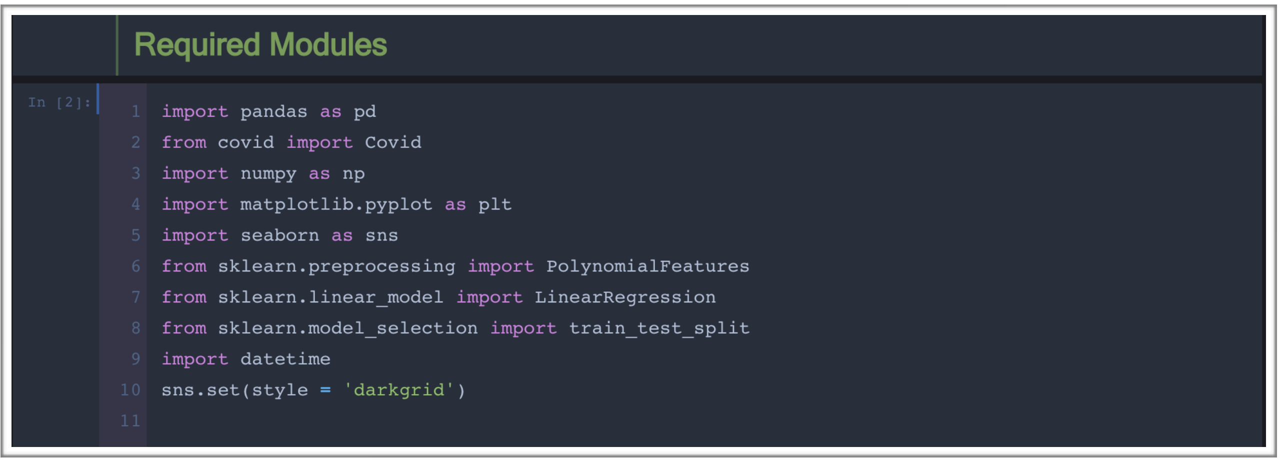
Data Analysis

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 [2]:

```
import pandas as pd
from covid import Covid
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.model_selection import train_test_split
import datetime
sns.set(style = 'darkgrid')

```

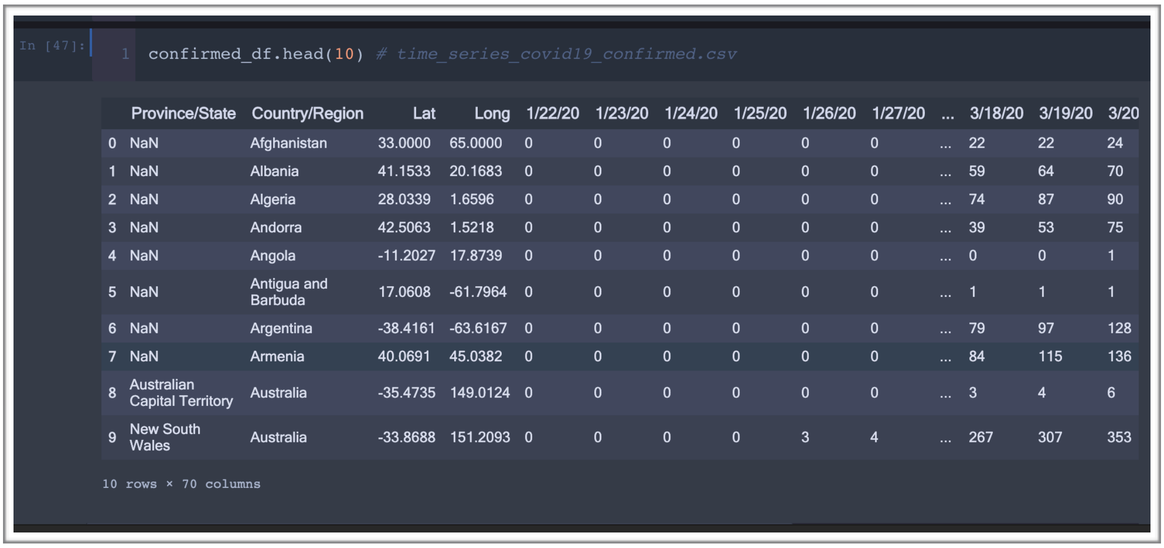
Fig-1.2 Modules and Libraries.

Covid is a package, it is used to dynamically fetch the entire data regarding the COVID19 effected countries.

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(**Fig-1.3)**

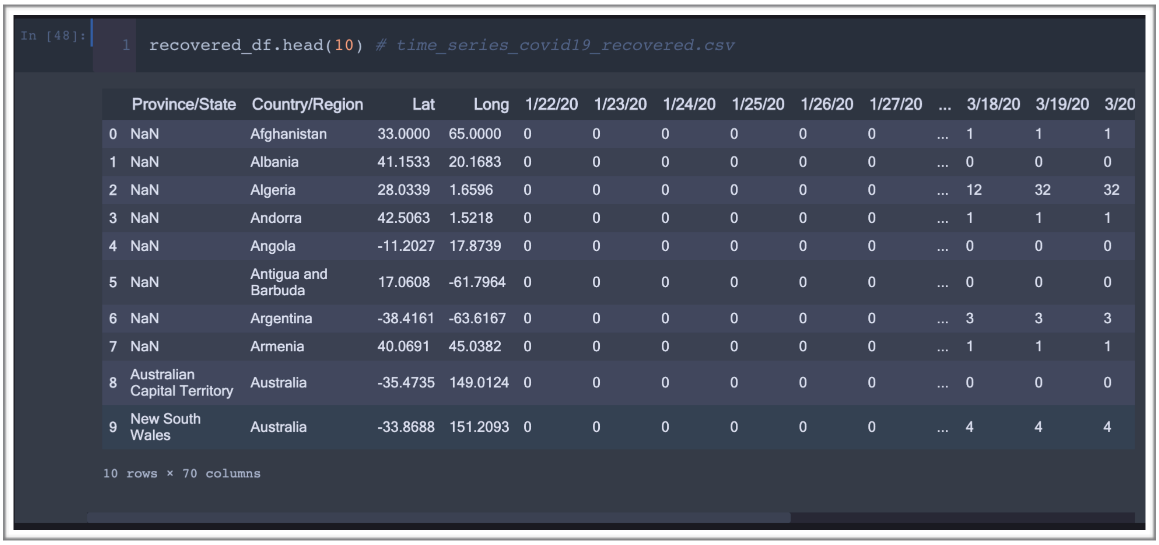
Required Datasets



```
In [47]:  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

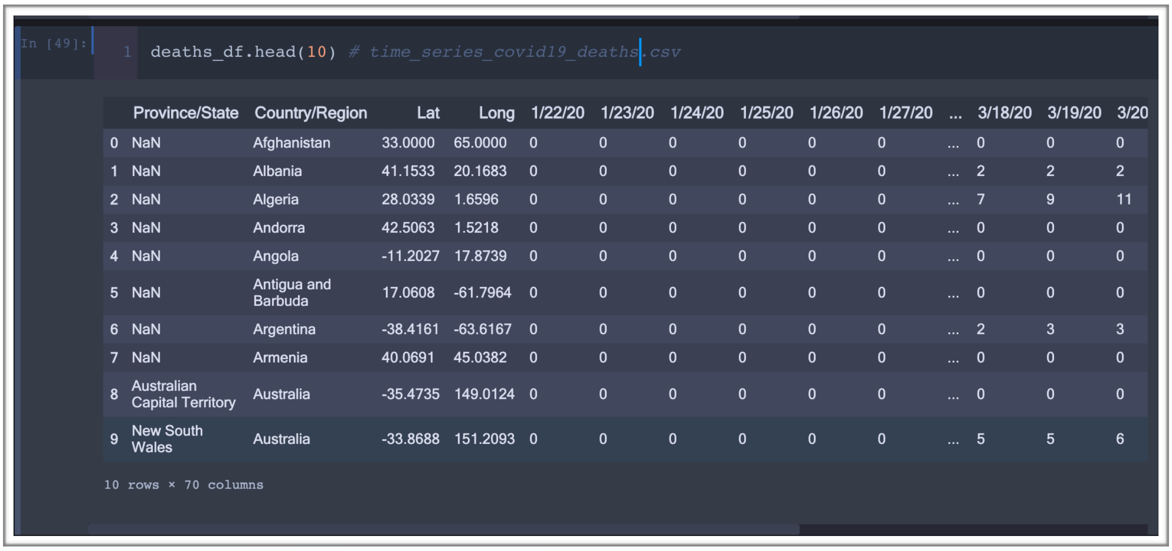
```
10 rows × 70 columns
```



```
In [48]:  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

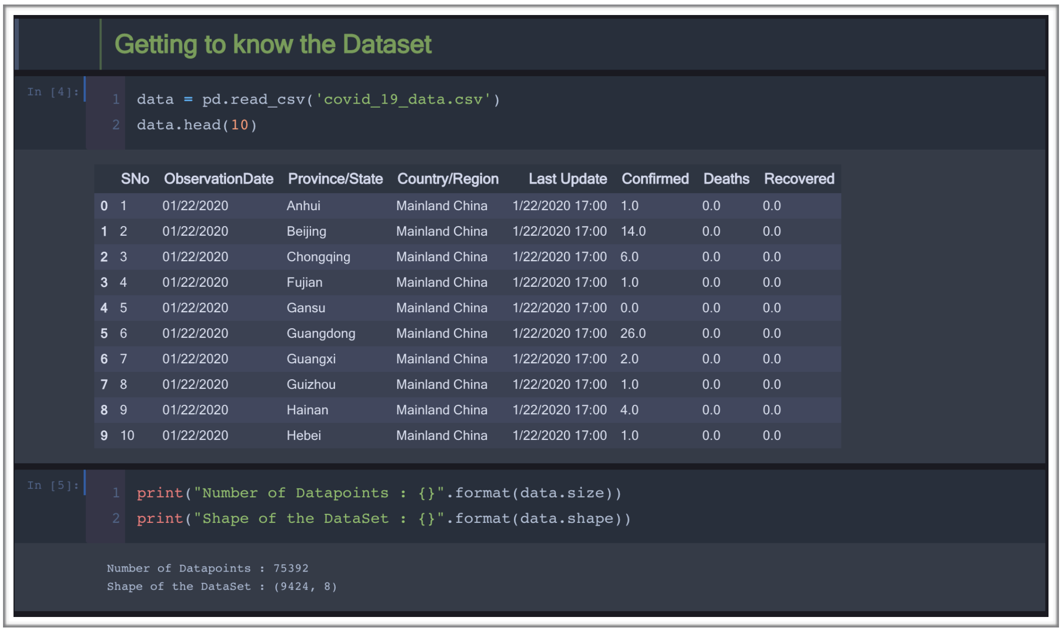
```
10 rows × 70 columns
```



```
In [49]:  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
```



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

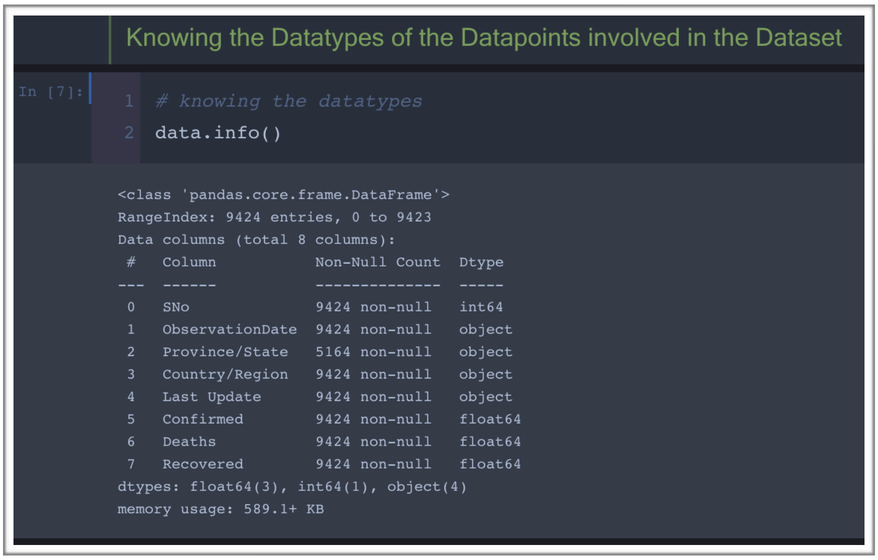
In [5]:
```
print("Number of Datapoints : {}".format(data.size))
print("Shape of the DataSet : {}".format(data.shape))
```

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

Fig-1.3 Reading a Dataset

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

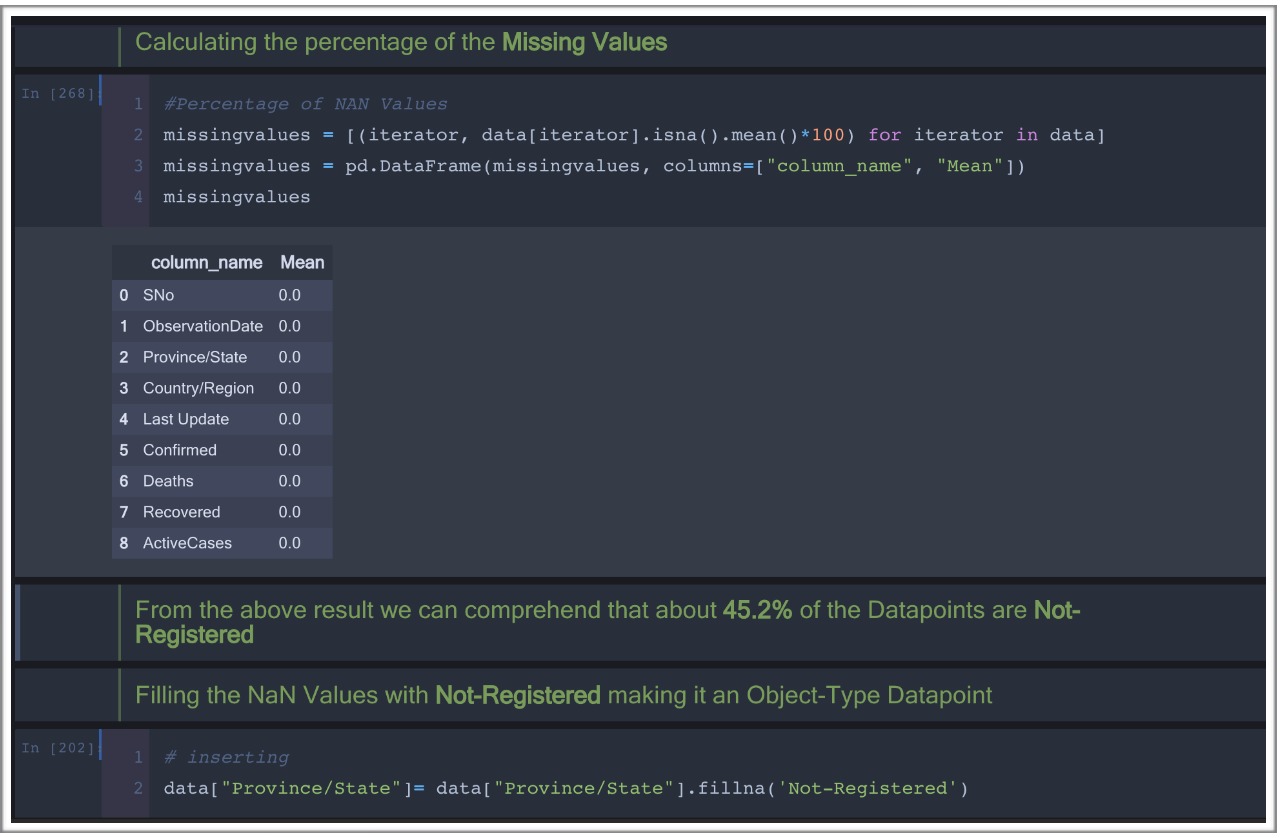
In [7]:
```
# knowing the datatypes
data.info()
```

```
<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
```

Fig-1.4 The *info()* Method.

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.**



Calculating the percentage of the **Missing Values**

In [268]

```
#Percentage of NAN Values
missingvalues = [(iterator, data[iterator].isna().mean()*100) for iterator in data]
missingvalues = pd.DataFrame(missingvalues, columns=["column_name", "Mean"])
missingvalues
```

	column_name	Mean
0	SNo	0.0
1	ObservationDate	0.0
2	Province/State	0.0
3	Country/Region	0.0
4	Last Update	0.0
5	Confirmed	0.0
6	Deaths	0.0
7	Recovered	0.0
8	ActiveCases	0.0

From the above result we can comprehend that about **45.2%** of the Datapoints are **Not-Registered**

Filling the NaN Values with **Not-Registered** making it an Object-Type Datapoint

In [202]

```
# inserting
data["Province/State"]= data["Province/State"].fillna('Not-Registered')
```

Fig-1.5 Data Preprocessing.

Finding the Mean or count of the missing values, enable us to layout and anticipate for the future consequences while building a model, machine learning algorithms need a lot of data points to create a predictive model. In the figure *1.5 we can observe that the column/feature 'Country/Region' has missing values in about 45.2 average of slots. From here there are two ways to deal with the missing values 1. Is to simply drop the entire row with missing values, 2. Is to take sample mean or median of the columns only if the values are not object type and fill them in the missing positions. In our case I dropped the rows with missing values under 'Province/State'.*

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.

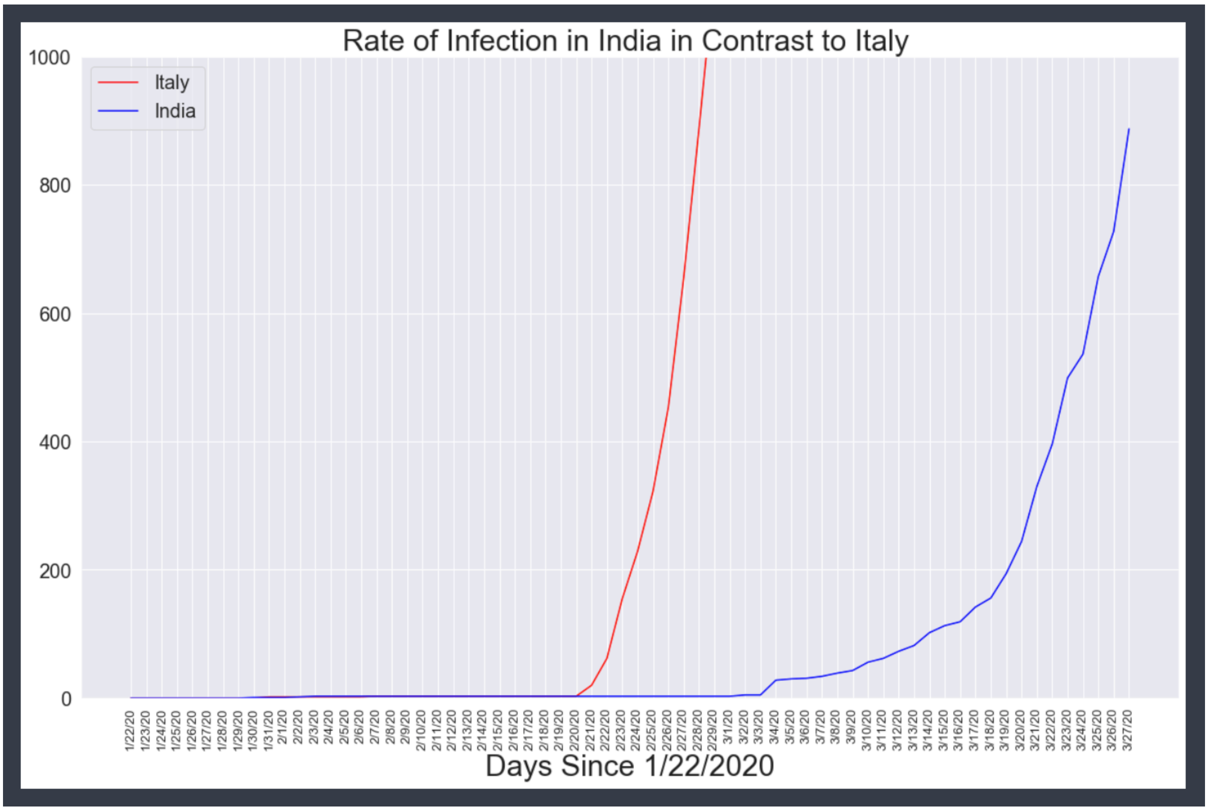


Fig-1.6 Rate of Infection Spread in India in contrast to Italy.

Graphs give us valuable insights instantly just by looking at them and so does this graph (Fig- 1.6). We can clearly observe the growth spiked rapidly in the span of two days in Italy, where as in India infection spread gradually during the dates.

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.

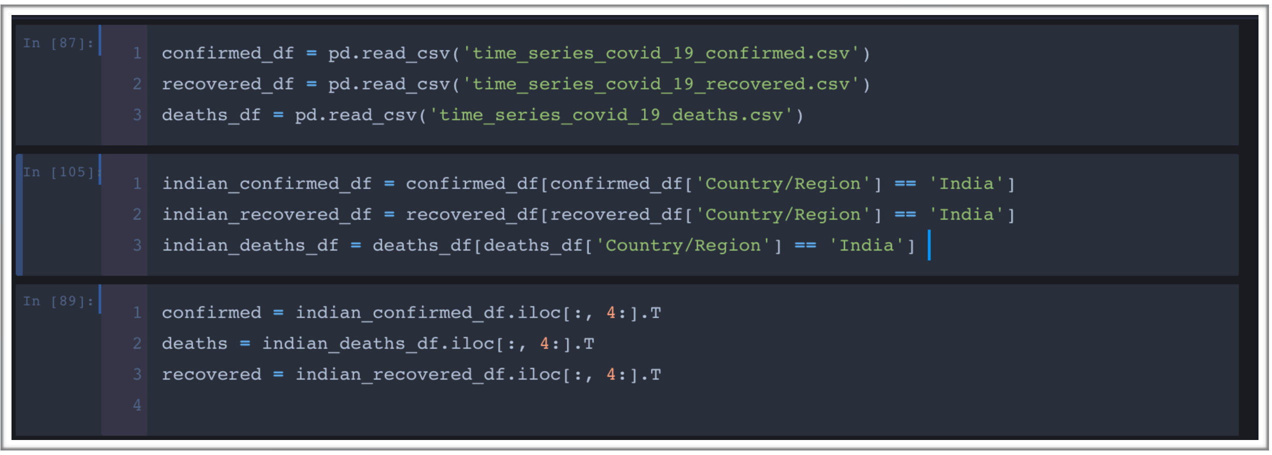
One can employ a particular Machine learning algorithm based on the previous conclusions made on data to classify the problem. A predictive model could either be one of the two kinds,A Supervised learning model, or an Unsupervised learning model.In Our case it is an Unsupervised learning model, since 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.



```
In [87]:
confirmed_df = pd.read_csv('time_series_covid_19_confirmed.csv')
recovered_df = pd.read_csv('time_series_covid_19_recovered.csv')
deaths_df = pd.read_csv('time_series_covid_19_deaths.csv')
```

```
In [105]:
indian_confirmed_df = confirmed_df[confirmed_df['Country/Region'] == 'India']
indian_recovered_df = recovered_df[recovered_df['Country/Region'] == 'India']
indian_deaths_df = deaths_df[deaths_df['Country/Region'] == 'India']
```

```
In [89]:
confirmed = indian_confirmed_df.iloc[:, 4:].T
deaths = indian_deaths_df.iloc[:, 4:].T
recovered = indian_recovered_df.iloc[:, 4:].T

```

Fig- 1.7 Reading the Data using pd.read_csv

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 Fig-1.7.

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.*

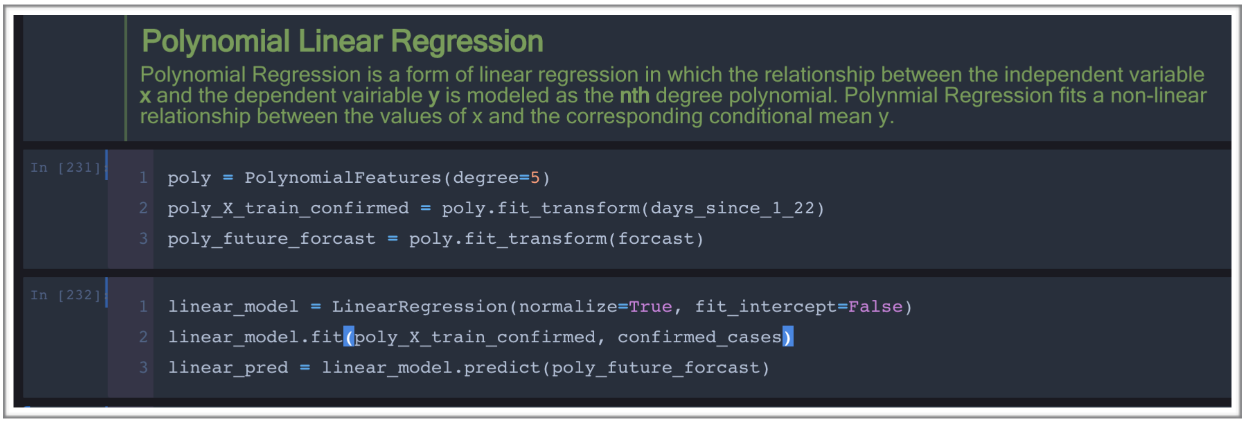
In our scenario out dataset has good amount of data points with minimal loss during the preprocessing. So, it is possible to create a complex model to avoid overfitting.

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)
```

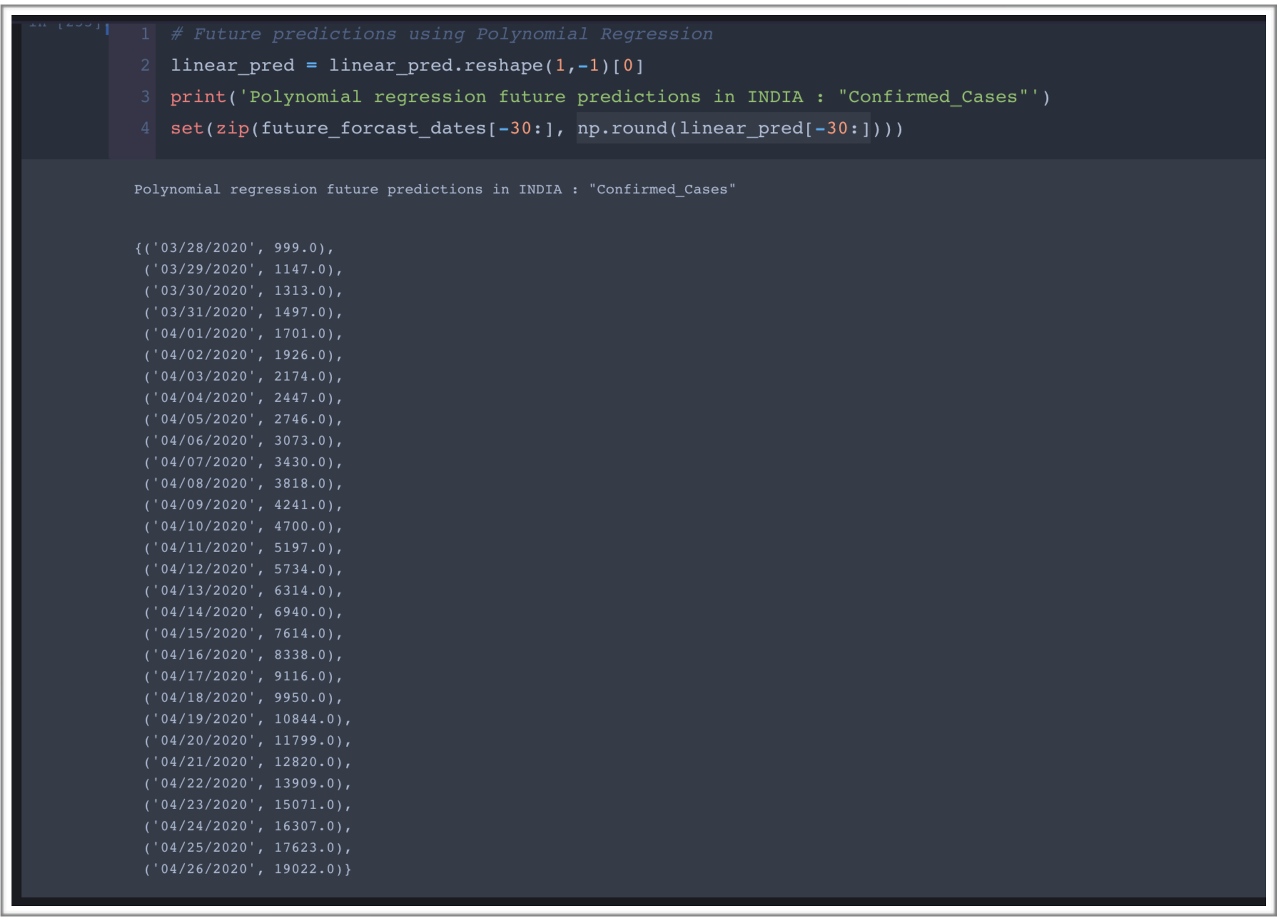
Fig- 1.8 Transforming the Data to Higher order

The transformed linear equation is still considered to be a linear model. Values associated with features are still linear. Curve will be quadratic in nature.

Technically to convert features to the higher order terms we use the Polynomial Features class provided by scikit-learn module. Then we pass the data into a linear model to train it.

After passing the transformed features to the linear model i.e **linear_model.predict()** we pretty much now have the target values. since we passed the data from the first registered date i.e 1/22/2020 to additional 30 days to the last reported date we totally have 96 data points in that features which are transformed through the polynomial features. The predicted target values are situated 30 positions from the last in the **linear_pred numpy array**(refer to Fig-1.9).

The Final Result



```
# Future predictions using Polynomial Regression
linear_pred = linear_pred.reshape(1,-1)[0]
print('Polynomial regression future predictions in INDIA : "Confirmed_Cases"')
set(zip(future_forcast_dates[-30:], np.round(linear_pred[-30:])))
```

```
Polynomial regression future predictions in INDIA : "Confirmed_Cases"


{('03/28/2020', 999.0),
 ('03/29/2020', 1147.0),
 ('03/30/2020', 1313.0),
 ('03/31/2020', 1497.0),
 ('04/01/2020', 1701.0),
 ('04/02/2020', 1926.0),
 ('04/03/2020', 2174.0),
 ('04/04/2020', 2447.0),
 ('04/05/2020', 2746.0),
 ('04/06/2020', 3073.0),
 ('04/07/2020', 3430.0),
 ('04/08/2020', 3818.0),
 ('04/09/2020', 4241.0),
 ('04/10/2020', 4700.0),
 ('04/11/2020', 5197.0),
 ('04/12/2020', 5734.0),
 ('04/13/2020', 6314.0),
 ('04/14/2020', 6940.0),
 ('04/15/2020', 7614.0),
 ('04/16/2020', 8338.0),
 ('04/17/2020', 9116.0),
 ('04/18/2020', 9950.0),
 ('04/19/2020', 10844.0),
 ('04/20/2020', 11799.0),
 ('04/21/2020', 12820.0),
 ('04/22/2020', 13909.0),
 ('04/23/2020', 15071.0),
 ('04/24/2020', 16307.0),
 ('04/25/2020', 17623.0),
 ('04/26/2020', 19022.0)}
```

Fig- 1.9 Prediction Result

Formatting the output

Zipping the predicted values and the 30 dates from the last reported date will organise the output as shown in Fig-1.9.

Points worth noting

According to the Manual verification even though this result is completely Generalised, the result yielded by the model is closely accurate to the values in reality, This is an intriguing point to consider because Without considering the ***Cultural Context*** of a country which includes the personality and odd situations such as disobeying QUARANTINE by even a single person has the potential to trigger an **exponential Spread** of this infection

Note ~ Over the period of time if the resultant numerical’s are varying to reality, then it could be a possibility that an event such as an unofficial gathering consisting an infected could have already happened in between dates where a significant difference is observed
This event is the catalyst to this exponential increase in the numbers. In order to use the prediction algorithm again with new data it is essential that it should have more data points from the date where the difference was observed. With out significant amount of data points it’s difficult to analyse the pattern.

Conclusion

Statistical analysis made in this report gives the perspective of the condition where without self isolating, the possibility of a community spread of infection is off the charts. Italy as mentioned above, is the major example and itself is a representation of a worst case scenario. 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 severe could be the situation, if a proper protocol is not employed.I believe Self Isolation is the only key factor for any of the above numerical value to variate positively.

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/

- x - x -x - x - x -x - x - x -x -x -