COVID 19 DATA ANALYSIS

A PROJECT REPORT

Submitted by

Angelin J G [Reg No: RA2112704010009]

Under the Guidance of

Dr. Kalpana A V
(Assistant Professor, Department of Data Science and Business Systems)

In partial fulfillment of the Requirements for the Degree of

MASTERS OF TECHNOLOGY
(INTEGRATED)
COMPUTER SCIENCE AND BUSINESS SYSTEMS

DEPARTMENT OF DATA SCIENCE AND BUSINESS SYSTEMS

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

NOVEMBER 2022

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this project report titled **"COVID 19 DATA ANALYSIS"** is the bonafide work of **"Angelin J G [Reg No: RA2112704010009]** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

Dr. Kalpana A V
GUIDE
Assistant Professor
Dept. of DSBS

Dr.M.Lakshmi
HEAD OFTHEDEPARTMENT
Dept. of DSBS

Signature of Internal Examiner

Signature of External Examiner

ABSTRACT

Corona Virus Disease- 19 (COVID-19) was first time reported in Wuhan, China. This disease has covered more than 200 countries till May 2020. World Health Organisation (WHO) has declared COVID-19 as Public Health Emergency of International Concern (PHEIC) on 30 January 2020. COVID- 19 causes severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) which was progressive earlier in China but now in maximum countries. Therefore, the different online platform are used which provides the latest update of confirmed corona cases throughout the globe for the analysis of data. The aim of data analysis for COVID-19 is to aware of the community against the infectious disease and forecast the COVID-19 confirmed cases, deaths, and recoveries through the data analysis methods. Different models are also used to study the behavior of the disease.

The models help to the patterns of comparison of data between different states and countries. In this project,We will be using cluster,SVM,Linear regression models to understand our data.

ACKNOWLEDGEMENTS

We express our humble gratitude to **Dr C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology, **Dr T.V.Gopal**, for his invaluable support.

We wish to thank **Dr Revathi Venkataraman**, Professor & Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We are incredibly grateful to our Head of the Department**, Dr M. Lakshmi** Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We want to convey our thanks to our program coordinators **Dr.G.Vadivu,** Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for her inputs during the project reviews and support.

We register our immeasurable thanks to our Faculty Advisor**, Dr.K.Shantha Kumari,** Assistant Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to my guide, **Dr. Kalpana A V,** Assistant Professor, Department of Data Science and Business Systems, for providing me with an opportunity to pursue my project under his mentorship. He provided us with the freedom and support to explore the research topics of our interest. His passion for solving problems and making a difference in the world has always been inspiring.

We sincerely thank the Data Science and Business Systems staff and students, SRM Institute of Science and Technology, for their help during our project. Finally, we would like to thank

parents, family members, and friends for their unconditional love, constant support, and encouragement.

Angelin J G

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

SVM Support Vector Machine

COVID coronavirus disease

SARS Severe Acute Respiratory Syndrome

LIST OF SYMBOLS

$\wedge$ Conjunction

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Coronavirus disease 2019 (COVID-19) is an infectious disease caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). It was first identified in December 2019 in Wuhan, China, and has resulted in an ongoing pandemic. The first case may be traced back to 17 November 2019. As of 8 June 2020, more than 6.98 million cases have been reported across 188 countries and territories, resulting in more than 401,000 deaths. More than 3.13 million people have recovered.

The virus is primarily spread between people during close contact, most often via small droplets produced by coughing, sneezing, and talking. The droplets usually fall to the ground or onto surfaces rather than travelling through air over long distances. Less commonly, people may become infected by touching a contaminated surface and then touching their face. It is most contagious during the first three days after the onset of symptoms, although spread is possible before symptoms appear, and from people who do not show symptoms The virus is primarily spread between people during close contact, most often via small droplets produced by coughing, sneezing, and talking. The droplets usually fall to the ground or onto surfaces rather than travelling through air over long distances. Less commonly, people may become infected by touching a contaminated surface and then touching their face.

It is most contagious during the first three days after the onset of symptoms, although spread is possible before symptoms appear, and from people who do not show symptoms.

PANDEMIC :

The COVID-19 pandemic, also known as the coronavirus pandemic, is an ongoing pandemic of coronavirus disease 2019 (COVID-19), caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2).

The outbreak was first identified in Wuhan, China, in December 2019. The World Health Organization declared the outbreak a Public Health Emergency of International Concern on 30 January, and a pandemic on 11 March. A global coordinated effort is needed to stop the further spread of the virus. A pandemic is defined as “occurring over a wide

geographic area and affecting an exceptionally high proportion of the population." The last pandemic reported in the world was the H1N1 flu pandemic in 2009.

Coronaviruses are important human and animal pathogens. At the end of 2019, a novel coronavirus was identified as the cause of a cluster of pneumonia cases in Wuhan, a city in the Hubei Province of China. It rapidly spread, resulting in an epidemic throughout China, followed by an increasing number of cases in other countries throughout the world.
On 30th January 2020 India recorded its first COVID-19 case in state of Kerala. It was a student who had travel history to china. And till the start of June India has over 200 thousand confirmed cases.

1.2 PROBLEM STATEMENT

In this project we dived deep into 'What does data say about Covid-19 situation in India and Comparision of data between countries ?'.
And with available data we came up with some observations and conclusions.
This analysis mainly focuses on:

✓Comparison of global data.

✓State-wise comparison.

CHAPTER 2

LITERATURE REVIEW

In this section, we briefly review the related work on credit card fraud system and their different techniques.

In [1] The outbreak of the Covid-19 pandemic is an unprecedented shock to the Indian economy. The economy was already in a parlous state before Covid-19 struck. With the prolonged country-wide lockdown, global economic downturn and associated disruption of demand and supply chains, the economy is likely to face a protracted period of slowdown. The magnitude of the economic impact will depend upon the duration and severity of the health crisis, the duration of the lockdown and the manner in which the situation unfolds once the lockdown is lifted. In this paper we describe the state of the Indian economy in the pre-Covid-19 period, assess the potential impact of the shock on various segments of the economy, analyse the policies that have been announced so far by the central government and the Reserve Bank of India to ameliorate the economic shock and put forward a set of policy recommendations for specific sectors.

In [2] The aim of data analysis for COVID-19 is to aware of the community against the infectious disease and forecast the COVID-19 confirmed cases, deaths, and recoveries through the data analysis methods. Different models are also used tostudy the behavior of the disease. The models help to forecast the patterns of public sentiments on health information with both the political and economical influence of the spread of the virus. Data analysis methods which are used are Exploratory Data Analysis (EDA) in which the number of confirmed cases, death, and recovered data are recorded, model like Susceptible Exposed Infectious-Recovered (SEIR) model is used to predict the time and the rate taken for the spreading up of disease throughout the globe. A statistical model can also be used to compare the data among different countries to make humans aware of the infection.

In [3] The first incident of COVID-19 case in India was recorded on 30th January, 2020 which turns to 100,000 marks on May 19th and by June 3rd it was over 200,000 active cases and 5,800 deaths. Geographic Information System (GIS) based spatial models can be helpful for better understanding of different factors that have triggered COVID-19 spread at district level in India. In the present study, 19 variables were considered that can explain the

variability of the disease. Different spatial statistical techniques were used to describe the spatial distribution of COVID-19 and identify significant clusters. Spatial lag and error models (SLM and SEM) were employed to examine spatial dependency, geographical weighted regression (GWR) and multi-scale GWR (MGWR) were employed to examine at local level. The results show that the global models perform poorly in explaining the factors for COVID-19 incidences. MGWR shows the best-fit-model to explain the variables affecting COVID-19 (R2= 0.75) with lowest AICc value. Population density, urbanization and bank facility were found to be most susceptible for COVID-19 cases. These indicate the necessity of effective policies related to social distancing, low mobility. Mapping of different significant variables using MGWR can provide significant insights for policy makers for taking necessary actions model.

In [4] Since December 2019 the world is experiencing a deadly disease caused by a novel coronavirus termed as severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). The disease associated with this virus is known as COVID-19. This paper focuses on COVID-19 based on freely available datasets including the ones in Kaggle repository. Data analytics is provided on a number of aspects of COVID-19 including the symptoms of this disease, the difference of COVID-19 with other diseases caused by severe acute respiratory syndrome (SARS), Middle East respiratory syndrome (MERS), and swine flu. The impact of temperature on the spread of COVID-19 is also discussed based on the datasets. Moreover, data visualization is provided on the comparison of infections in males/females which shows that males are more prone to this disease and the older people are more at risk. Based on the data, the pattern in the increase of confirmed cases is found to be an exponential curve in nature. Finally, the relative number of confirmed, recovered and death cases in different countries are shown with data visualization.

In[5] In the current era of big data, a huge amount of data has been generated and collected from a wide variety of rich data sources. Embedded in these big data are useful information and valuable knowledge. An example is healthcare and epidemiological data such as data related to patients who suffered from epidemic diseases like the coronavirus disease 2019 (COVID-19). Knowledge discovered from these epidemiological data helps researchers, epidemiologists and policy makers to get a better understanding of the disease, which may inspire them to come up ways to detect, control and combat the disease. As “a picture is worth a thousand words”, having methods to visualize and visually analyze these big data makes it easily to comprehend the data and the discovered knowledge. In this paper, we

present a big data visualization and visual analytics tool for visualizing and analyzing COVID-19 epidemiological data. The tool helps users to get a better understanding of information about the confirmed cases of COVID-19. Although this tool is designed for visualization and visual analytics of epidemiological data, it is applicable to visualization and visual analytics of big data from many other real-life applications and services.

CHAPTER-3

OBJECTIVES

The primary objective of our project is:

 To study timely trend of Covid-19 in our home districts along with the comparative study of situations, The secondary objectives of our project are:

 To determine the possible analytical outcomes of COVID-19 situations based on different age groups and other divisions.

 To study and evaluate the provided facilities, works done to minimize the effect and spreading of virus from a local level.

 To develop the idea and concept of analytical skills to us.

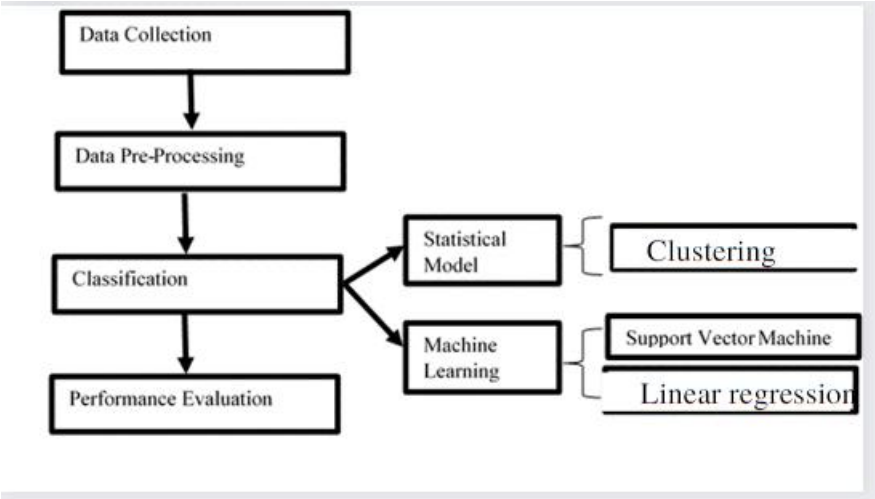
CHAPTER 4

BLOCK DIAGRAM

A cross-sectional quantitative study was conducted to build up a reliable trusted model to forecast COVID-19 diagnosis from the signs and symptoms that participants had.The signs and symptoms of novel COVID-19 used in this study were obtained from world and India level.

Our strategy includes four processing stages, namely data collection, data preprocessing,classification, and performance evaluation. The classification stage can be accomplished either by building statistical modelor by invoking machine learning model. In the statistical model we used Logistic Regression (LR), while in machine learning model we invoked Support Vector Machine (SVM), and Mulit-Layer Perceptron (MLP). Finally the performance of each classifier is quantified.Ablock diagram of the proposed

work is illustrated below



CHAPTER 5

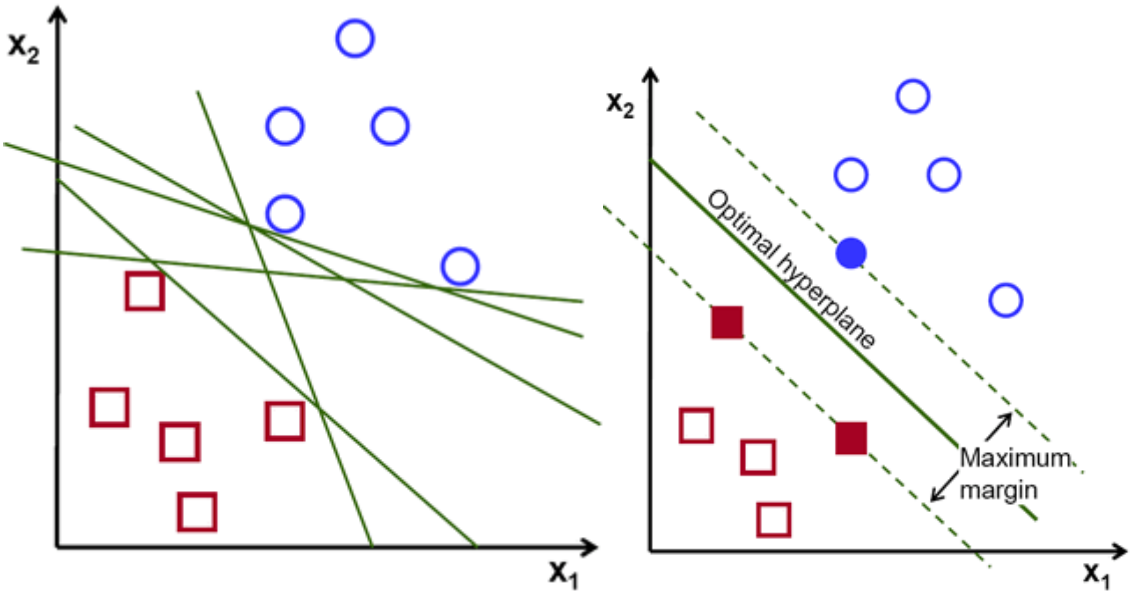
WORKING MODEL

COVID 19 DATA ANALYSIS TECHNIQUES

5.1 Support Vector Machine

Support vector machine is another simple algorithm that every machine learning expert should have in his/her arsenal. Support vector machine is highly preferred by many as it produces significant accuracy with less computation power. Support Vector Machine, abbreviated as SVM can be used for both regression and classification tasks. But, it is widely used in classification objectives.

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space(N — the number of features) that distinctly classifies the data points.



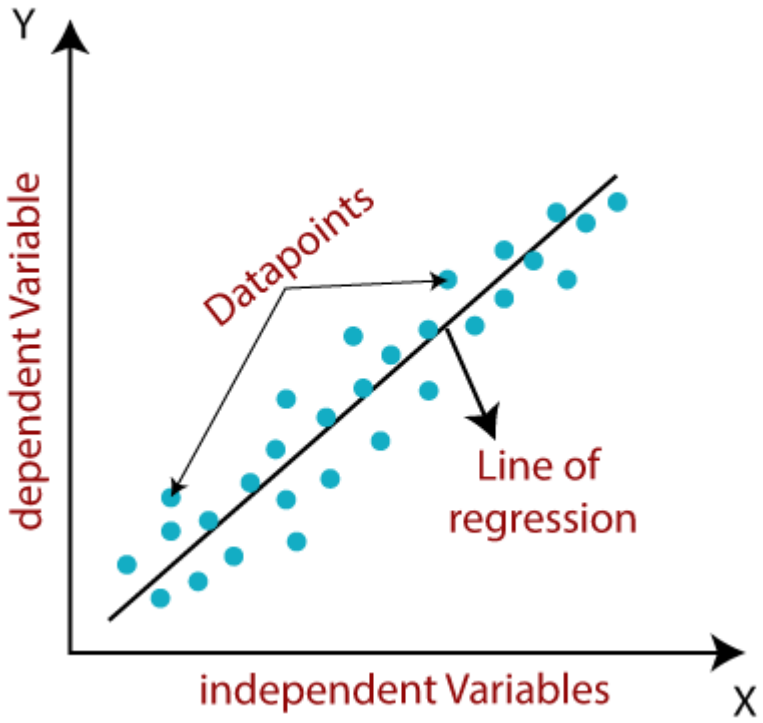
To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

5.2 Linear regression

Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as **sales, salary, age, product price,** etc.

Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.

The linear regression model provides a sloped straight line representing the relationship between the variables. Consider the below image:



Mathematically, we can represent a linear regression as:

$y= a0+a1x+ \varepsilon$

5.3 Clustering

It is basically a type of unsupervised learning method. An unsupervised learning method is a method in which we draw references from datasets consisting of input data without labeled responses. Generally, it is used as a process to find meaningful structure, explanatory underlying processes, generative features, and groupings inherent in a set of examples.

Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity and dissimilarity between them.

CHAPTER 6

PROJECT CODE

6.1 Algorithm

Step 1: Read the dataset.

Step 2: Random Sampling is done on the data set to make it balanced.

Step 3: Divide the dataset into two parts i.e., Train dataset and Test dataset.

Step 4: Feature selection are applied for the proposed models.

Step 5: Accuracy and performance metrics has been calculated to know the efficiency for different algorithms.

Step 6: Then retrieve the best algorithm based on efficiency for the given dataset.

6.2 Program Code

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
from plotly.subplots import make_subplots
from datetime import datetime
from google.colab import files
uploaded = files.upload()
covid_df=pd.read_csv("covid_19_india.csv")
covid_df.head(10)
covid_df.info()
covid_df.describe()
vaccine_df=pd.read_csv("covid_vaccine_statewise.csv")
vaccine_df.head(10)
statewise=pd.pivot_table(covid_df,values=["Confirmed","Deaths","Cured"],index="State/
UnionTerritory",aggfunc=max)
```

```
statewise["Recovery Rate"]=statewise["Cured"]*100/statewise["Confirmed"]
statewise["Mortality Rate"]=statewise["Deaths"]*100/statewise["Confirmed"]
statewise=statewise.sort_values(by="Confirmed",ascending=False)
statewise.style.background_gradient(cmap="cubehelix")
#top 5 active cases states
top_5_active_cases=covid_df.groupby(by='State/UnionTerritory').max()[['Active_Cases','
Date']].sort_values(by=['Active_Cases'],ascending=False).reset_index()
fig=plt.figure(figsize=(16,9))
plt.title("Top 5 states with most active cases in India",size=25)
ax=sns.barplot(data=top_5_active_cases.iloc[:5],y="Active_Cases",x="State/UnionTerrito
ry",linewidth=2,edgecolor='blue')
top_5_active_cases=covid_df.groupby(by='State/UnionTerritory').max()[['Active_Cases','
Date']].sort_values(by=['Active_Cases'],ascending=False).reset_index()
fig=plt.figure(figsize=(16,9))
plt.title("Top 5 states with most active cases in India",size=25)
ax=sns.barplot(data=top_5_active_cases.iloc[:5],y="Active_Cases",x="State/UnionTerrito
ry",linewidth=2,edgecolor='blue')
plt.xlabel("States")
plt.ylabel("Total Active Cases")
plt.show()
top_5_deaths=covid_df.groupby(by='State/UnionTerritory').max()[['Deaths','Date']].sort_v
alues(by=['Deaths'],ascending=False).reset_index()
fig=plt.figure(figsize=(18,5))
plt.title("Top 5 states with most deaths",size=25)
ax=sns.barplot(data=top_5_deaths.iloc[:7],y="Deaths",x="State/UnionTerritory",linewidth
=2,edgecolor='yellow')
plt.xlabel("States")
plt.ylabel("Total Death Cases")
plt.show()
vaccination=vaccine_df.drop(columns=['Sputnik V (Doses Administered)','AEFI','18-44
Years (Doses Administered)','45-60 Years (Doses Administered)','60+ Years (Doses
Administered)'],axis=1)
#Male vs Female vaccination
male= vaccination["Male(Individuals Vaccinated)"].sum()
female= vaccination["Female(Individuals Vaccinated)"].sum()
px.pie(names=["Male", "Female"], values=[male, female], title = "Male and Female
Vaccination")
vaccine=vaccine_df[vaccine_df.State!='India']
vaccine
#most vaccinated state
max_vac=vaccine.groupby('State')['Total'].sum().to_frame('Total')
```

```
max_vac=max_vac.sort_values('Total',ascending=False)[:5]
max_vac
fig= plt.figure(figsize = (10,5))
plt.title("Top 5 Vaccinated States in India" , size = 20)
x= sns.barplot(data = max_vac.iloc[:10],y = max_vac.Total, x = max_vac.index,
linewidth=2, edgecolor='red')
plt.xlabel("States")
plt.ylabel("Vaccination")
plt.show()


# IMPORTING PACKAGES

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.colors as mcolors
import random
import math
import time
from sklearn.model_selection import RandomizedSearchCV, train_test_split
from sklearn.svm import SVR
from sklearn.metrics import mean_squared_error, mean_absolute_error
import datetime
import operator
plt.style.use('seaborn')
%matplotlib inline
confirmed_cases = pd.read_csv("time_series_covid-19_confirmed.csv")
deaths_reported = pd.read_csv("time_series_covid-19_deaths.csv")
recovered_cases = pd.read_csv("time_series_covid-19_recovered.csv")
confirmed_cases.head()
deaths_reported.head()
# Finding the total confirmed cases, death cases and the recovered cases and append them
to an 4 empty lists
# Also, calculate the total mortality rate which is the death_sum/confirmed cases

dates = confirmed.keys()
world_cases = []
total_deaths = []
mortality_rate = []
total_recovered = []
```

```
for i in dates:
    confirmed_sum = confirmed[i].sum()
    death_sum = deaths[i].sum()
    recovered_sum = recoveries[i].sum()
    world_cases.append(confirmed_sum)
    total_deaths.append(death_sum)
    mortality_rate.append(death_sum/confirmed_sum)
    total_recovered.append(recovered_sum)
days_since_1_22 = np.array([i for i in range(len(dates))]).reshape(-1, 1)
world_cases = np.array(world_cases).reshape(-1, 1)
total_deaths = np.array(total_deaths).reshape(-1, 1)
total_recovered = np.array(total_recovered).reshape(-1, 1)
days_in_future = 10
future_forecast = np.array([i for i in range(len(dates)+days_in_future)]).reshape(-1, 1)
adjusted_dates = future_forecast[:-10]
future_forecast
# Convert all the integers into datetime for better visualization

start = '1/22/2020'
start_date = datetime.datetime.strptime(start, '%m/%d/%Y')
future_forcast_dates = []
for i in range(len(future_forcast_dates)):
    future_forcast_dates.append((start_date +
datetime.timedelta(days=i)).strftime('%m/%d/%Y'))
latest_confirmed = confirmed_cases[dates[-1]]
latest_deaths = deaths_reported[dates[-1]]
latest_recoveries = recovered_cases[dates[-1]]
unique_countries =  list(confirmed_cases['Country/Region'].unique())
unique_countries
# The next line of code will basically calculate the total number of confirmed cases by
each country

country_confirmed_cases = []
no_cases = []
for i in unique_countries:
    cases = latest_confirmed[confirmed_cases['Country/Region']==i].sum()
    if cases > 0:
        country_confirmed_cases.append(cases)
    else:
        no_cases.append(i)
```

```
for i in no_cases:
    unique_countries.remove(i)

unique_countries = [k for k, v in sorted(zip(unique_countries, country_confirmed_cases),
key=operator.itemgetter(1), reverse=True)]
for i in range(len(unique_countries)):
    country_confirmed_cases[i] =
latest_confirmed[confirmed_cases['Country/Region']==unique_countries[i]].sum()
print('Confirmed Cases by Countries/Regions:')
for i in range(len(unique_countries)):
    print(f'{unique_countries[i]}: {country_confirmed_cases[i]} cases')
# Find the list of unique provinces

unique_provinces =  list(confirmed_cases['Province/State'].unique())
# those are countries, which are not provinces/states.
outliers = ['United Kingdom', 'Denmark', 'France']
for i in outliers:
    unique_provinces.remove(i)
# Finding the number of confirmed cases per province, state or city

province_confirmed_cases = []
no_cases = []
for i in unique_provinces:
    cases = latest_confirmed[confirmed_cases['Province/State']==i].sum()
    if cases > 0:
        province_confirmed_cases.append(cases)
    else:
        no_cases.append(i)

for i in no_cases:
    unique_provinces.remove(i)
# number of cases per province/state/city

for i in range(len(unique_provinces)):
    print(f'{unique_provinces[i]}: {province_confirmed_cases[i]} cases')
# handling nan values if there is any

nan_indices = []
```

```
for i in range(len(unique_provinces)):
    if type(unique_provinces[i]) == float:
        nan_indices.append(i)

unique_provinces = list(unique_provinces)
province_confirmed_cases = list(province_confirmed_cases)

for i in nan_indices:
    unique_provinces.pop(i)
    province_confirmed_cases.pop(i)
# Plot a bar graph to see the total confirmed cases across different countries

plt.figure(figsize=(32, 32))
plt.barh(unique_countries, country_confirmed_cases)
plt.title('Number of Covid-19 Confirmed Cases in Countries')
plt.xlabel('Number of Covid19 Confirmed Cases')
plt.show()
# Plot a bar graph to see the total confirmed cases between mainland china and outside
mainland china

china_confirmed = latest_confirmed[confirmed_cases['Country/Region']=='China'].sum()
outside_mainland_china_confirmed = np.sum(country_confirmed_cases) -
china_confirmed
plt.figure(figsize=(16, 9))
plt.barh('Mainland China', china_confirmed)
plt.barh('Outside Mainland China', outside_mainland_china_confirmed)
plt.title('Number of Confirmed Coronavirus Cases')
plt.show()
# Print the total cases in mainland china and outside of it

print('Outside Mainland China {} cases:'.format(outside_mainland_china_confirmed))
print('Mainland China: {} cases'.format(china_confirmed))
print('Total: {} cases'.format(china_confirmed+outside_mainland_china_confirmed))
# Only show 10 countries with the most confirmed cases, the rest are grouped into the
category named others

visual_unique_countries = []
visual_confirmed_cases = []
others = np.sum(country_confirmed_cases[10:])
for i in range(len(country_confirmed_cases[:10])):
    visual_unique_countries.append(unique_countries[i])
```

```
    visual_confirmed_cases.append(country_confirmed_cases[i])

visual_unique_countries.append('Others')
visual_confirmed_cases.append(others)
# Visualize the 10 countries

plt.figure(figsize=(32, 18))
plt.barh(visual_unique_countries, visual_confirmed_cases)
plt.title('Number of Covid-19 Confirmed Cases in Countries/Regions', size=20)
plt.show()
# Create a pie chart to see the total confirmed cases in 10 different countries outside China

c = random.choices(list(mcolors.CSS4_COLORS.values()),k = len(unique_countries))
plt.figure(figsize=(20,20))
plt.title('Covid-19 Confirmed Cases in Countries Outside of Mainland China')
plt.pie(visual_confirmed_cases[1:], colors=c)
plt.legend(visual_unique_countries[1:], loc='best')
plt.show()

# MODELING

# 1. SVM

# Building the SVM model

kernel = ['poly', 'sigmoid', 'rbf']
c = [0.01, 0.1, 1, 10]
gamma = [0.01, 0.1, 1]
epsilon = [0.01, 0.1, 1]
shrinking = [True, False]
svm_grid = {'kernel': kernel, 'C': c, 'gamma' : gamma, 'epsilon': epsilon, 'shrinking' :
shrinking}

svm = SVR()
svm_search = RandomizedSearchCV(svm, svm_grid, scoring='neg_mean_squared_error',
cv=3, return_train_score=True, n_jobs=-1, n_iter=40, verbose=1)
svm_search.fit(X_train_confirmed, y_train_confirmed)
svm_search.best_params_
svm_confirmed = svm_search.best_estimator_
svm_pred = svm_confirmed.predict(future_forecast)
svm_confirmed
```

```
svm_pred
# check against testing data

svm_test_pred = svm_confirmed.predict(X_test_confirmed)
plt.plot(svm_test_pred)
plt.plot(y_test_confirmed)
print('MAE:', mean_absolute_error(svm_test_pred, y_test_confirmed))
print('MSE:',mean_squared_error(svm_test_pred, y_test_confirmed))
# Total Number of coronavirus cases over time

plt.figure(figsize=(20, 12))
plt.plot(adjusted_dates, world_cases)
plt.title('Number of Coronavirus Cases Over Time', size=30)
plt.xlabel('Days Since 1/22/2020', size=30)
plt.ylabel('Number of Cases', size=30)
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
# Confirmed vs Predicted cases

plt.figure(figsize=(20, 12))
plt.plot(adjusted_dates, world_cases)
plt.plot(future_forcast, svm_pred, linestyle='dashed', color='purple')
plt.title('Number of Coronavirus Cases Over Time', size=30)
plt.xlabel('Days Since 1/22/2020', size=30)
plt.ylabel('Number of Cases', size=30)
plt.legend(['Confirmed Cases', 'SVM predictions'])
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
# Predictions for the next 10 days using SVM

print('SVM future predictions:')
set(zip(future_forcast_dates[-10:], svm_pred[-10:]))

# 2.Linear Regression

from sklearn.linear_model import LinearRegression
linear_model = LinearRegression(normalize=True, fit_intercept=True)
linear_model.fit(X_train_confirmed, y_train_confirmed)
test_linear_pred = linear_model.predict(X_test_confirmed)
```

```
linear_pred = linear_model.predict(future_forcast)
print('MAE:', mean_absolute_error(test_linear_pred, y_test_confirmed))
print('MSE:',mean_squared_error(test_linear_pred, y_test_confirmed))
plt.plot(y_test_confirmed)
plt.plot(test_linear_pred)
plt.figure(figsize=(20, 12))
plt.plot(adjusted_dates, world_cases)
plt.plot(future_forcast, linear_pred, linestyle='dashed', color='orange')
plt.title('Number of Coronavirus Cases Over Time', size=30)
plt.xlabel('Days Since 1/22/2020', size=30)
plt.ylabel('Number of Cases', size=30)
plt.legend(['Confirmed Cases', 'Linear Regression Predictions'])
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
# Predictions for the next 10 days using Linear Regression

print('Linear regression future predictions:')
print(linear_pred[-10:])
# Total deaths over time

plt.figure(figsize=(20, 12))
plt.plot(adjusted_dates, total_deaths, color='red')
plt.title('Number of Coronavirus Deaths Over Time', size=30)
plt.xlabel('Time', size=30)
plt.ylabel('Number of Deaths', size=30)
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
mean_mortality_rate = np.mean(mortality_rate)
plt.figure(figsize=(20, 12))
plt.plot(adjusted_dates, mortality_rate, color='orange')
plt.axhline(y = mean_mortality_rate,linestyle='--', color='black')
plt.title('Mortality Rate of Coronavirus Over Time', size=30)
plt.legend(['mortality rate', 'y='+str(mean_mortality_rate)])
plt.xlabel('Time', size=30)
plt.ylabel('Mortality Rate', size=30)
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
# Coronavirus Cases Recovered Over Time
```

```
plt.figure(figsize=(20, 12))
plt.plot(adjusted_dates, total_recovered, color='green')
plt.title('Number of Coronavirus Cases Recovered Over Time', size=30)
plt.xlabel('Time', size=30)
plt.ylabel('Number of Cases', size=30)
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
# Number of Coronavirus cases recovered vs the number of deaths

plt.figure(figsize=(20, 12))
plt.plot(adjusted_dates, total_deaths, color='r')
plt.plot(adjusted_dates, total_recovered, color='green')
plt.legend(['deaths', 'recoveries'], loc='best', fontsize=20)
plt.title('Number of Coronavirus Cases', size=30)
plt.xlabel('Time', size=30)
plt.ylabel('Number of Cases', size=30)
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
# Coronavirus Deaths vs Recoveries

plt.figure(figsize=(20, 12))
plt.plot(total_recovered, total_deaths)
plt.title('Coronavirus Deaths vs Coronavirus Recoveries', size=30)
plt.xlabel('Total number of Coronavirus Recoveries', size=30)
plt.ylabel('Total number of Coronavirus Deaths', size=30)
plt.xticks(size=15)
plt.yticks(size=15)
plt.show()
```

CHAPTER 7
REQUIREMENTS

7.1Hardware requirements

Processor : Pentium i3 or higher.

RAM : 4 GB or higher.

Hard Disk Drive : 20 GB (free).

Peripheral Devices : Monitor, Mouse and Keyboard

7.2 Software Requirements

Operating system : Windows 8/11.

IDE Tool : Google Colab

Visualization tool: Tableau

Coding Language : Python

APIs : Numpy, Pandas, Matplotlib

Dataset: Kaggle

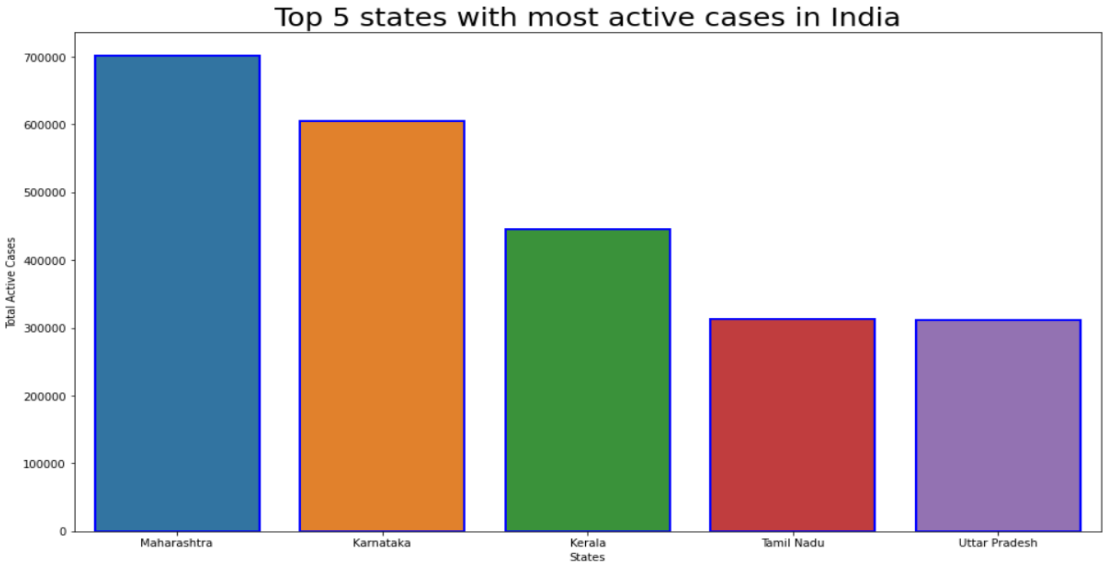
CHAPTER 8

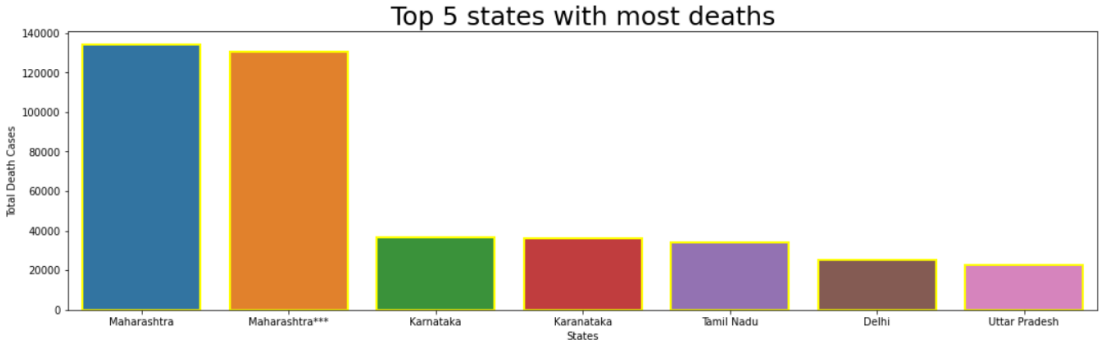
PROJECT FINDINGS

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed
0	1	2020-01-30	6:00 PM	Kerala	1	0	0	0	1
1	2	2020-01-31	6:00 PM	Kerala	1	0	0	0	1
2	3	2020-02-01	6:00 PM	Kerala	2	0	0	0	2
3	4	2020-02-02	6:00 PM	Kerala	3	0	0	0	3
4	5	2020-02-03	6:00 PM	Kerala	3	0	0	0	3
5	6	2020-02-04	6:00 PM	Kerala	3	0	0	0	3
6	7	2020-02-05	6:00 PM	Kerala	3	0	0	0	3
7	8	2020-02-06	6:00 PM	Kerala	3	0	0	0	3
8	9	2020-02-07	6:00 PM	Kerala	3	0	0	0	3
9	10	2020-02-08	6:00 PM	Kerala	3	0	0	0	3

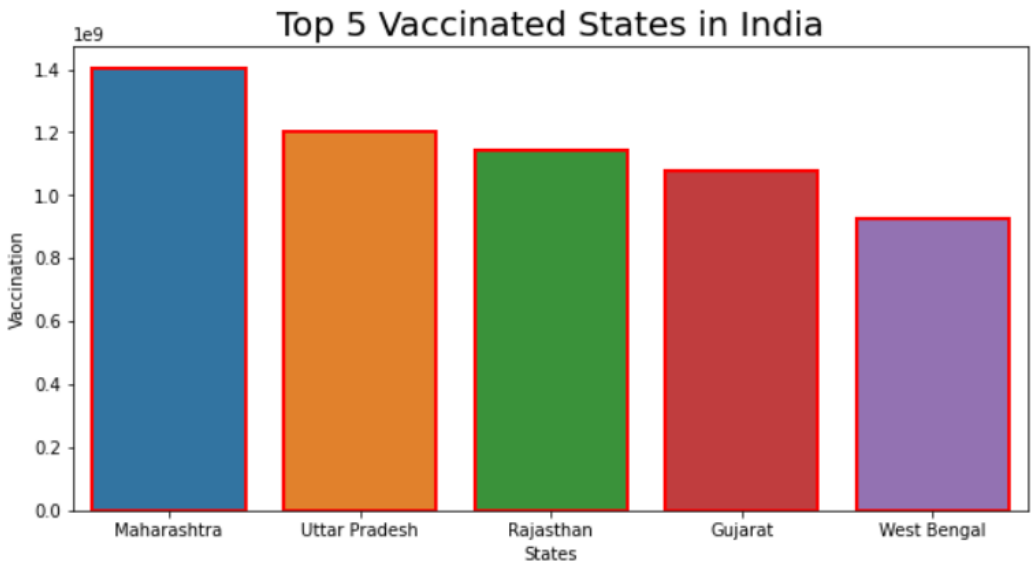
]

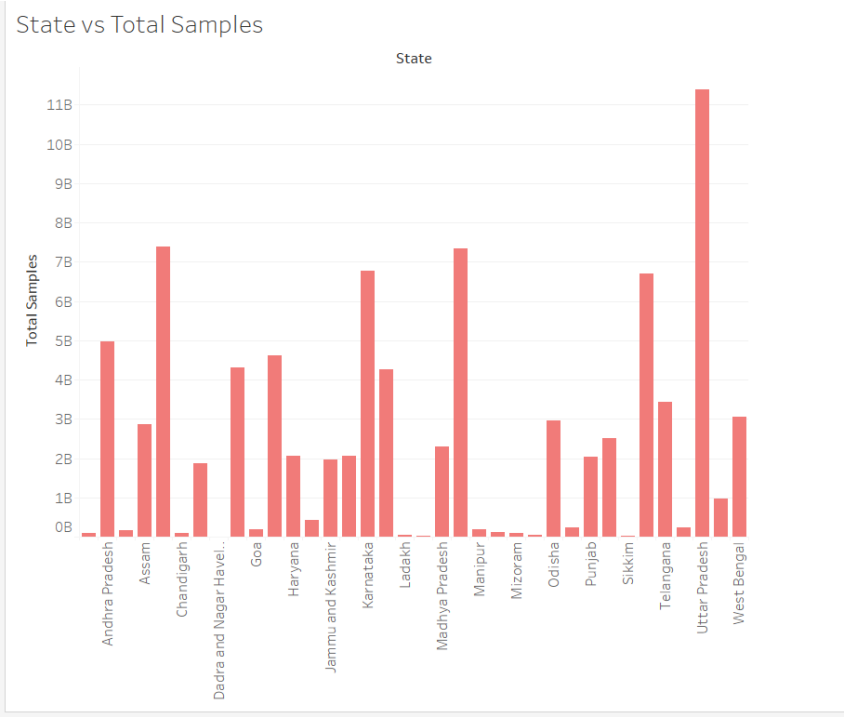
State/UnionTerritory	Confirmed	Cured	Deaths	Recovery Rate	Mortality Rate
Maharashtra	6363442	6159676	134201	96.797865	2.108937
Maharashtra* **	6229596	6000911	130753	96.329056	2.098900
Kerala	3586693	3396184	18004	94.688450	0.501967
Karnataka	2921049	2861499	36848	97.961349	1.261465
Karanataka	2885238	2821491	36197	97.790581	1.254559
Tamil Nadu	2579130	2524400	34367	97.877967	1.332504
Andhra Pradesh	1985182	1952736	13564	98.365591	0.683262
Uttar Pradesh	1708812	1685492	22775	98.635309	1.332797
West Bengal	1534999	1506532	18252	98.145471	1.189056
Delhi	1436852	1411280	25068	98.220276	1.744647
Chhattisgarh	1003356	988189	13544	98.488373	1.349870
Odisha	988997	972710	6565	98.353180	0.663804
Rajasthan	953851	944700	8954	99.040626	0.938721
Gujarat	825085	814802	10077	98.753704	1.221329
Madhya Pradesh	791980	781330	10514	98.655269	1.327559
Madhya Pradesh* **	791656	780735	10506	98.620487	1.327092
Haryana	770114	759790	9652	98.659419	1.253321
Bihar	725279	715352	9646	98.631285	1.329971
Bihar** **	715730	701234	9452	97.974655	1.320610
Telangana	650353	638410	3831	98.163613	0.589065



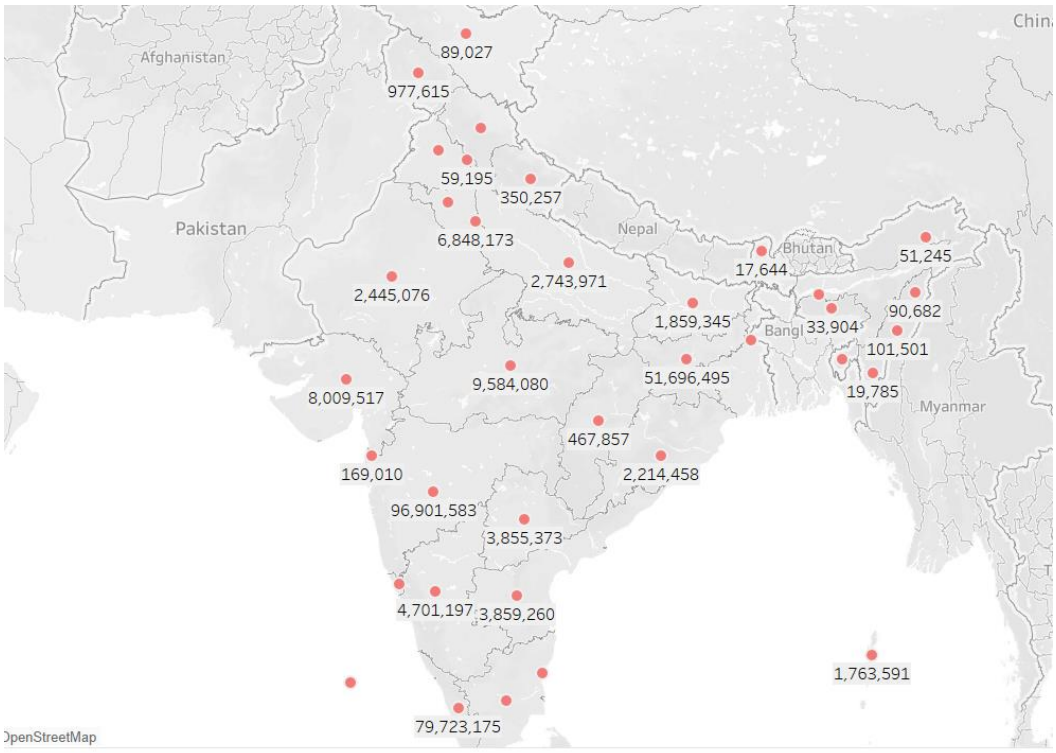


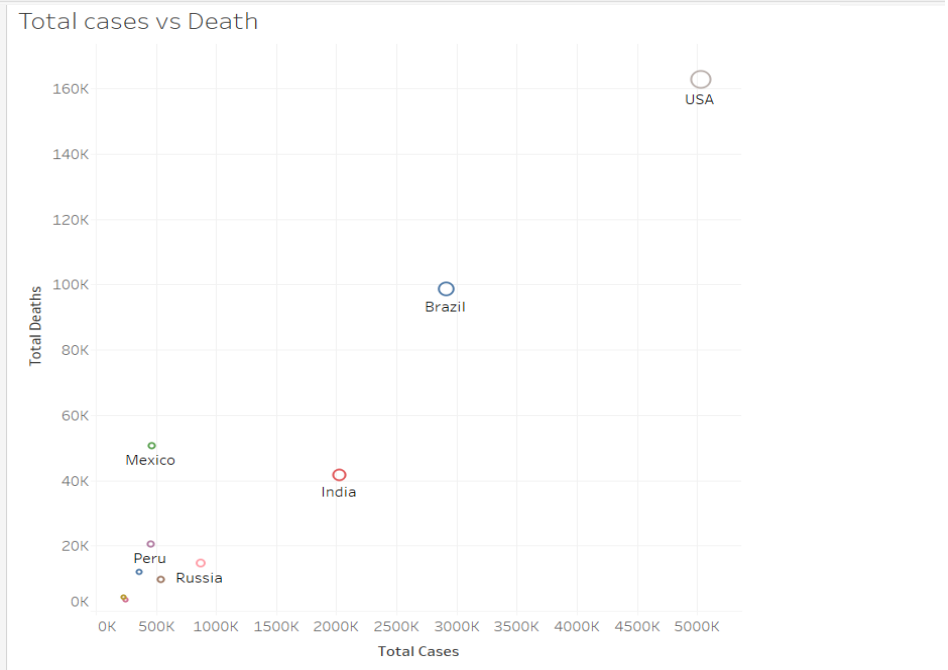
Male and Female Vaccination

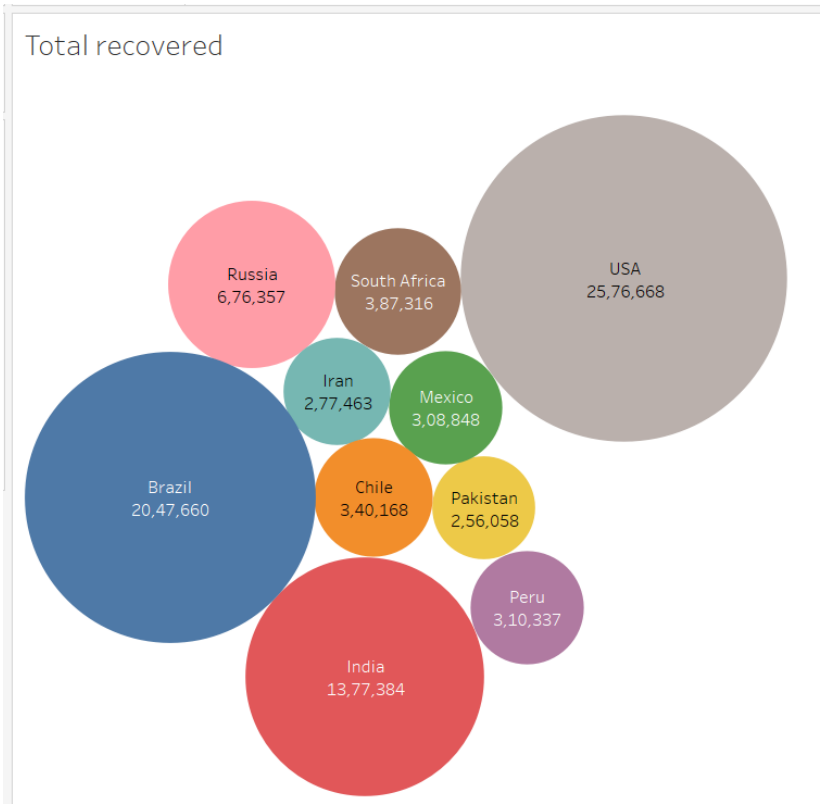

Top 5 Vaccinated States in India
1e9
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Vaccination
Maharashtra
Uttar Pradesh
Rajasthan
Gujarat
West Bengal
States


State vs Total Samples
State
11B
10B
9B
8B
7B
6B
5B
4B
3B
2B
1B
0B
Total Samples
Andhra Pradesh
Assam
Chandigarh
Dadra and Nagar Havel..
Goa
Haryana
Jammu and Kashmir
Karnataka
Ladakh
Madhya Pradesh
Manipur
Mizoram
Odisha
Punjab
Sikkim
Telangana
Uttar Pradesh
West Bengal

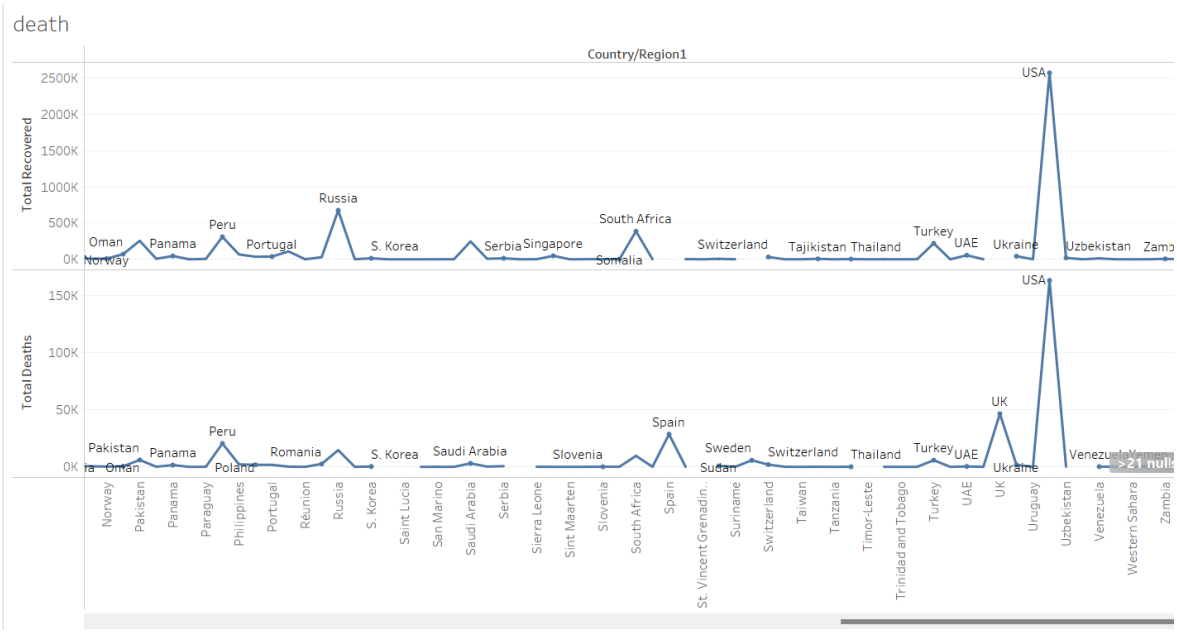
Total number of cases reported in each state:-



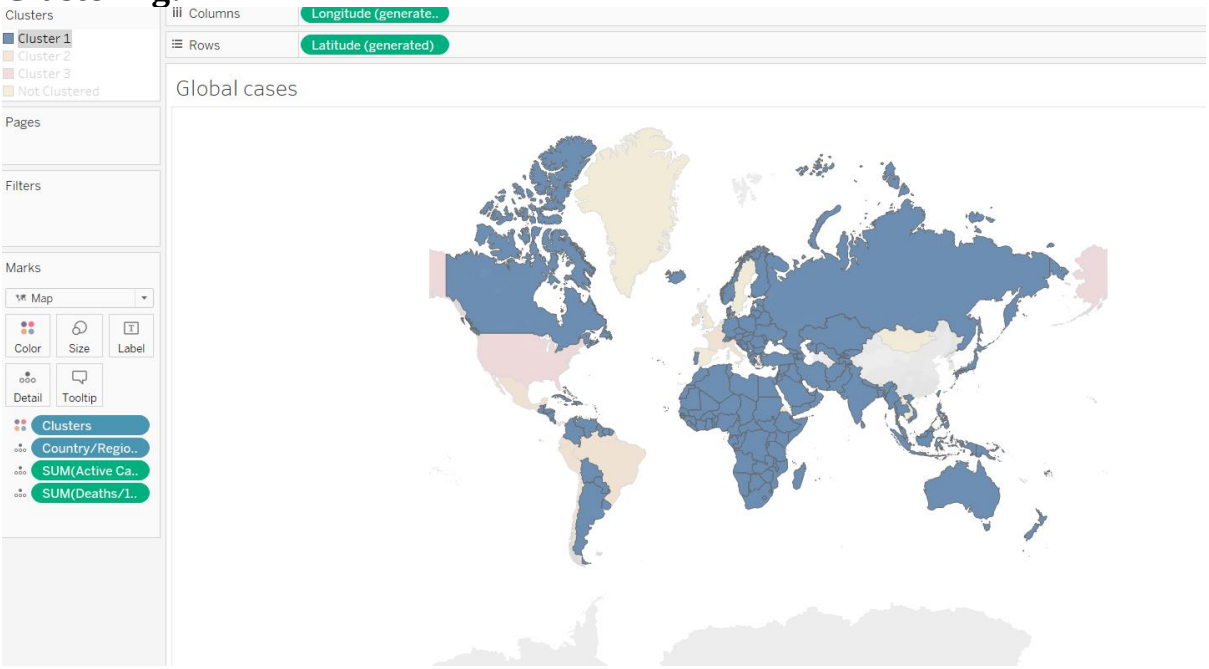


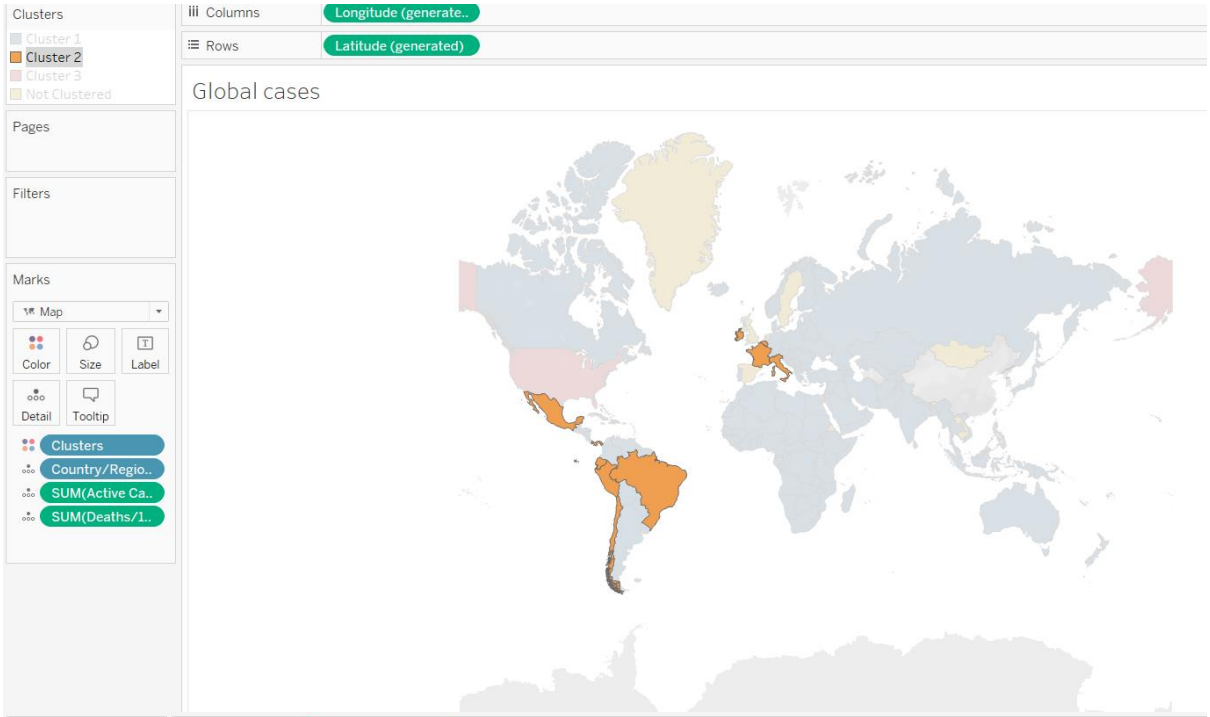


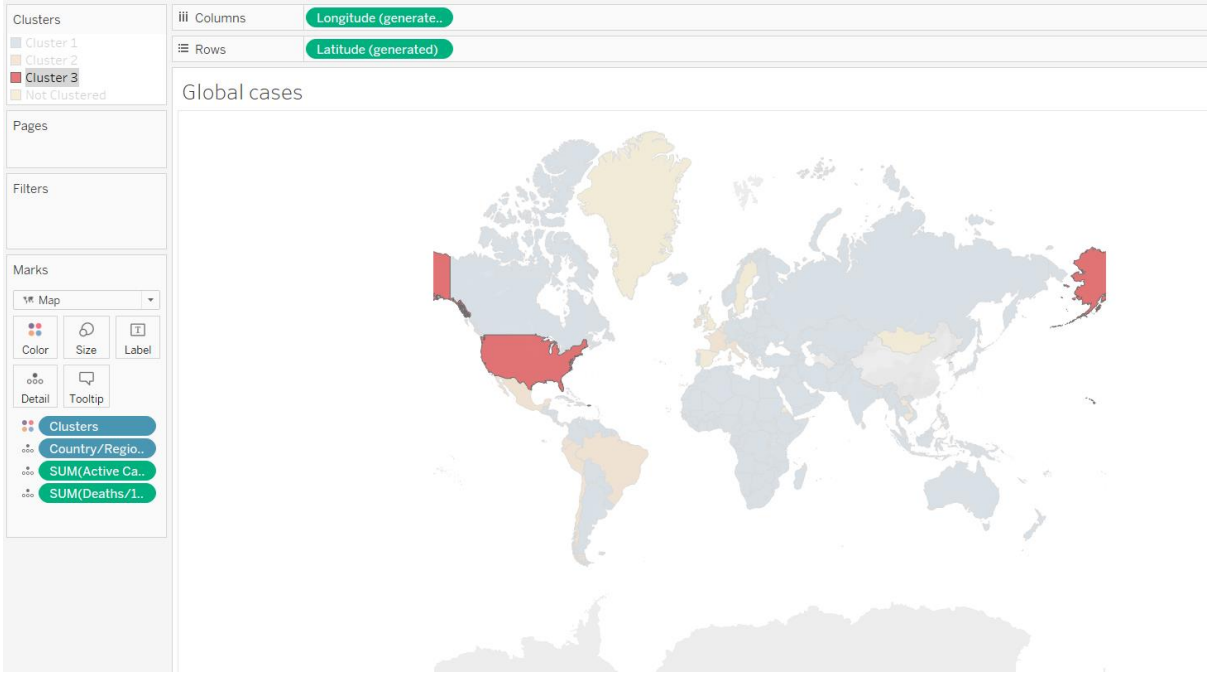
Global cases


death
Country/Region1
Total Recovered
2500K
2000K
1500K
1000K
500K
0K
Oman
Norway
Panama
Peru
Portugal
Russia
S. Korea
Serbia
Singapore
South Africa
Somalia
Switzerland
Tajikistan
Thailand
Turkey
UAE
Ukraine
USA
Uzbekistan
Total Deaths
150K
100K
50K
0K
Pakistan
Oman
Panama
Peru
Poland
Romania
S. Korea
Saudi Arabia
Slovenia
Spain
Sweden
Sudan
Switzerland
Thailand
Turkey
UAE
UK
Ukraine
USA
Uruguay
Uzbekistan
Venezuela
>21 null
Norway
Pakistan
Panama
Paraguay
Philippines
Portugal
Réunion
Russia
S. Korea
Saint Lucia
San Marino
Saudi Arabia
Serbia
Sierra Leone
Sint Maarten
Slovenia
South Africa
Spain
St. Vincent Grenadin...
Suriname
Switzerland
Taiwan
Tanzania
Timor-Leste
Trinidad and Tobago
Turkey
UAE
UK
Uruguay
Uzbekistan
Venezuela
Western Sahara
Zambia

Clustering:

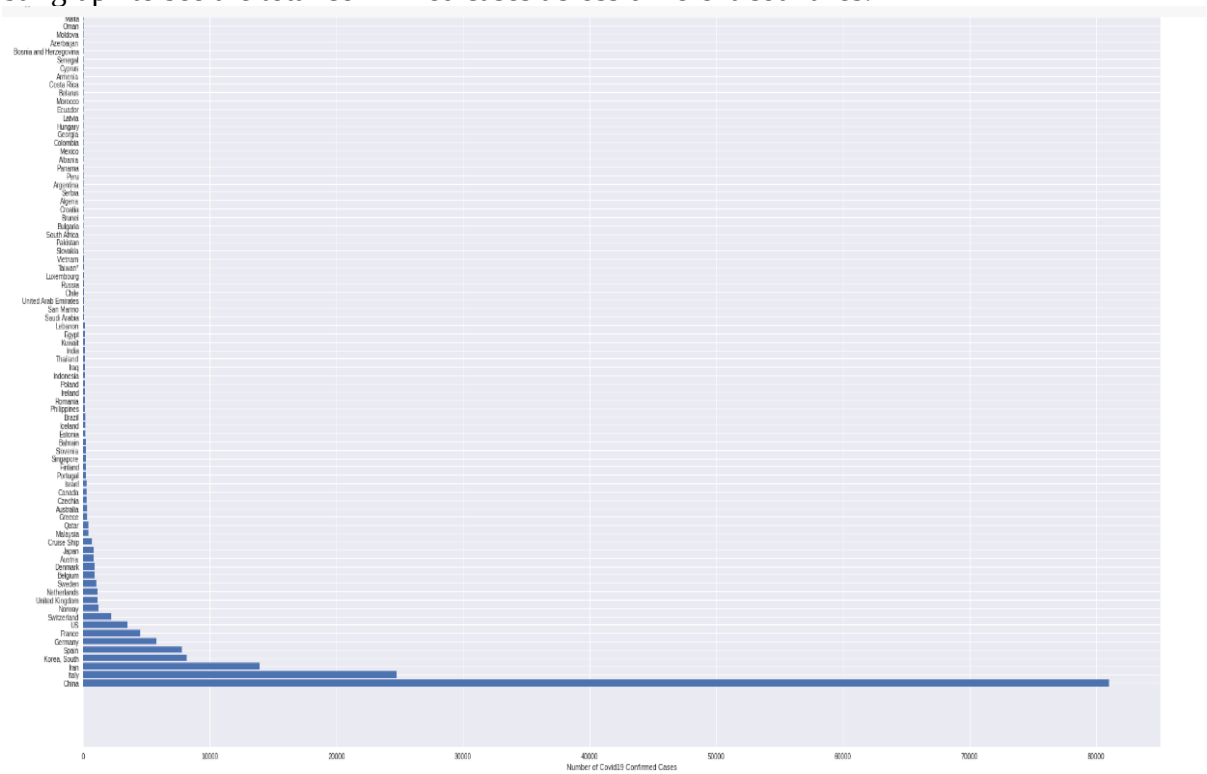

Clusters
Cluster 1
Cluster 2
Cluster 3
Not Clustered
Pages
Filters
Marks
Map
Color
Size
Label
Detail
Tooltip
Clusters
Country/Regio..
SUM(Active Ca..
SUM(Deaths/1..
Columns
Longitude (generate..
Rows
Latitude (generated)
Global cases


Clusters
Cluster 1
Cluster 2
Cluster 3
Not Clustered
Pages
Filters
Marks
Map
Color
Size
Label
Detail
Tooltip
Clusters
Country/Regio..
SUM(Active Ca..
SUM(Deaths/1..
Columns
Longitude (generate..
Rows
Latitude (generated)
Global cases

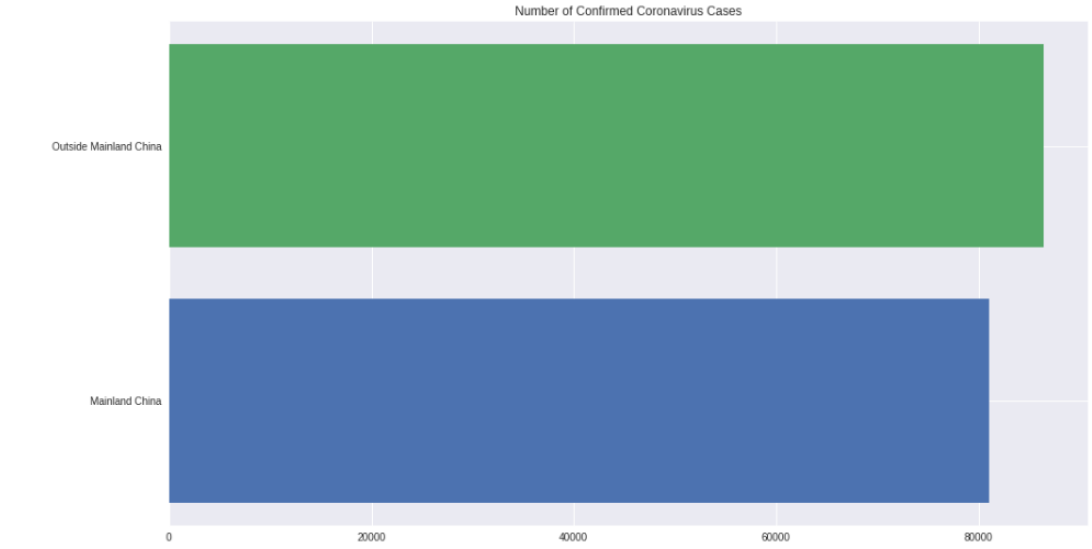

Clusters
Cluster 1
Cluster 2
Cluster 3
Not Clustered
Pages
Filters
Marks
Map
Color
Size
Label
Detail
Tooltip
Clusters
Country/Regio..
SUM(Active Ca..
SUM(Deaths/1..
Columns
Longitude (generate..
Rows
Latitude (generated)
Global cases

Analysing data set-2:-

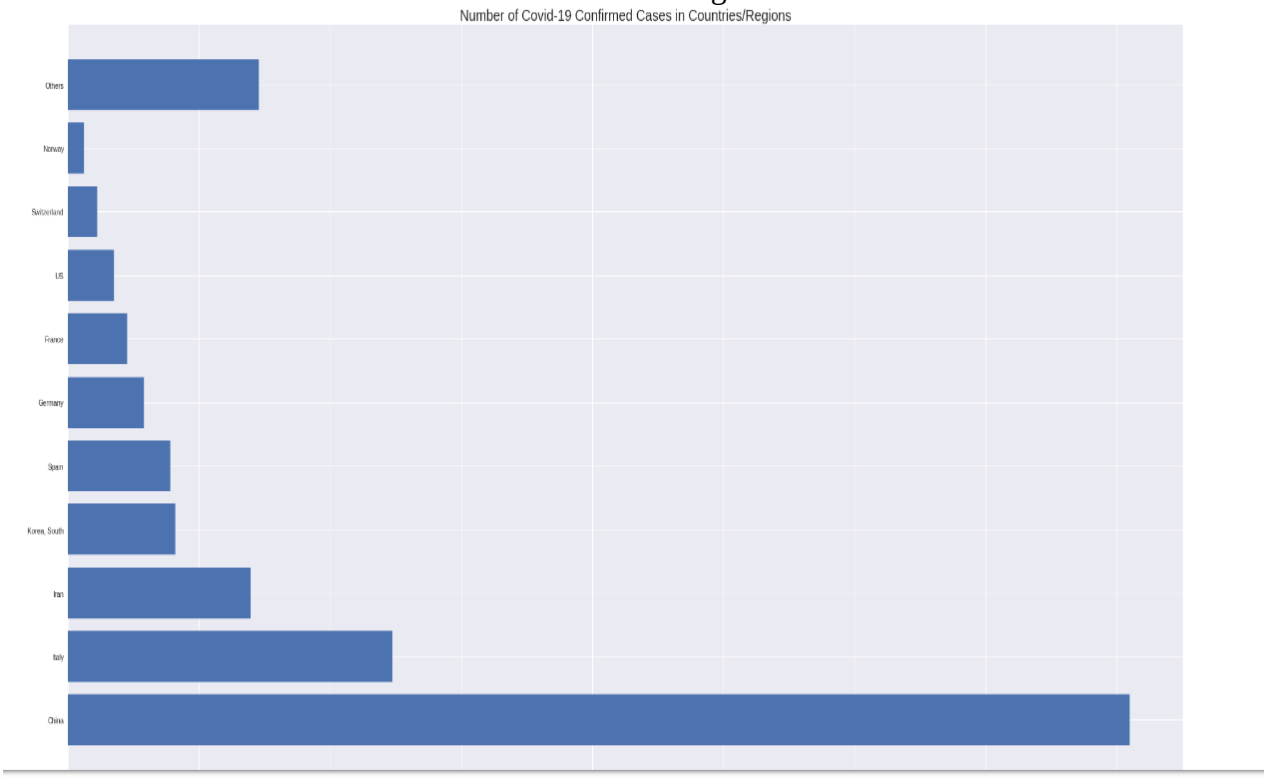
bar graph to see the total confirmed cases across different countries:-



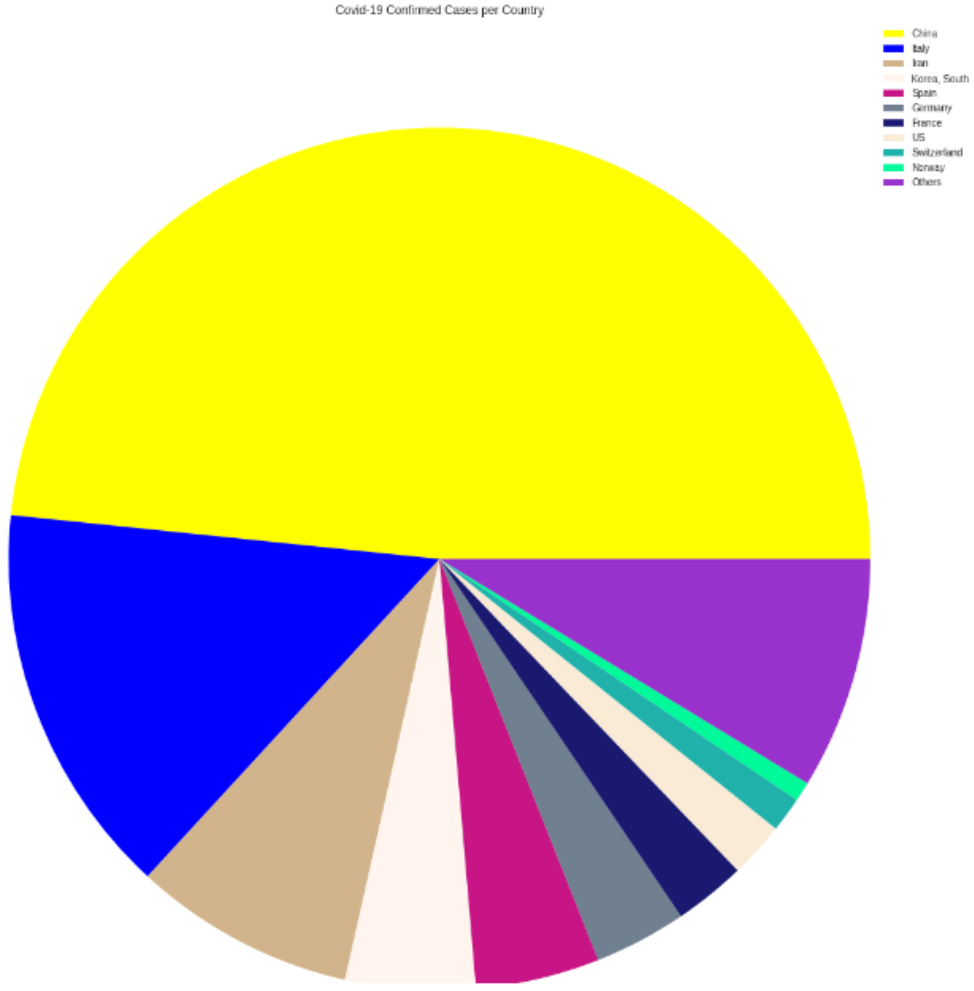
bar graph to see the total confirmed cases between mainland china and outside mainland china:-



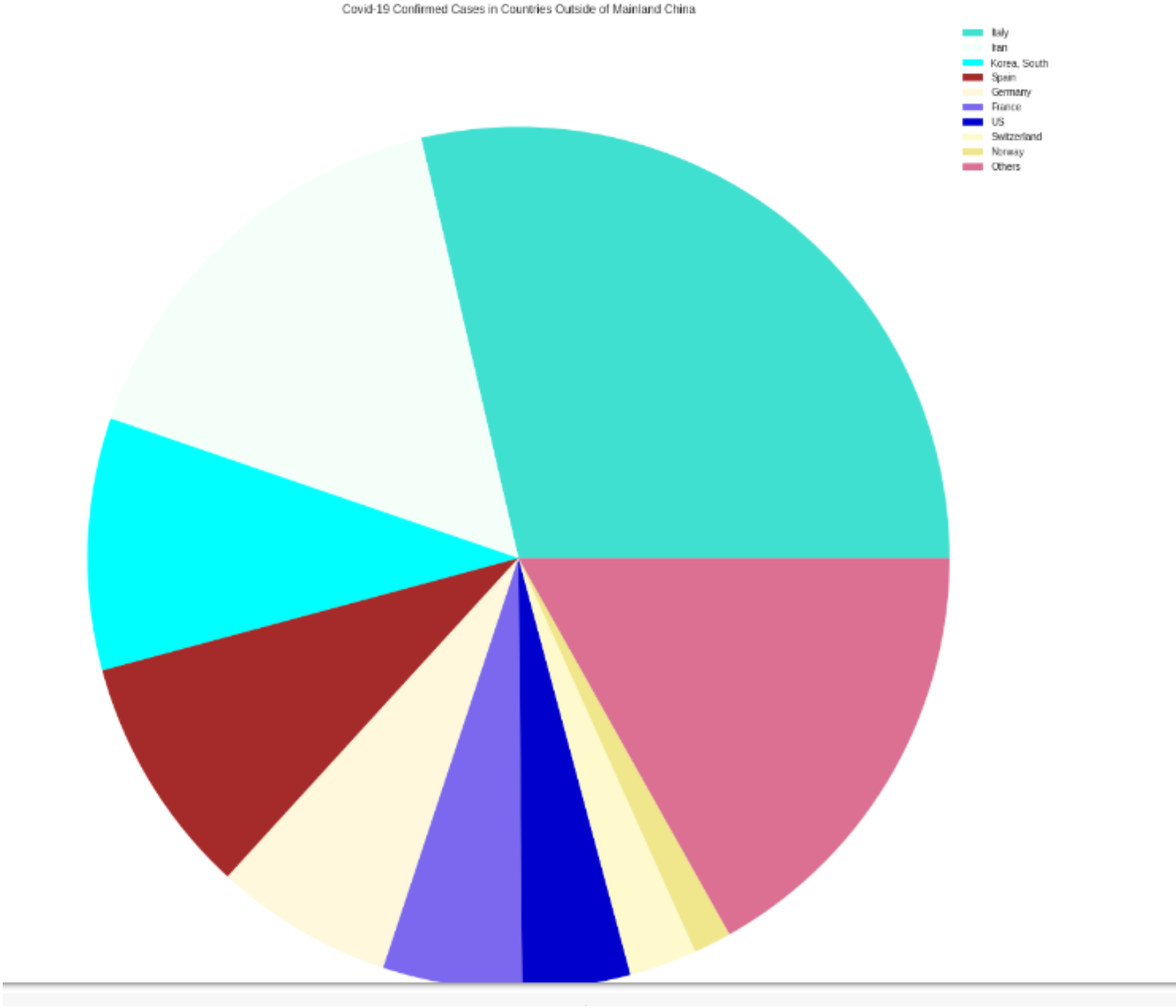
Number of Covid-19 Confirmed Cases in Countries/Regions:-



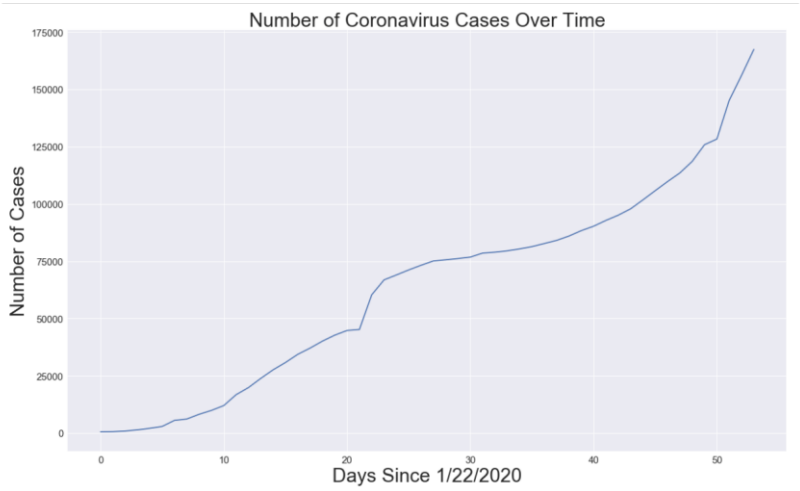
Covid-19 Confirmed Cases per Country:-



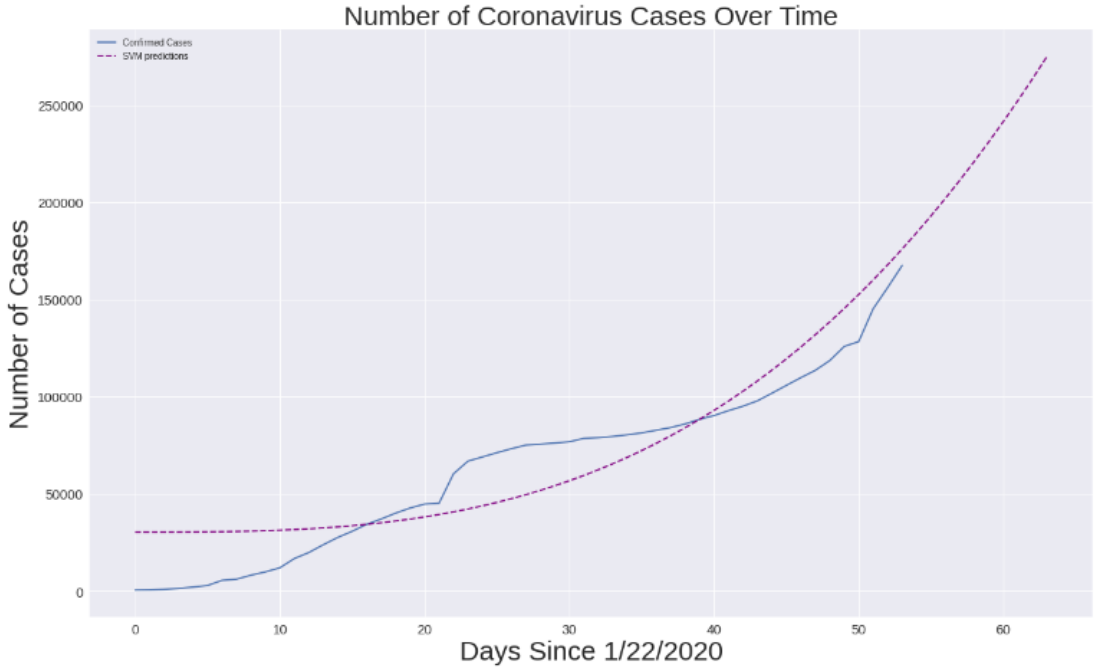
pie chart to see the total confirmed cases in 10 different countries outside China:-



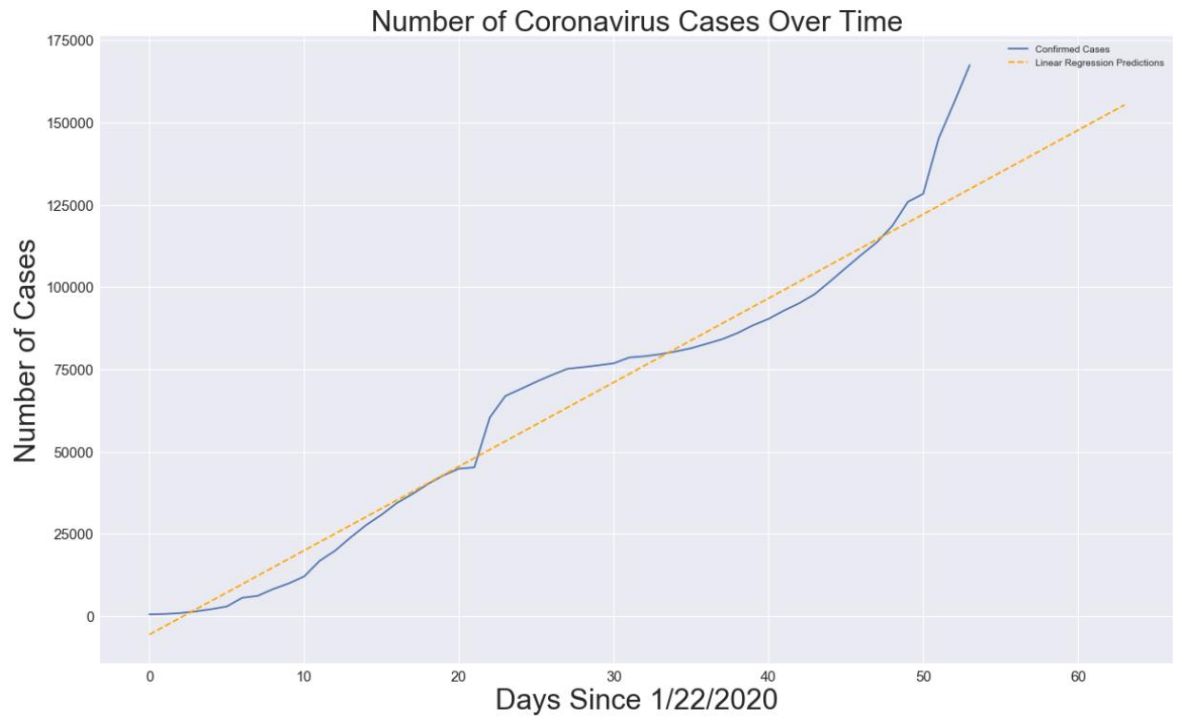
Implementing SVM model:-
Total Number of coronavirus cases over time:-



Confirmed vs Predicted cases:-



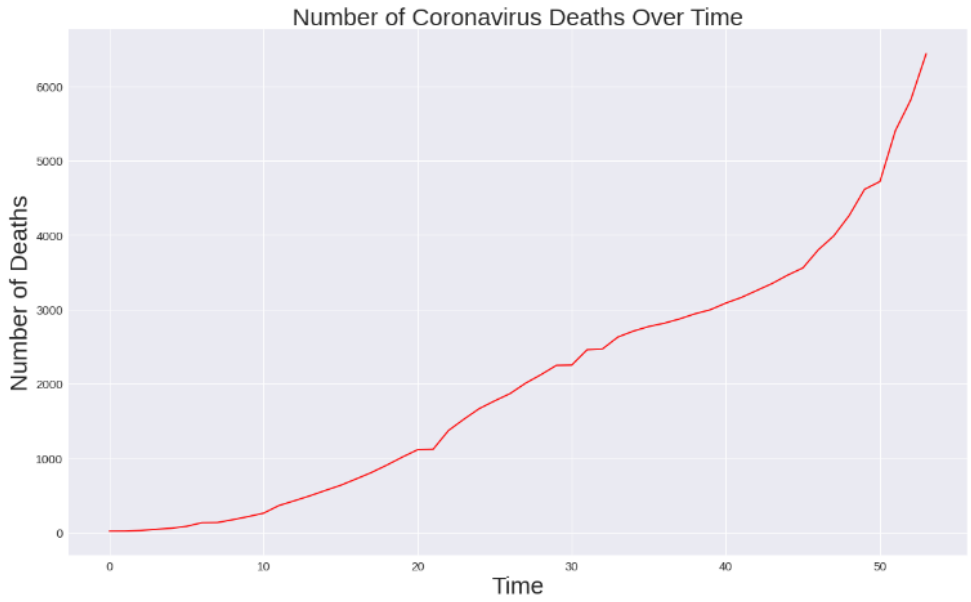
Linear regression model:



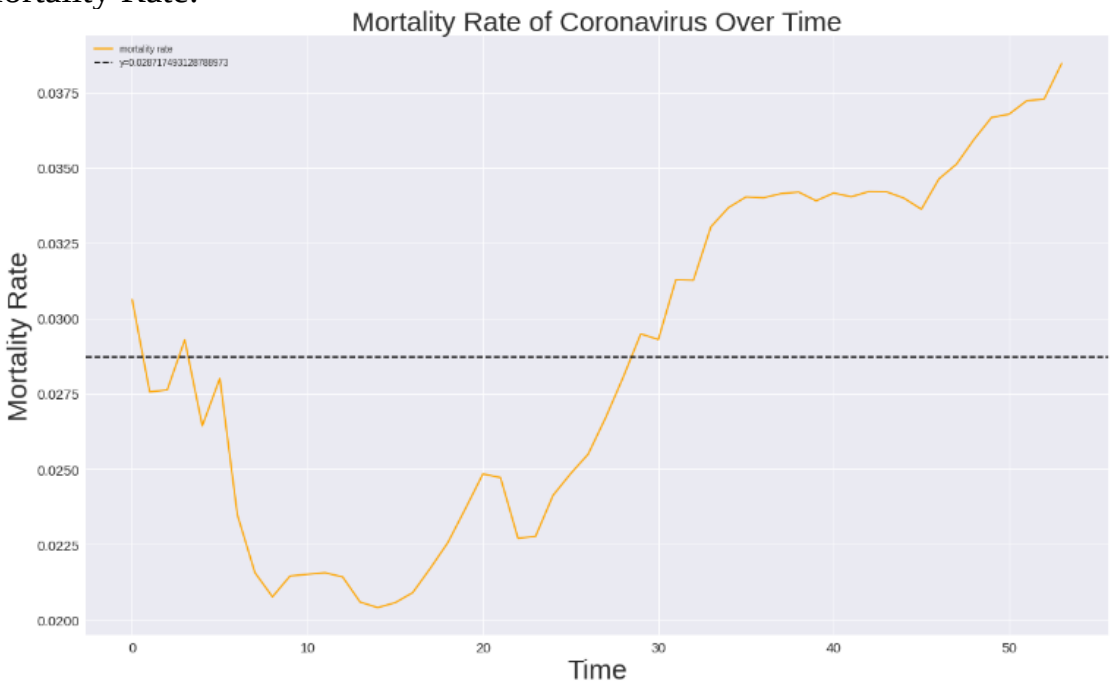
Predictions for the next 10 days using Linear Regression :

```
Linear regression future predictions:
[[132336.25252525]
 [134890.72222222]
 [137445.19191919]
 [139999.66161616]
 [142554.13131313]
 [145108.6010101 ]
 [147663.07070707]
 [150217.54040404]
 [152772.01010101]
 [155326.47979798]]
```

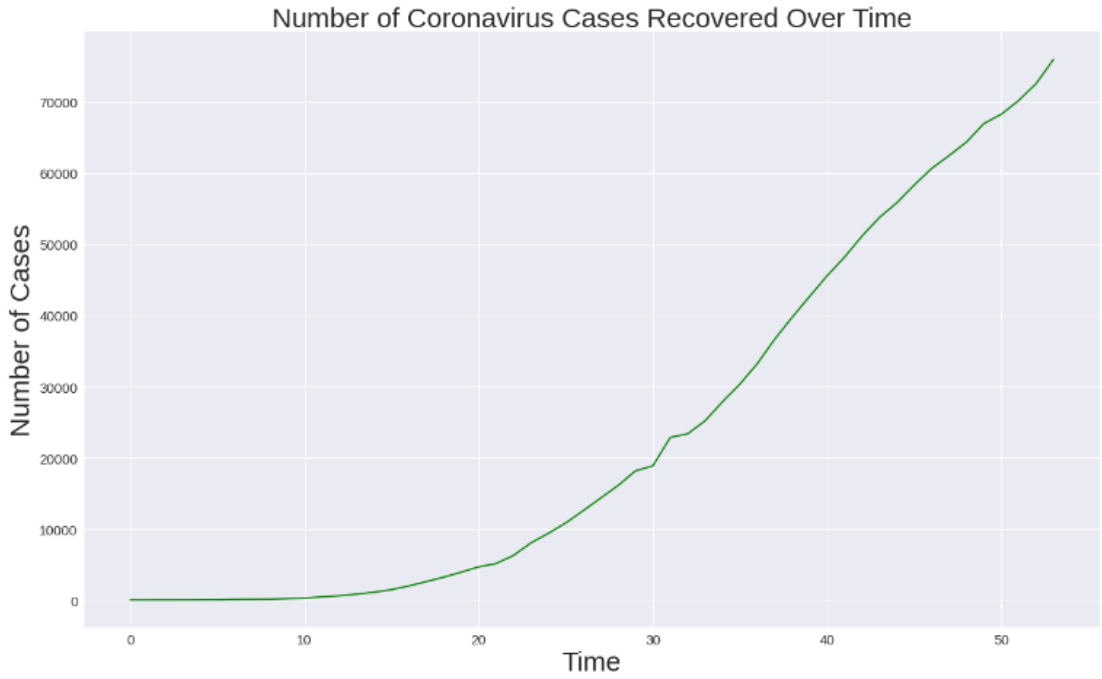
Total deaths over time:-



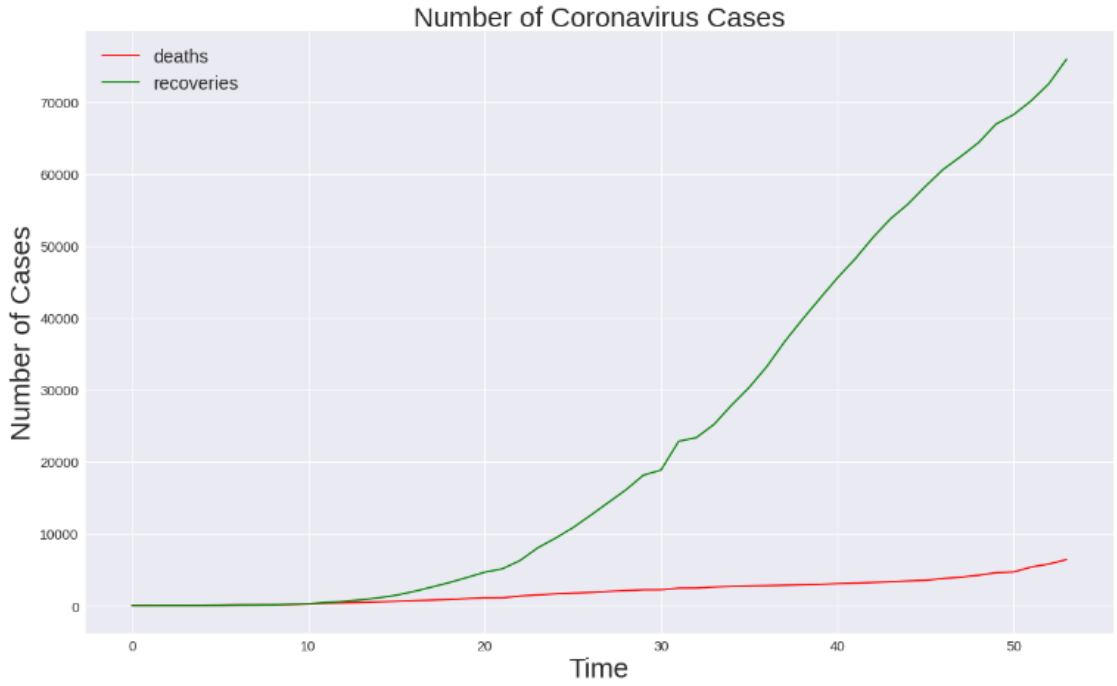
Mortality Rate:-



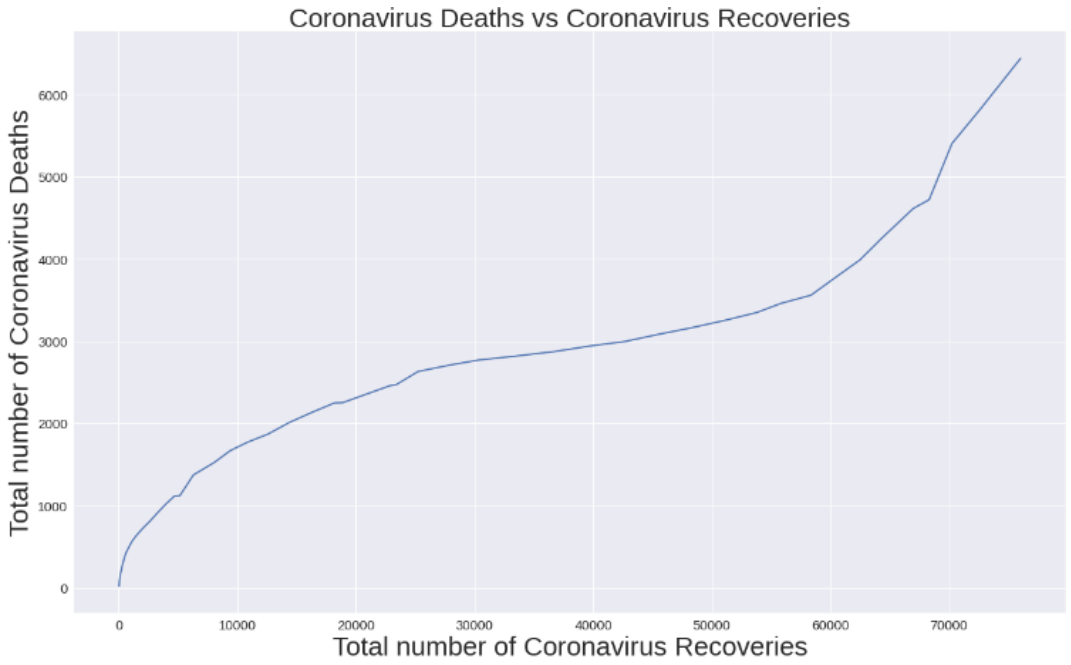
Coronavirus Cases Recovered Over Time:



Number of Coronavirus cases recovered vs the number of deaths:-



Coronavirus Deaths vs Recoveries:-



CHAPTER 9
CONCLUSION

The deadly novel coronavirus termed as SARS2 has caused thousands of deaths across the world since December 2019. This COVID-19 has been declared as a pandemic and the whole world was not in more danger after World War II.

Since no treatment is available, the WHO has recommended that infection should be avoided by frequent hand washing, social distancing, keeping unwashed hands away from the face, and covering coughs and sneezes with a tissue or inner elbow. Since firmed cases and deaths are increasing every day, and the virus hotspot has changed several times, it is very difficult to completely describe the nature of COVID current data. Still this work provides data analytics and data visualization to describe different aspects of the disease using the currently available datasets. Based on the dataset used and the data analytics of this paper, it can be seen that the combination of fever and cough is one of the major indicators of carrying this virus. It is also found that many patients develop the symptoms within 14 days of exposure. Furthermore, the currently available data shows that males and elderly people are more affected by the disease. It is also shown that the number of confirmed infections is much more in countries which have low average temperature compared to countries with high completely describe the nature of COVID-19 with current data. Still this work provides data analytics and data visualization to describe different aspects of the disease using the currently available datasets. Based on the dataset used and the data analytics of this paper,it can be seen that the combination of fever and cough is one of the major indicators of carrying also found that many patients develop the symptoms within 14 days of exposure. Furthermore, the currently available data shows that males and elderly people are more affected by the disease. It is also shown that the number of confirmed infections is much more in countries which have low average temperature compared to countries with high average temperature. It can be seen that although the disease started in China, currently China has managed to restrict the spread of the disease. On the other hand, Brazil ,USA now have very high number of confirmed cases and deaths. More investigation is required to gain a clear understanding of the disease and to find means to deal with it.

CHAPTER 10

FUTURE ENHANCEMENTS

The pandemic of COVID-19 has affected the entire globe. It has spread in more than 85 countries as of Apr. 2020. Scientists have made every effort to find solutions to it; according to claims by the United States and India, some vaccines have been made that are being trialed. The use of computers by scientists for early prediction has been widespread. A lot of research is taking place using ML to combat COVID-19. This chapter can be used by different researchers to learn how ML can be employed to forecast not only this situation but also other cases. The chapter specifically used the SVM method of time to forecast the stability and growth of COVID-19. Many countries have seen high totals of deaths owing to COVID-19. It is believed that the performance of the model can be improved or the model can give more accurate data if more datasets are available. The model gives results on the basis of data developed by information given by health agencies. Thus, forecasting may not be 100% accurate, but it can surely be used as a corrective measure. For future work further enhancement can be done by combining new factors and algorithms with SVM to get more accurate results.

REFERENCES:-

1)World Health Organization, "Coronavirus disease 2019 (COVID-19) Situation Report- 13," World Health Organization, 2020.
2) .Corona Tracker Community, "Corona Tracker," Corona Tracker, (2020). https://www.coronatracker.com/
3) WHO. Novel coronavirus – China. Jan 12, 2020. http://www.who.int/csr/don/12-january-2020-novel-coronavirus-china/en/
4) COVID19_line_list_data.csv
Available:https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-dataset
[1] **1)Covid-19 impact on Indian Economy**
Authors:
- S. Mahendra Dev (Indira Gandhi Institute of Development Research)
- Rajeshwari Sengupta (Indira Gandhi Institute of Development Research)

[2] **Data Analysis for Covid-19**
Authors:
- Meenakshi Jha
- Noopur Khare
- Abhimanyu Kumar Jha

[3] **Spatial Analysis of COVID-19**
Authors:-
- Pavani Pattipati
- Mahendra Aseri

[4] **Data Visualization & Analysis of Covid 19**
Authors:
- Fahima Khanam
- Itisha Norwin

[5] **Big Data Visualization and Visual Analytics of COVID-19 Data**
Authors:-
- Carson K. Leung;
- Yubo Chen;
- Calvin S.H. Hoi;
- Siyuan Shang;
- Yan Wen;
- Alfredo Cuzzocrea