**A**

**Minor Project Report on**

**“Prediction of the growth of CORONA VIRUS”**

In partial fulfillment of requirements for the degree of

**Bachelor of Technology (B. Tech.)**

in

**Computer Science and Engineering**

**Submitted by**

Ms. Anjali Sharma (170279)

**Under the Guidance of**

Dr. Sunil Kumar Jangir

**SCHOOL OF ENGINEERING AND TECHNOLOGY**

**Mody University and Science and Technology Lakshmangarh, Distt. Sikar-332311**

December 2020

**A C K N O W L E D G E M E N T**

I sincerely express my gratitude to my guide for his benevolent guidance in completing the report on **Prediction of the growth of CORONA VIRUS.** His kindness and help have been the source of encouragement for me.

I am grateful to his for the guidance, inspiration and constructive suggestions that helpful us in the preparation of this project.

**Anjali Sharma-170279**

**CERTIFICATE**

This is to certify that the minor project report entitled “Prediction of the growth of CORONA VIRUS” submitted by Ms. Anjali Sharma, as a partial fulfillment for the requirement of B. Tech. VII Semester examination of the School of Engineering and Technology, Mody University of Science and Technology, Lakshmangarh for the academic session 2019-2020 is an original project work carried out under the supervision and guidance of Dr. Sunil Kumar Jangir has undergone the requisite duration as prescribed by the institution for the project work.

**PROJECT GUIDE: HEAD OF DEPARTMENT**

**Approval Code: AUT\_20\_CSE\_F12\_03 Signature:**

**Name: Dr. Sunil Kumar Jangir Name: A.Senthil**

**Date: 27-12-2020 Date:27-12-2020**

**EXAMINER-I: EXAMINER-II**

**Name: Ajay Kumar Name: Praneet Suarabh**

**Dept: SET Dept:SET**

**ABSTRACT**

The upsurge of this disease that is CORONA VIRUS has created life and death situation in the world of living. Virus is increasing day by day and effective lives. Machine Learning can be established very effectively in tracing the disease predict the growth and form an effective strategy in order to manage the effect of the virus. This report gives us full glance and the best mathematical computation with modeling for predicting the growth.

An ML based project we come with various computation and modeling to suspect or predict the growth on a particular dataset. Although this concept can be used on dynamic dataset which are changing day to day, but here in this report we will study on a particular dataset.

Working on the dataset led to various challenges such as modeling on different algorithms of machine learning but finally worked on them in order to get the best result. This report is an insight of the working brief of the project such as descriptive information about machine learning, algorithms, statistical description and most important the programming language used here which is python.

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Sr.no.** | **Topics** | **Page no.** |
| **1.** | **Introduction** | 1 |
| 1.1 | *Present System* | 1 |
| 1.2 | *Proposed System* | 1 |
| **2.** | **System Design** | **2** |
| 2.1 | *-System flowchart \** | 2 |
| 2.2 | *-Data tables (database if taken)/ Decision Tree/ Data Dictionary \** | 3 |
| **3.** | **Hardware and Software details** | **4** |
|  | * *Mention various H/w S/w tools used with their standards e.g., HTML 5, Net Beans 8.0.2 etc.* | 5 |
| **4.** | **Implementation Work Details** | **6** |
| 4.1 | *-Real life applications* | 6 |
| 4.2 | *-Data implementation and program execution* | 6 |
| **5.** | **Source Code/ Simulation Code/ Commands etc.** | **7-10** |
| **6.** | **Input/output Screens/ Model’s Photograph** | **11-12-13** |
| **7.** | **System Testing** |  |
| **8.** | **Individual Contribution**  *(Separate paragraph for each group member)* |  |
| **9.** | **Conclusion** | **15** |
| 9.1 | *-Limitations* |  |
| 9.2 | *-Scope for future work* |  |
| **10.** | **Bibliography**  *(Strictly in IEEE style only)* | 16 |
| **11.** | **Annexures**   * *Publication (if any) Communicated or Published* * *Plagiarism Report (Maximum 10-15% excluding code)*   *(Report to be checked by Mentor from his/ her Urkund Id)*   * Anything else *(suggested by mentor)* | 20 |

\*Minor changes can be done at departmental level

**Chapter1: Introduction**

1. **INTRODUCTION**

This deadly disease is caused by the spread of various germs harmful bacteria(pathogens) which transmits from one human to many humans, from one animal to many, and from animal to human. Early diagnoses are curable, while the patients suffering from it with maximum number of days are not 100% curable.

There is need of innovation on predicting the growth with deep through analysis, on huge global data on the rise of the virus.

The project comprises of two main features or methods we can say, first predicting and analyzing with cumulative confirmed cases and then representing with visuals that is data visualization. Second one is predicting the growth on total, confirmed, new cases and finding accuracy.

* 1. **PRESENT SYSTEM**

Many employers are working on same data, and with same idea on predicting the growth of virus as by analyzing cases. The COVID crises has led many colleges and students work in team to get into a solution against corona.

There are many ongoing researches and many projects has already been developed in predicting creating an awareness on the same

* 1. **PROPOSED SYSTEM**

Working on the dataset led to various challenges such as modeling on different algorithms of machine learning but finally worked on them in order to get the best result. It is an insight of the working brief of the project such as descriptive information about machine learning, algorithms, statistical description and most important the programming language used here which is python.

**Chapter2: System Design**

1. **System Design**

**2.1 System Flow Chart**

Problem

Importing Dataset

Data Pre-Processing

Data Visualization

Algorithm Selection

Training

Testing

Accuracy /Output

OK?

**2.1 Data Dictionary**

**Data Pre-Processing**: Our dataset needs to be pre-processed. Therefore, data pre-processing is required in this project.

**Definition of Training Set**: The training set is the data that the algorithm will learn from. Learning looks different depending on which algorithm you are using.

**Algorithm Selection:** Our project has been implemented using various algorithms such as linear regression, random forest, decision tree.

**Decision Tree:** In python we use decision tree to observe and figure out the trained data in the structure of tree in order for any future implementation. Decision Tree, here the target variables take continuous values called as regression tree

**Chapter 3: Hardware and Software details**

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda® distribution that enables us to launch applications and easily manage conda packages, environments, and channels without using command-line commands. Navigator can look for packages on Anaconda Cloud or in a local Anaconda Repository. It is accessible for Windows, macOS, and Linux.

**Why to use Navigator?**

In order to run, numerous scientific packages rely on specific versions of other packages. Data scientists often use various versions of many packages and use multiple environments to separate these different versions.

The command-line program conda is both a package manager and an environment manager. This helps data scientists ensure that each version of each package has all the dependencies it requires and works accurately.

Navigator is an easy, point-and-click approach to work with packages and environments without needing to type conda commands in a terminal window. We can utilize it to find the packages we want, install them in an environment, run the packages, and update them – all inside Navigator.

**What applications can we access using Navigator?**

The following applications are available by default in Navigator:

* [JupyterLab](https://jupyterlab.readthedocs.io/en/stable/)
* [Jupyter Notebook](https://jupyter.readthedocs.io/en/latest/)
* [Spyder](https://www.spyder-ide.org/)
* [VSCode](https://code.visualstudio.com/docs)
* [Glueviz](http://glueviz.org/en/stable/)
* [Orange 3 App](http://orange.biolab.si/docs/)
* [RStudio](http://docs.rstudio.com/)

**How can we run code with Navigator?**

The simplest way is with Spyder. From the Navigator Home tab, click Spyder, and write and execute the code.

We can also use Jupyter Notebooks the same way. Jupyter Notebooks are an increasingly popular system that combines the code, descriptive text, output, images, and interactive interfaces into a single notebook file that is edited, viewed, and used in a web browser.

**Using Jupyter Notebook**

The Jupyter Notebook application allows us to create and edit documents that display the input and output of a Python or R language script. Once saved, we can share these files with others.

Python and R language are included by default, but with customization, Notebook can run several other kernel environments.

Basically, it is a combination of an IDE, server to run our projects (called notebooks) either on our local computer or remotely, and has support for approximately 40 computer languages. It originally was for the languages, Julia, Python, and R. The notebooks contain both code and presentation elements, such as images or calculations together in one place. The notebooks are run/ interpreted via kernels, which seem like virtual machines and will use memory of the computer running it. The memory will not be released until exiting the execution of the notebook. We can also use it with Docker containers.

**Chapter 4: Implementation Work Details**

* 1. **Data Implementation and program details**

### Libraries used

### Numpy

### NumPy is the fundamental package for scientific computing with Python. It contains among other things:

* a powerful N-dimensional array object

### broadcasting Functions

### Tools for integrating

### Useful linear algebra etc.

### 

### Pandas

### Pandas is an open source, BSD-authorized library giving superior, simple to-utilize information structures and information investigation apparatuses for the Python programming language.

* Benefits:

Python has for some time been incredible for information munging and planning, however less so for information examination and displaying. pandas helps fill this hole, empowering you to do your whole information examination work process in Python without changing to a more space explicit language like R.

Joined with the amazing IPython toolbox and different libraries, the earth for doing information examination in Python exceeds expectations in execution, profitability, and the capacity to work together.

More work is as yet expected to make Python a top notch measurable displaying condition.

**Chapter 5:Source Code**

#Importing necessary libraries

import numpy as np

import pandas as pd

#Importing the dataset

dipsi=pd.read\_csv('owid-covid-data.csv')

dipsi.head(6)

#To see all features together

dipsi.columns.tolist()

#For determining no. of rows and columns

dipsi.shape

#Checking whether column is categorical column or numerical column

dipsi.dtypes

#To check if there are any Nan values

dipsi.isnull().sum()

#To subset only those rows that have “India” in the “location” column

dipsi = dipsi[dipsi.location == 'India']

dipsi.head()

#Converting date coloumn to ordinal coloumn

import datetime as dt

dipsi["date"]=pd.to\_datetime(dipsi["date"])

dipsi["date"]=dipsi["date"].map(dt.datetime.toordinal)

dipsi.head()

#If more than 50% values in a column are null, then drop that entire column

dipsi = dipsi.drop(dipsi.columns[dipsi.apply(lambda col : col.isnull().sum()/len(dipsi) > 0.5)],axis=1)

import matplotlib.pyplot as plt

%matplotlib inline

import seaborn as sns

#Dropping categorical columns in data frame bd

mody = dipsi.drop(['iso\_code','continent','location','tests\_units'],axis=1)

#Loop for plotting histograms

for i in mody.columns :

plt.hist(mody[i],bins=10,color='r')

plt.title("histogram")

plt.xlabel(i)

plt.ylabel("Frequency")

plt.show()

#Finding mean of each column

dipsi.mean(numeric\_only=True)

#Finding median of each column

dipsi.median(numeric\_only=True)

#Finding mode of each column

dipsi.mode(numeric\_only=False)

#Loop for plotting Scatter plots

for k in range(len(mody.columns)) :

for i in range(len(mody.columns)) :

for j in range(1) :

plt.scatter(x=mody.columns[i:i+1],y=mody.columns[j+k:j+k+1],color='r')

plt.show()

#Loop for plotting Line plots

flag=0

col = mody.columns

for k in range(len(mody.columns)) :

if flag==1 :

col = col.drop(col[0])

for i in mody.columns :

for j in col :

sns.lineplot(x=mody[i],y=mody[j],data=mody,color='r')

plt.show()

flag=1

break

#Modeling

from sklearn.linear\_model import LinearRegression

from sklearn.model\_selection import train\_test\_split

from sklearn.ensemble import RandomForestRegressor

from sklearn.metrics import mean\_squared\_error as MSE

# X is our feature coloumn and Y is our target coloumn

features = mody.columns #input values

features = features.drop('total\_cases')

X = mody[features].values

y = mody['total\_cases'].values.reshape(-1,1)

# Splitting the dataset into the Training set and Test set

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.1, random\_state = 42)

print("Shape of X\_train: ",X\_train.shape)

print("Shape of X\_test: ", X\_test.shape)

print("Shape of y\_train: ",y\_train.shape)

print("Shape of y\_test: ",y\_test.shape)

# Create the Regressor LinReg

LinReg = LinearRegression()

# Fit the regressor to the training data

LinReg.fit(X\_train,y\_train.ravel())

# Predict on the test data: y\_pred

y\_pred = LinReg.predict(X\_test)

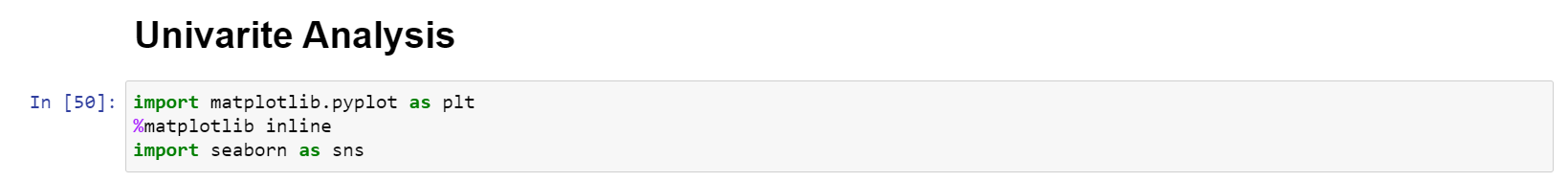
# Compute the accuracy

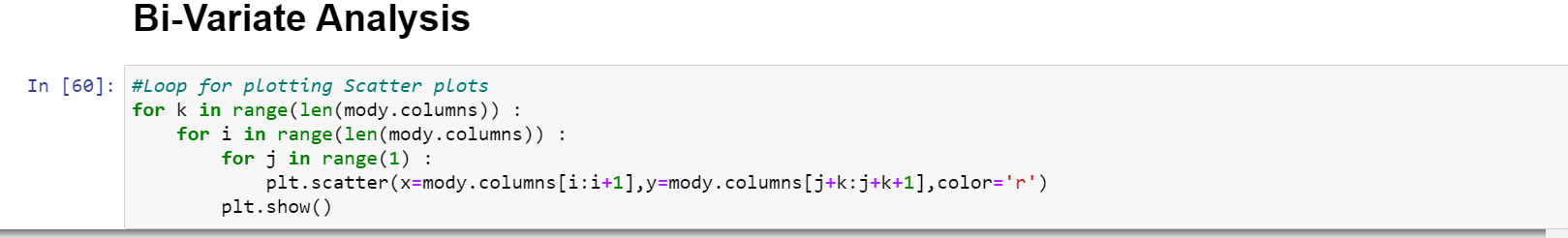
LinReg.score(X\_test,y\_test)

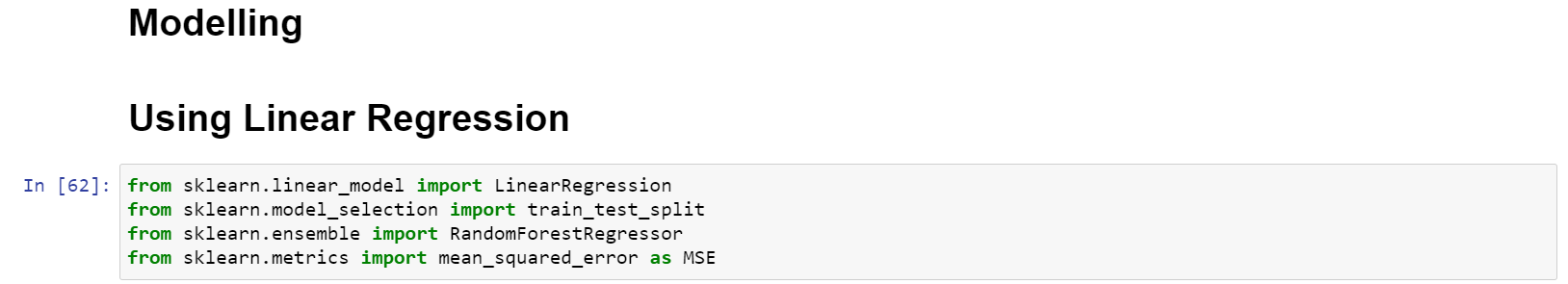
**Chapter 6: Output Screenshots**

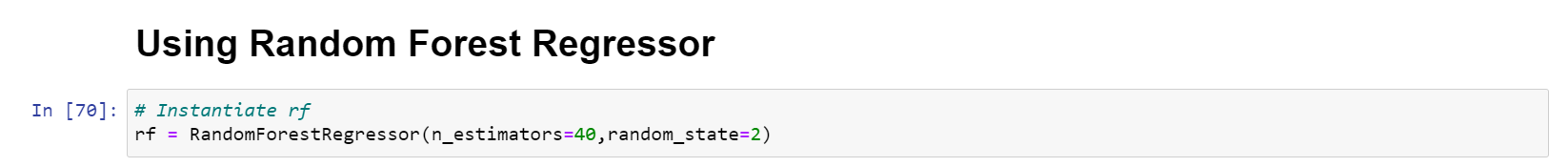
1. **Screenshots of the source code**

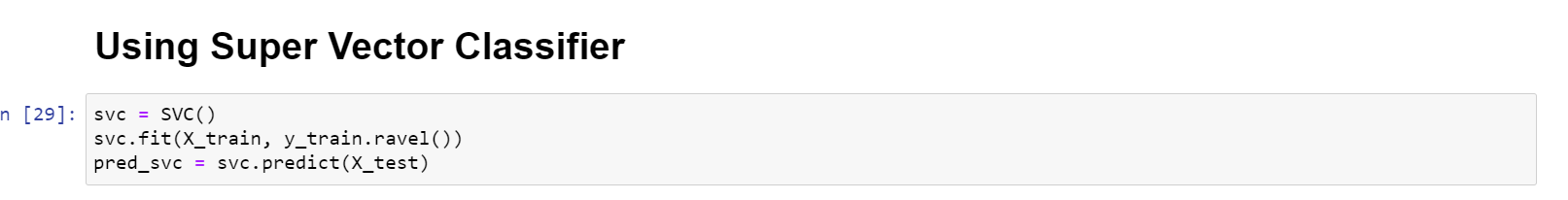


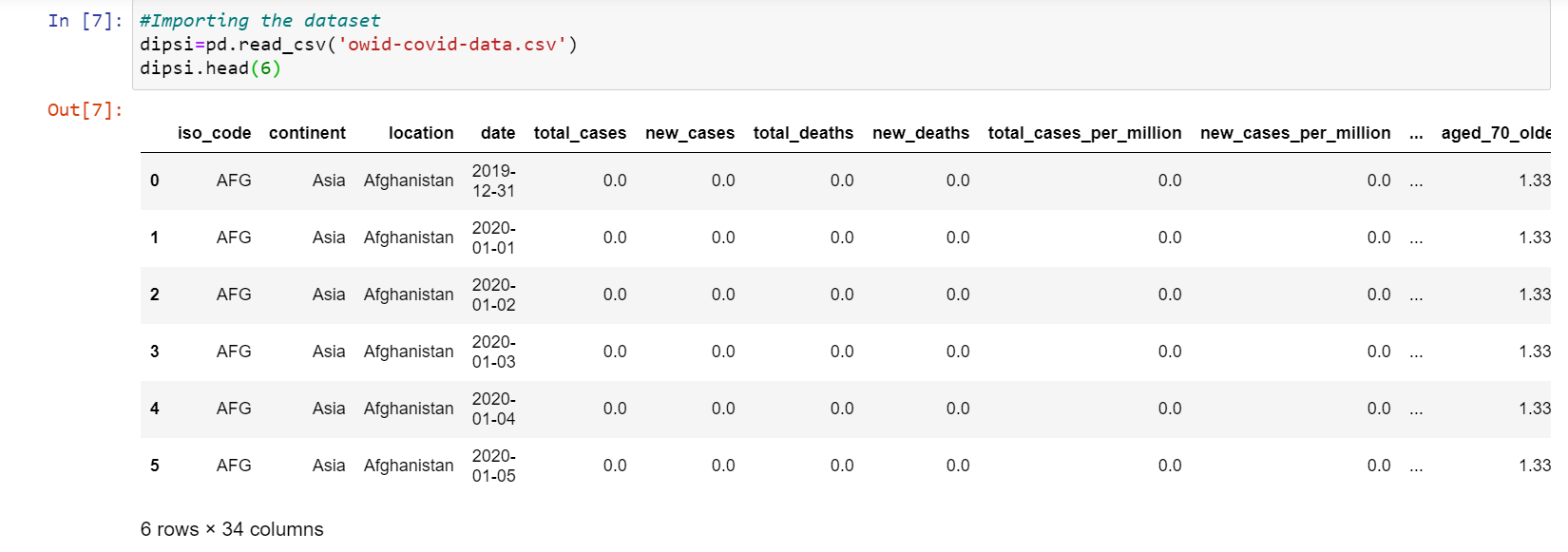


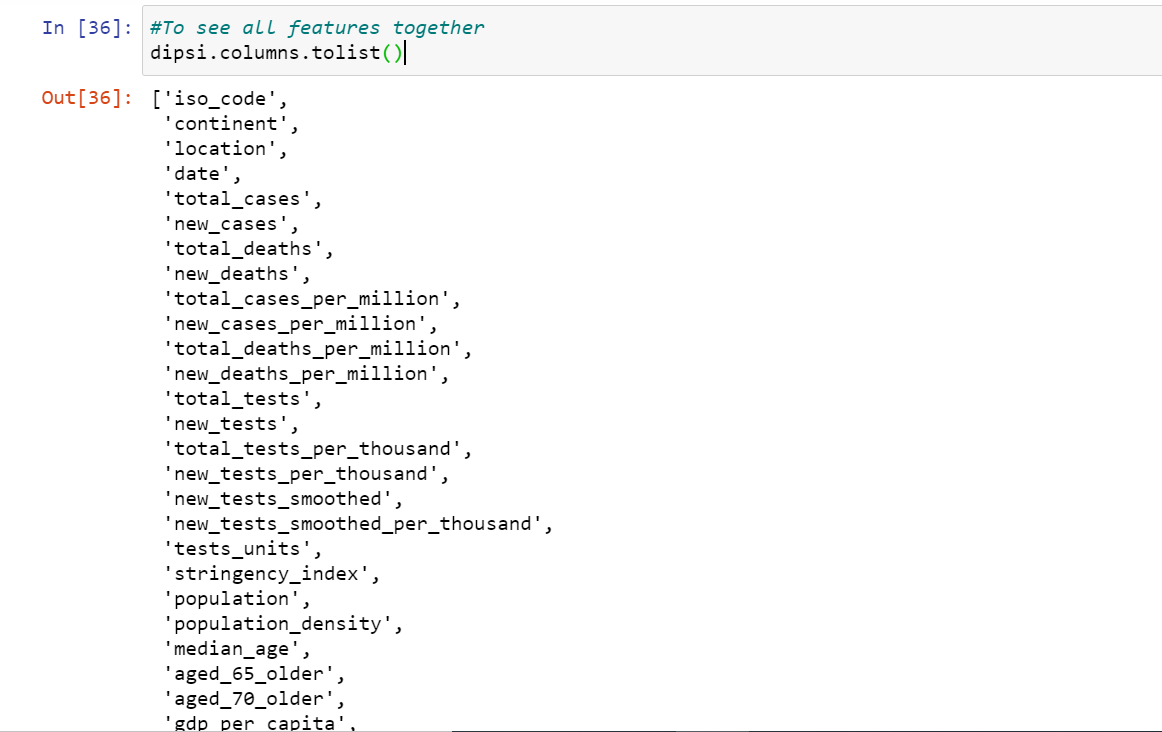


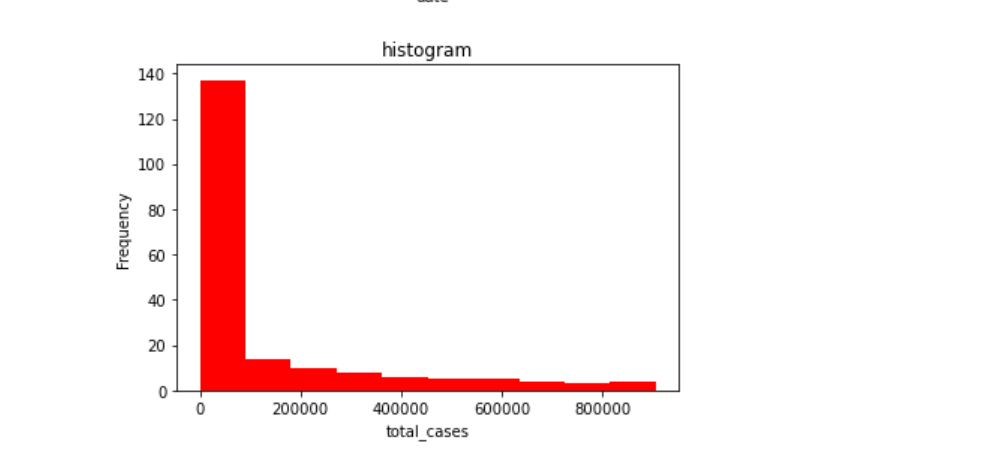


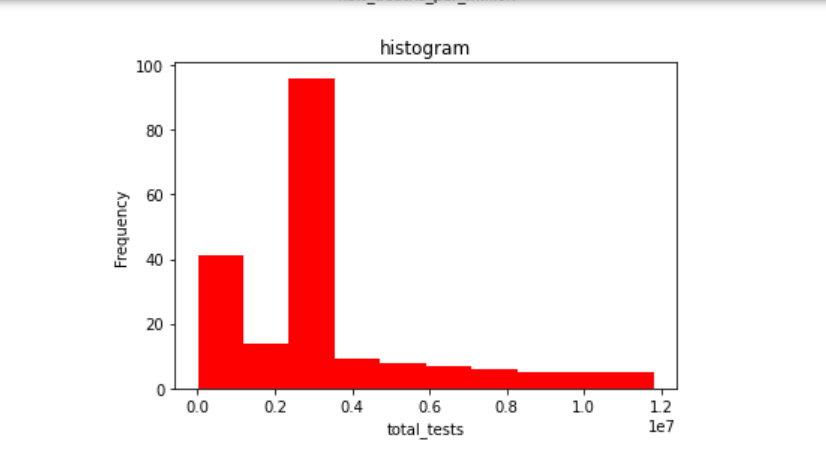


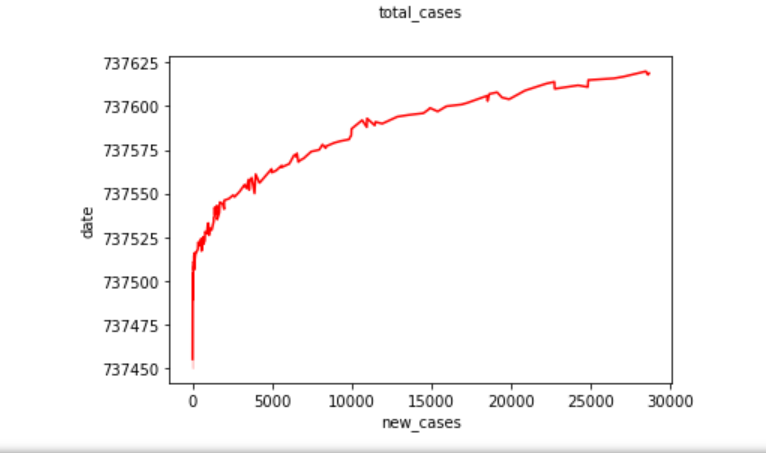












**Chapter 9: Conclusion**

It is estimated that COVID-19 will be over probably in late-April, 2020 in Wuhan and before late-March, 2020 in other areas respectively.

The cumulative number of confirmed COVID-19 cases is 49852-57447 in Wuhan, 12972-13405 in non-Hubei areas and 80261-85140 in China mainland. According to the current trend, the cumulative death toll predicted by the three models are: 2502-5108 in Wuhan, 107-125 in non-Hubei areas, and 3150-6286 in China mainland.

According to the fitting analysis of the existing data by the three mathematical models, the inflection points of the COVID-19 epidemic in Wuhan, non-Hubei areas and China mainland is basically in the middle of February 2020. The prediction results of three different mathematical models are different for different parameters and in different regions.

In general, the fitting effect of Logistic model may be the best among the three models studied in this paper, while the fitting effect of Gompertz model may be better than Bertalanffy model.

**Chapter 10: Bibliography**

1. (n.d.). *State Health Commission of the People's Republic of China.*
2. (Rong Z. Design and validation of Ebola epidemic disease prediction model and prevention and cure)