PREDICTION & ANALYSIS ON SPREAD OF NOVEL COVID-19

CAPSTONE PROJECT REPORT

June 2020

Submitted by

JOSINA P JOY
KRISHNAMRAJU NALIMELA
SUBHALAXMI BABOO

To

The Amity University Online in partial fulfillment of the requirements for the award of the Post Graduate Diploma

In

BUSINESS ANALYTICS & INTELLIGENCE

Contents

List of Tables

ACRONYMS & ABBREVIATIONS

ACF	Auto Correlation Factor
AIC	Akaike information criterion
BIC	Bayesian information criterion
BI	Business Intelligence
COVID-19	Corona Virus Disease 2019
EDA	Exploratory Data Analysis
MAE	Mean Absolute Error
MAPE	Mean Absolute Percent Error
MASE	Mean Absolute Scaled Error
ME	Mean Error
MPE	Mean Percentage Error
PACF	Partial Auto Correlation Factor
RMSE	Root Mean Square Error
TS	Time Series
WBS	Work Breakdown Structure

ACKNOWLDGEMENT

The completion of this project required a lot of guidance and assistance from many people and we are extremely thankful to have received this along the duration of the project work.

We express our gratitude to Dr. SURESH VARADARAJAN & Dr. KARTHIC NARAYANAN, program co-directors of BA&I for providing us basic knowledge for our work.

We express our heartfelt thanks to Prof: GAURAV MISHRA for providing us valuable insights through the webinar sessions, this helped us a lot in our project work.

We express our sincere thanks to VISHAL SAWANT (project coordinator) for his valuable suggestions and support and Johns Hopkins University for making the data set available.

Last but not the least we thank the Almighty God for helping us in successfully completing the report of this capstone project.

1.INTRODUCTION

Corona Virus Disease-2019 (COVID-19), an infectious disease caused by a novel coronavirus, is currently a major worldwide threat. COVID-19 outbreak is first observed in Wuhan City, Hubei Province of China in December 2019. This virus mainly affects the human respiratory system and is highly contagious, it can spread from animals to humans and from humans to humans. Within a short period, more than 200 countries are infected with this novel coronavirus. The number of COVID cases are increasing day by day. Now, the world is in a health war against this pandemic, over 6 million people were infected globally leading to around 4 lakhs of deaths. In such grave circumstances, it is very important to analyze and predict future infected cases to support the prevention of the disease and aid in healthcare service preparation.

1.1 OBJECTIVE

This project aims to analyze, visualize, perform live data comparisons using Johns Hopkins University dataset. Our major activities will comprise of prediction on overall Growth Rate, Recovery Rate, and Mortality Rate by using Auto-Regressive Integrated Moving Average (ARIMA) & PROPHET modelling on the number of cumulative cases overtime on virus spread. Additionally, perform short-term transmission prediction by the development of a time series model. Further, the results to be validated over MAPE, MASE, ACF statistics using R Programming.

1.2 ADVANTAGES

The prediction model will have the below advantages such as:

- One can study and analyse the growth and spread of the virus.
- One can alarm the authorities about the approximate number of infected cases in the next 40 days.
- Helps the Government to take adequate health care measures such as arranging necessary equipments and covid specialised hospitals, preparing front line workers.

2.SCOPE

The project scope is detailed below :

- Identification of dataset with all required attributes such as reporting date, number of confirmed cases, recovered cases, deceased cases, country.
- Data preparation & Exploratory Data Analysis (EDA) on the global dataset using R visualization tools.
- To develop a dynamic Power BI dashboard to get better insights from the dataset.
- To build time series forecasting models and predict the approximate number of infected people in the next 40 days.

2.1 GANTT CHART

The project planning was initiated using a Gantt chart. Keeping the project objectives, all the activities were listed down and milestones with target dates were defined. Further, a regular progress follow-up of the individual tasks will be made. Any deviations from the planned work will be studied and addressed effectively.

Key highlights:

Project start date: 05/06/2020
Project end date: 23/06/2020
Total number of tasks: 6

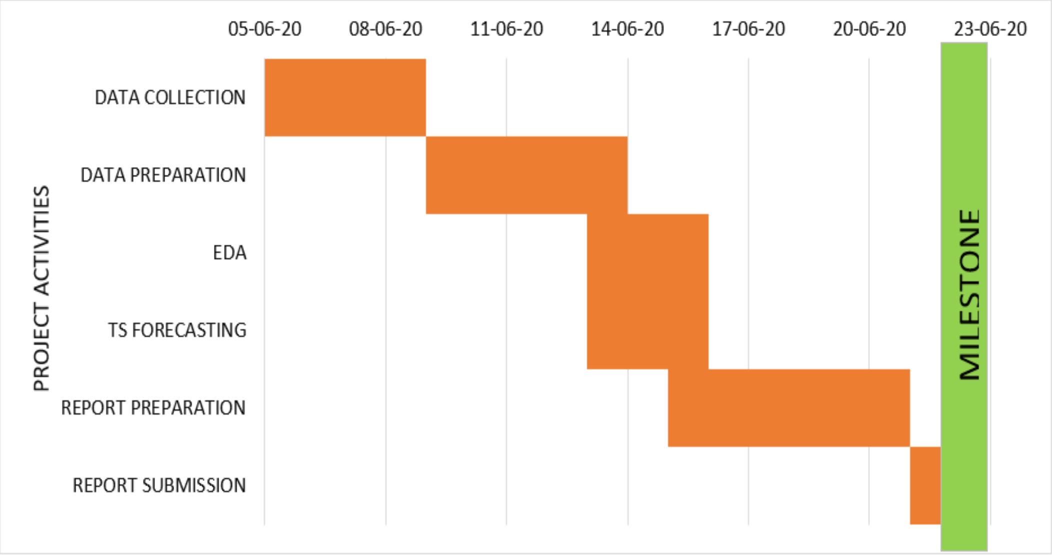


Figure 1 Timeline Of Project Activities

2.2 WBS (Work Break Down Structure)

This tool divides the project into different branches which helps the team to structure the work packages and understand the dependency between various sub-tasks. In addition to a visual representation, the amount of work effort could be estimated from the WBS. The project has 3 major tasks which are Data preparation, EDA and TS forecasting.

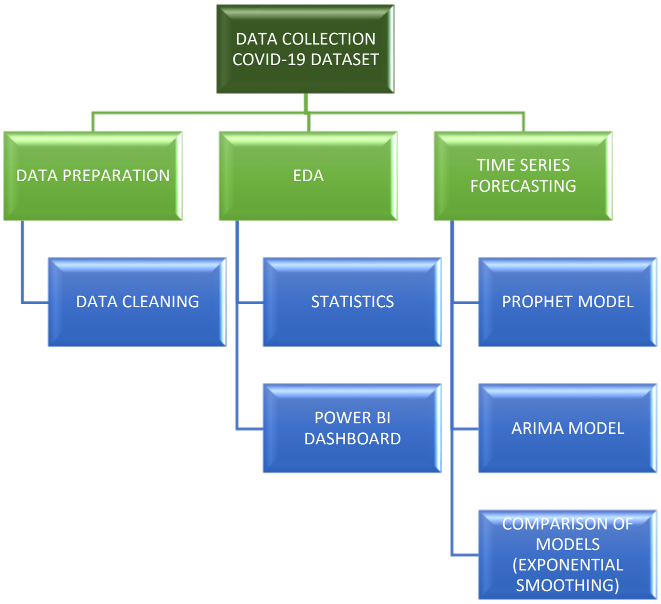


Figure 2 Work Break Down Structure

2.3 RISK ANALYSIS

After defining the project requirements and WBS, team has analyzed different scenarios and identified 14 risks. These risks were further evaluated on the basis of severity (the potential effect of the failure) and occurrence (likelihood that the failure will occur) to deduces the criticality. The criticality was classified as Very High, High, Medium and Low based on:

Criticality = Severity x Occurrence   Where, Severity scale is from 1 to 10

Occurrence scale is from 1 to 10

Initially, at the initial project phase with a goal to convert risk into opportunity, these risks were categorized and mitigation actions were adopted.

Risk no:	Risk	Severity	Occurrence	Criticality	Mitigation
1	Miscommunication between team members	9	8	72	Created a WhatsApp group with team members and project coordinator
2	Lack of coordination	6	8	48	Scheduled a daily zoom meeting to share the project updates
3	Lack of knowledge in setting ts frequency	9	5	45	Referred similar journal, websites.
4	Improper requirement definition	10	3	30	Identified the necessary attributes

Table 1 : Risk Matrix

It was observed that miscommunication is one of the major factor to be considered with high criticality value. Suitable mitigation action were adopted to avoid it.

3.PEDAGOGY

The major challenge we have faced after defining the problem statement was the collection of a relevant dataset. As we know, based on the type of data collected, data is grouped into 3 classes. They are Cross-Sectional data, Time Series data and Panel data. Data collected on many variables of interest at the same time or duration of time is called cross-sectional data. Data collected for a single variable over several time intervals (daily, weekly, monthly etc.) is called a time series data. And panel data is data collected on several variables (multiple dimensions) over several time intervals. In this project, we are using univariate time series forecasting. We have divided the project into 3 major tasks, they are data preparation, EDA and time series forecasting.

3.1 DATA PREPARATION & CLEANING

Data preparation is a vital step in statistical analysis and for model building.

Data Source Link

Column Descriptions of this dataset

Observation Date: Observation date in mm/dd/yyyy

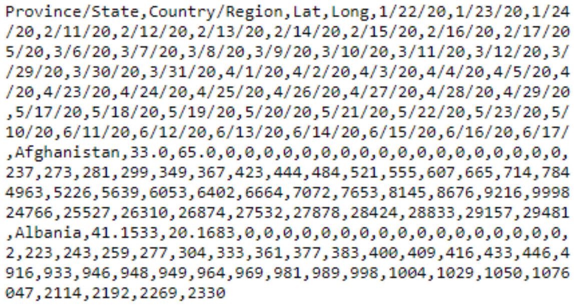
Country/Region: Country or region

Confirmed - Cumulative number of confirmed cases till that date

Deaths - Cumulative number of deaths till that date

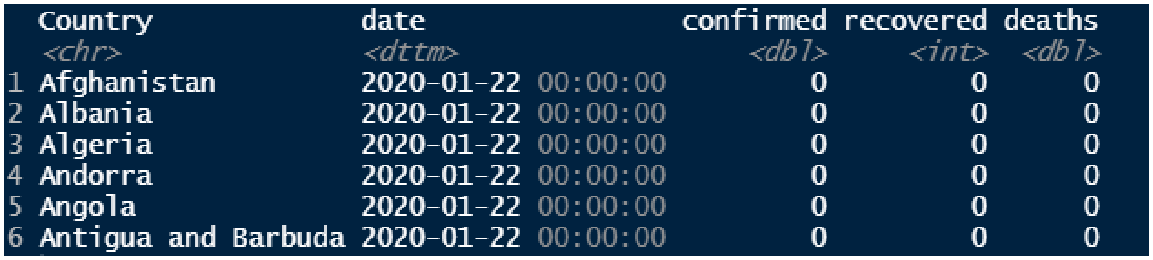
Recovered - Cumulative number of recovered cases till that date

Confirmed, recovered and death cases of COVID-19 infection are collected from Johns Hopkins University dataset as per World Health Organization region classification, from the official website of Johns Hopkins University (https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series).



```
Province/State,Country/Region,Lat,Long,1/22/20,1/23/20,1/24
/20,2/11/20,2/12/20,2/13/20,2/14/20,2/15/20,2/16/20,2/17/20
5/20,3/6/20,3/7/20,3/8/20,3/9/20,3/10/20,3/11/20,3/12/20,3/
/29/20,3/30/20,3/31/20,4/1/20,4/2/20,4/3/20,4/4/20,4/5/20,4
/20,4/23/20,4/24/20,4/25/20,4/26/20,4/27/20,4/28/20,4/29/20
,5/17/20,5/18/20,5/19/20,5/20/20,5/21/20,5/22/20,5/23/20,5/
10/20,6/11/20,6/12/20,6/13/20,6/14/20,6/15/20,6/16/20,6/17/
,Afghanistan,33.0,65.0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
237,273,281,299,349,367,423,444,484,521,555,607,665,714,784
4963,5226,5639,6053,6402,6664,7072,7653,8145,8676,9216,9998
24766,25527,26310,26874,27532,27878,28424,28833,29157,29481
,Albania,41.1533,20.1683,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
2,223,243,259,277,304,333,361,377,383,400,409,416,433,446,4
916,933,946,948,949,964,969,981,989,998,1004,1029,1050,1076
047,2114,2192,2269,2330
```

Raw data transformed



```
  Country             date                confirmed recovered deaths
  <chr>               <dttm>                  <dbl>     <int>  <dbl>
1 Afghanistan         2020-01-22 00:00:00         0         0      0
2 Albania             2020-01-22 00:00:00         0         0      0
3 Algeria             2020-01-22 00:00:00         0         0      0
4 Andorra             2020-01-22 00:00:00         0         0      0
5 Angola              2020-01-22 00:00:00         0         0      0
6 Antigua and Barbuda 2020-01-22 00:00:00         0         0      0
```

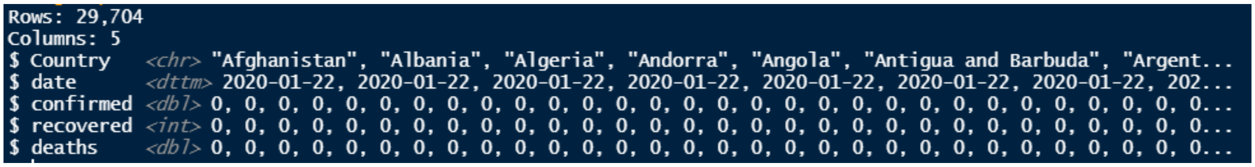
Below is R Program to clean up the raw data:

```
install.packages("tidyverse")
install.packages("lubridate")
install.packages("ggthemes")

library(tidyverse)
library(lubridate)
library(ggthemes)

confirmed <- read.csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv",stringsAsFactors = F)
deaths <- read.csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_deaths_global.csv",stringsAsFactors = F)
recovered <- read.csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_recovered_global.csv",stringsAsFactors = F)

# Get evolution data by country
  data_confirmed_sub <- confirmed %>%
  pivot_longer(names_to = "date", cols = 5:ncol(confirmed)) %>%
  select(-Province.State,-Lat, -Long) %>%
  group_by(`Country.Region`, date) %>%
  summarise("confirmed" = sum(value, na.rm = T))

  data_deaths_sub <- deaths %>%
  pivot_longer(names_to = "date", cols = 5:ncol(deaths)) %>%
  select(-Province.State,-Lat, -Long) %>%
  group_by(`Country.Region`, date) %>%
  summarise("deaths" = sum(value, na.rm = T))

  data_recovered_sub <- recovered %>%
  pivot_longer(names_to = "date", cols = 5:ncol(recovered)) %>%
  select(-Province.State,-Lat, -Long) %>%
  group_by(`Country.Region`, date) %>%
  summarise("recovered" = sum(value, na.rm = T))

  COVID19GlobalData <- data_confirmed_sub %>%
  full_join(data_recovered_sub) %>%
  full_join(data_deaths_sub)%>%
  ungroup() %>%
  mutate(date = as.POSIXct(date, format = "X%m.%d.%y")) %>%
  arrange(date) %>%
  group_by(`Country.Region`,date) %>%
  replace_na(list(deaths = 0, confirmed = 0)) %>%
  rename(Country=Country.Region)%>%
  ungroup()
```

3.2 EXPLORATORY DATA ANALYSIS & STATISTICS

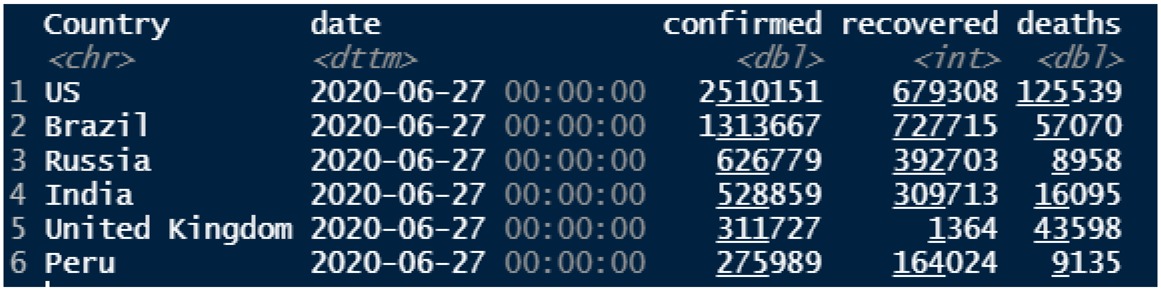
```
#Number of observations (rows) and variables, and a head of the first cases.
glimpse(COVID19GlobalData)

#Current Day Report by Country::::
TodayData <- COVID19GlobalData  %>% filter(date %in% max(COVID19GlobalData$date))%>%
arrange(desc(confirmed))
head(TodayData)
```

Sample data :



```
Rows: 29,704
Columns: 5
$ Country   <chr> "Afghanistan", "Albania", "Algeria", "Andorra", "Angola", "Antigua and Barbuda", "Argent...
$ date      <dttm> 2020-01-22, 2020-01-22, 2020-01-22, 2020-01-22, 2020-01-22, 2020-01-22, 2020-01-22, 202...
$ confirmed <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
$ recovered <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
$ deaths    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
```

Data as of : 27th Jun 2020 by top countries:

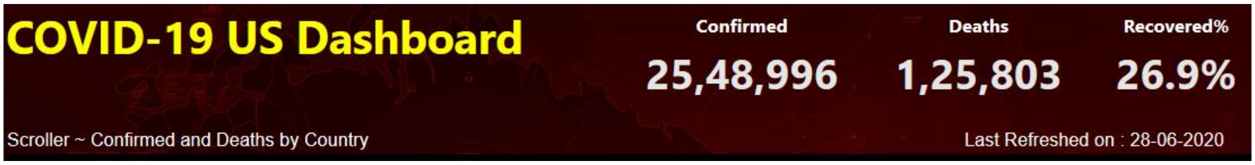


```
  Country        date                confirmed recovered deaths
  <chr>          <dttm>                  <dbl>     <int>  <dbl>
1 US             2020-06-27 00:00:00   2510151    679308 125539
2 Brazil         2020-06-27 00:00:00   1313667    727715  57070
3 Russia         2020-06-27 00:00:00    626779    392703   8958
4 India          2020-06-27 00:00:00    528859    309713  16095
5 United Kingdom 2020-06-27 00:00:00    311727      1364  43598
6 Peru           2020-06-27 00:00:00    275989    164024   9135
```

Summary Plot of Worldwide Cases - Confirmed, Deaths & Recovered Graphs:

Top10_Confirmed Cases by Country:

USA has most confirmed cases in world with 25,48,996 cases till June 28th and continues to be severe in in the upcomming weeks as well.. after that Brazil, Russia, india is in 2nd, 3rd, 4th places. India is among the 10 worst-affected countries by COVID-19.



COVID-19 US Dashboard

Confirmed	Deaths	Recovered%
25,48,996	1,25,803	26.9%

Scroller ~ Confirmed and Deaths by Country

Last Refreshed on : 28-06-2020

```r
Top10_Confirmed <- TodayData %>% select(Country,date,confirmed)%>%
  arrange(desc(confirmed)) %>% head(10) %>%
  ggplot(aes(x = reorder(`Country`,confirmed), y = confirmed )) +
  geom_bar(stat = "identity", fill  = "Orange", width = 0.8) +
  theme_economist() +
  scale_y_continuous(breaks = seq(0, 2000000, by = 200000)) +
  coord_flip() +
  labs(x = "", y = "", title = "Top 10 Countries by Confirmed Cases")+
  theme(axis.text.x = element_text(angle = 45)) +
    theme(axis.title = element_text(size = 14, colour = "black"),
          axis.text.y = element_text(size = 11, face = "bold"))
```

Here is a list of the most affected countries in terms of registered cases

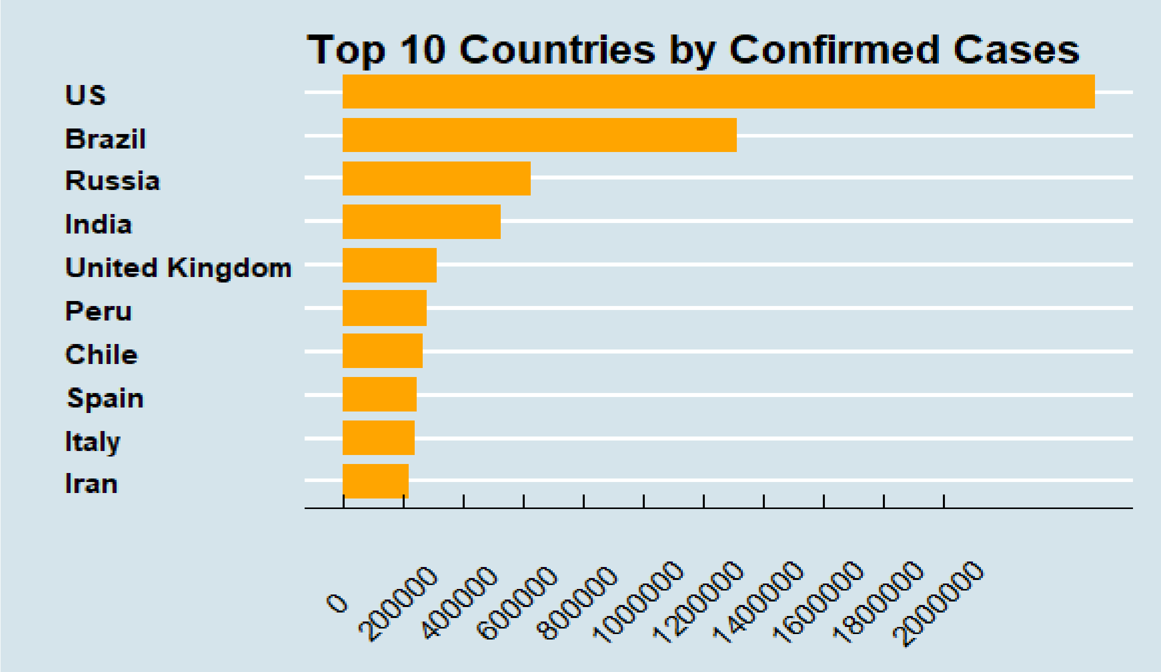
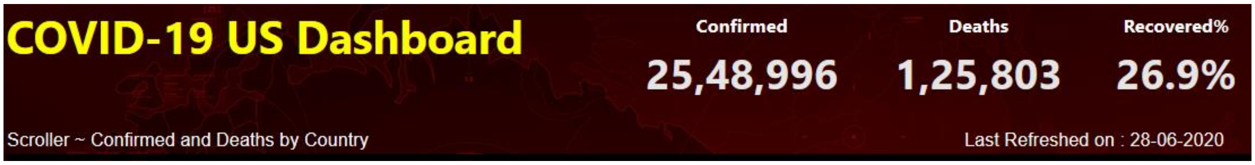


Figure 3 Top 10 countries by confirmed cases

Top10_Deaths Cases by Country:

USA has most Death cases in world with 1,25,803 cases till June 28th and continues to be severe in in the upcomming weeks as well.. after that Brazil, UK, Italy is in 2nd, 3rd, 4th places.



COVID-19 US Dashboard

Confirmed	Deaths	Recovered%
25,48,996	1,25,803	26.9%

Scroller ~ Confirmed and Deaths by Country

Last Refreshed on : 28-06-2020

```
Top10_Deaths <- TodayData %>% select(Country,date,deaths)%>%
  arrange(desc(deaths)) %>% head(10) %>%
  ggplot(aes(x = reorder(`Country`,deaths), y = deaths )) +
  geom_bar(stat = "identity", fill  = "red", width = 0.8) +
  theme_economist() +
  scale_y_continuous(breaks = seq(0, 220000, by = 20000)) +
  coord_flip() +
  labs(x = "", y = "", title = "Top 10 Countries by Death Cases") +
  theme(axis.text.x = element_text(angle = 45)) +
   theme(axis.title = element_text(size = 14, colour = "black"),
         axis.text.y = element_text(size = 11, face = "bold"))
```

Here is a list of the most affected countries in terms of death cases

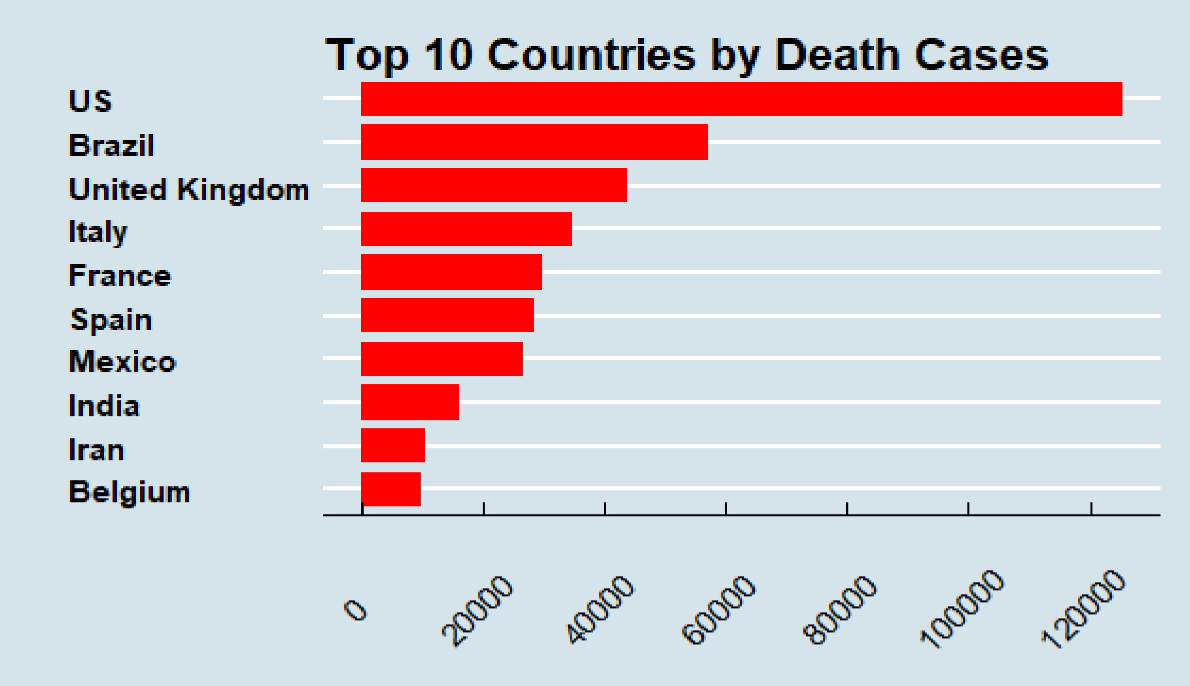


Figure 4 Top 10 countries by deaths

Top10_Recovered Cases by Country

Brazil has most recovered cases in world till June 28th after that USA, Russia, India is in 2nd, 3rd, 4th places.

```
Top10_Recovered <- TodayData %>% select(Country,date,recovered)%>%
  arrange(desc(recovered)) %>% head(10) %>%
  ggplot(aes(x = reorder(`Country`,recovered), y = recovered )) +
  geom_bar(stat = "identity", fill  = "green", width = 0.8) +
  theme_economist() +
  scale_y_continuous(breaks = seq(0, 2200000, by = 50000)) +
  coord_flip() +
  labs(x = "", y = "", title = "Top 10 Countries by Recovered Cases") +
  theme(axis.text.x = element_text(angle = 45)) +
   theme(axis.title = element_text(size = 14, colour = "black"),
         axis.text.y = element_text(size = 11, face = "bold"))
```

Here is a list of the most affected countries in terms of death cases

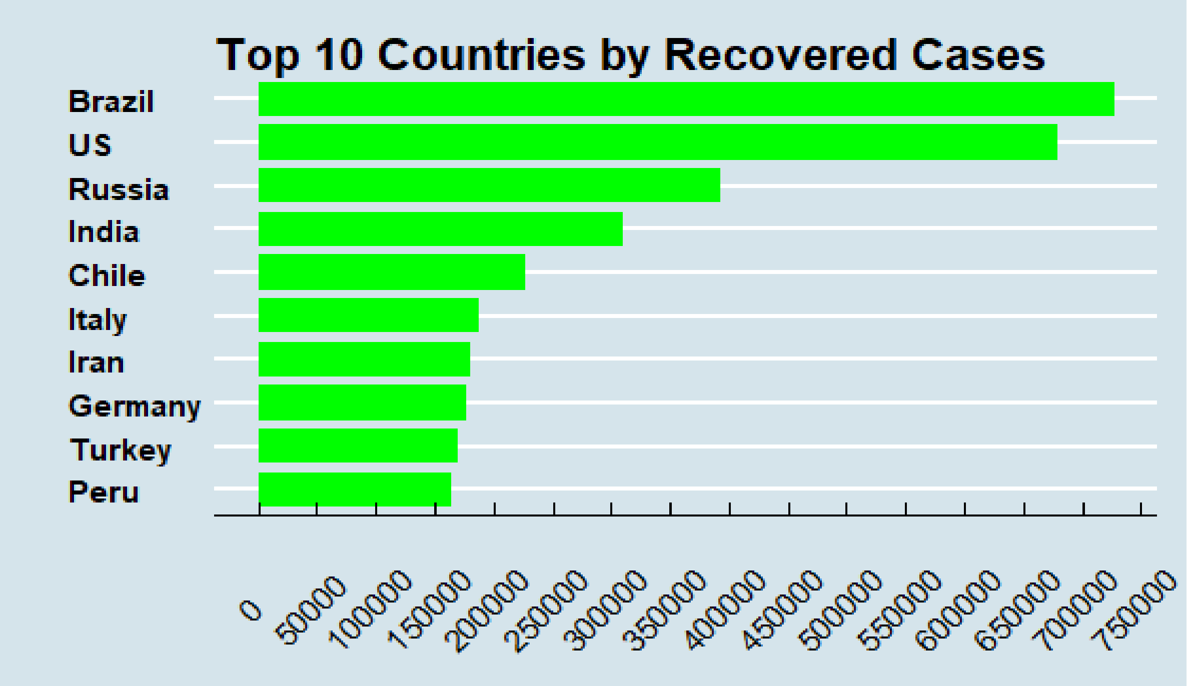
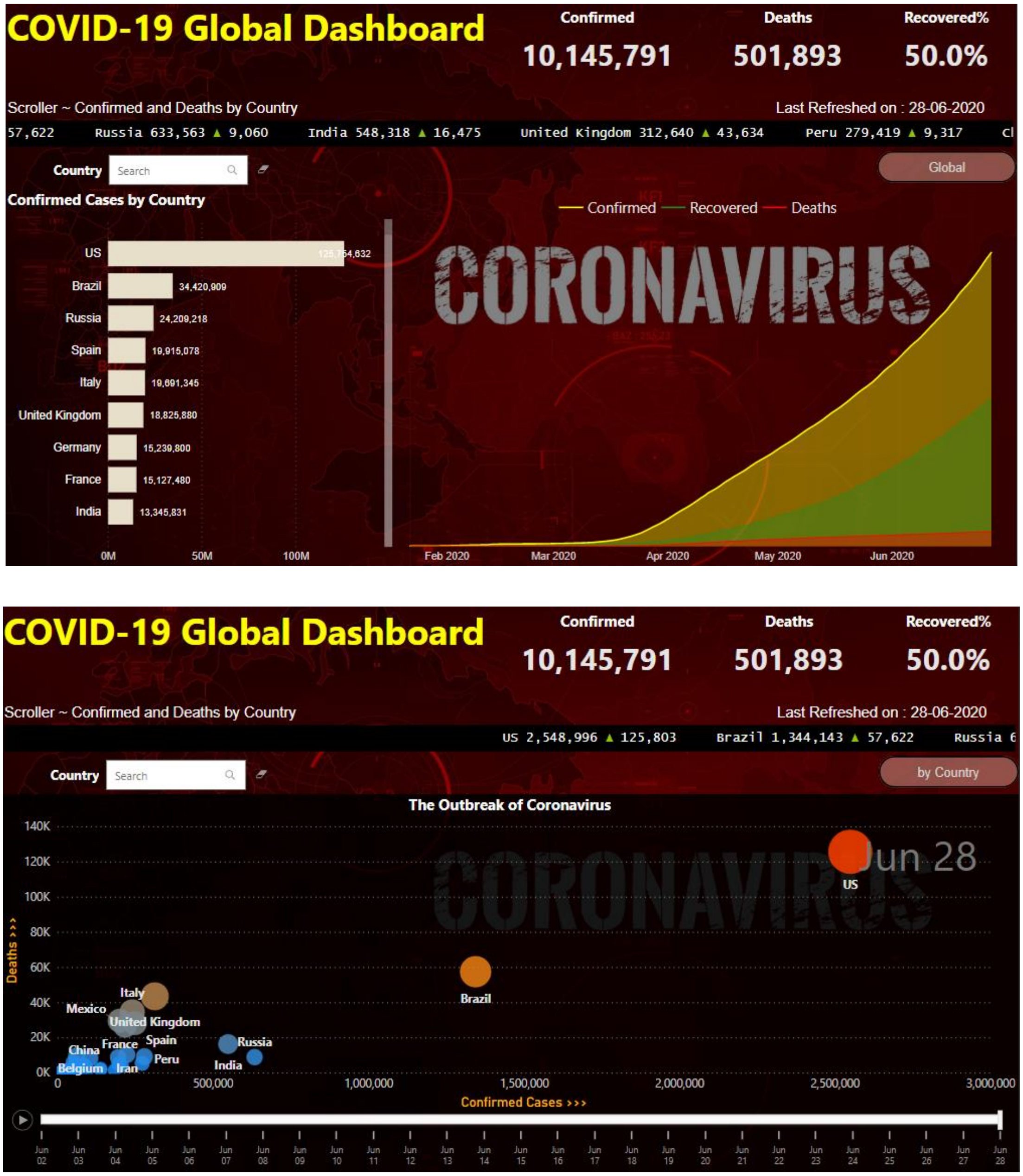


Figure 5 Top 10 countries by recovered cases

3.2.1 POWER BI

We also explore time-series data using visual data analysis to provide a clear and understandable outcome of this extreme outbreak of COVID-19 using Power BI Live dashboard. We have created this dashboard and given awareness of how COVID 19 spread around the globe from 22 January 2020 to till date; it allows individuals to grasp the epidemiological essence of COVID-19.

Dashboard link



COVID-19 Global Dashboard

Confirmed 10,145,791 | Deaths 501,893 | Recovered% 50.0%

Scroller ~ Confirmed and Deaths by Country

Last Refreshed on : 28-06-2020

57,622 Russia 633,563 ▲ 9,060 India 548,318 ▲ 16,475 United Kingdom 312,640 ▲ 43,634 Peru 279,419 ▲ 9,317 Cl

COVID-19 Global Dashboard

Confirmed 10,145,791 | Deaths 501,893 | Recovered% 50.0%

Scroller ~ Confirmed and Deaths by Country

Last Refreshed on : 28-06-2020

US 2,548,996 ▲ 125,803 Brazil 1,344,143 ▲ 57,622 Russia 6

Figure 6 Snapshot of power BI dashboard

3.3 TIME SERIES FORECASTING

Time series forecasting is performing forecasting techniques on a time series data. In other words, it is the prediction over time. A time series is a set of observations generated sequentially with time on a single variable or it is indexed by time.

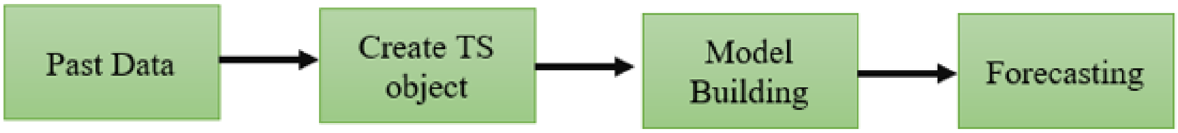


Figure 7 Block diagram of ts modelling

Forecasting is by far the most important and frequently used application of predictive analytics. Inaccurate forecasting can have significant impact on both top line and bottom line of an organization. For example, non-availability of product in the market can result in customer dissatisfaction, whereas too much inventory can erode the organization’s profit. So, forecasting accuracy is very important.

3.3.1PROPHET

Prophet is a forecasting tool developed by Sean J Taylor & Ben Letham from Facebook, they open-sourced it in 2017.This package is available in R & Python, analysts across the world consider prophet as a boon for forecasting. Prophet is mainly used for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well.

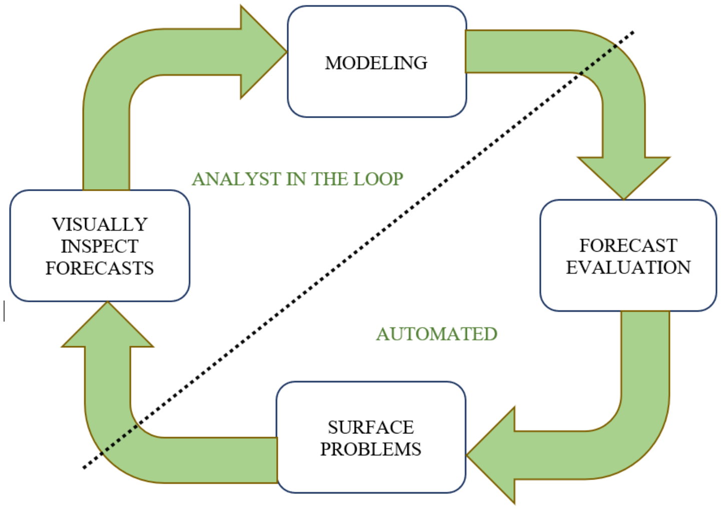


Figure 8 Prophet model working

So, from the above block diagram, it is clear that the analysts need to only build the model and visually inspect the forecasts, the forecast evaluation and surface problems are automated by the prophet. In this project, we are performing a time series analysis by building a prophet forecast model. As we discussed earlier, the first step to perform time series forecasting after data cleaning is creating a time series object. Unlike other time series forecasting techniques the Prophet model does not require a time series object. Instead of that, we are creating a data frame consisting of time as one column and the outcome (confirmed/recovered/deaths) variable as another column.

The very first step in time series forecasting is to check whether the time interval of the data set is equal or not. In case the data set has unequal time intervals we need to make it equal otherwise we cannot perform time series analysis. Here we have periods of equal length that is daily data. The second step is to check for missing values, if data has any missing values we need to remove or impute missing values with appropriate values. We have already performed these steps in the data preparation part.

Now to build the prophet model it is required to install and load the prophet and associated packages. We can perform this by using 'install' and 'library' functions in R. Then check the structure and summary statistics of the cleaned data to get a clear understanding of our data set. The outcome variable (that is the variable which we are going to forecast) must be numeric. So, we need to check the class of that variable and also make sure that time variable is in ‘Date’ format. If the outcome variable is non numeric such as character we should transform it into numeric. This can be performed using nested function in R, as.numeric(as.character(“variable”)).

To forecast the number of confirmed cases in the upcoming days, we need to create a new data frame having the date and confirmed cases as the column values. In our data set we have past confirmed cases and corresponding reported dates. Now we have our historical data frame and we can fit a prophet model to this historical data frame. We can do this by calling the prophet () function and using our prepared data frame as input. The next step is to create a future data frame, here we are making predictions for future dates. The built-in helper function make_future_data frame of prophet package will perform this. This function lets us specify the frequency and number of periods we would like to forecast into the future. By default, the frequency of this data frame is set to days. Since we are using daily data, we will leave `freq’ at its default and set the ‘periods’ argument to 40, indicating that we would like to forecast 40 days into the future. That is we have created a future data frame to predict the number of confirmed cases in the next 40 days.

Now, we have our data frame ready to make predictions for each row in the future data frame. The predict () function will perform this role. We can see the Prophet has created a new data frame assigned to the forecast variable that contains the forecasted values for future dates under a column called `yhat’, as well as uncertainty intervals and components for the forecast. We can visualize the forecast using the prophet’s built-in `plot’ helper function. To visualise the plot components one can use the prophet_plot_components () function. As we know the time series data has various components such as seasonality (yearly, monthly, weekly), level, trend, noise (irregularity, error). We can observe time series components by just calling the above-mentioned function. Here, in this dataset, since the time is less than 1 year, that is the prophet will automatically disable the yearly seasonality. This eases the job of analysts, they do not need to work for seasonal decomposition like in exponential and moving average method.

Global Confirmed Cases – Forecasting

Given below is the R code for forecasting confirmed Covid-19 cases in the world. The most important step is class of date column must be in date format, it should not be in character or numeric type. The first step here is to converting the date column into default date format using the lubridate package. And grouping the confirmed, recovered and deaths columns country wise and creating a new variable.

```r
COVID19GlobalData$date<- ymd(COVID19GlobalData$date)
  class(COVID19GlobalData$date)
  str(COVID19GlobalData)

  #Global Confirmed Cases Forecasting
  coviddata.world <- COVID19GlobalData %>% group_by(date) %>%
    summarise(Country='World',confirmed=sum(confirmed,na.rm = 1),deaths=sum(deaths,na.rm = 1),recovered=sum(recovered,na.rm = 1))

  qplot(date,confirmed,data=coviddata.world,main ='COVID-19 Global Confirmed Cases') +
      scale_y_continuous(breaks = seq(0, 10000000, by = 900000))
```

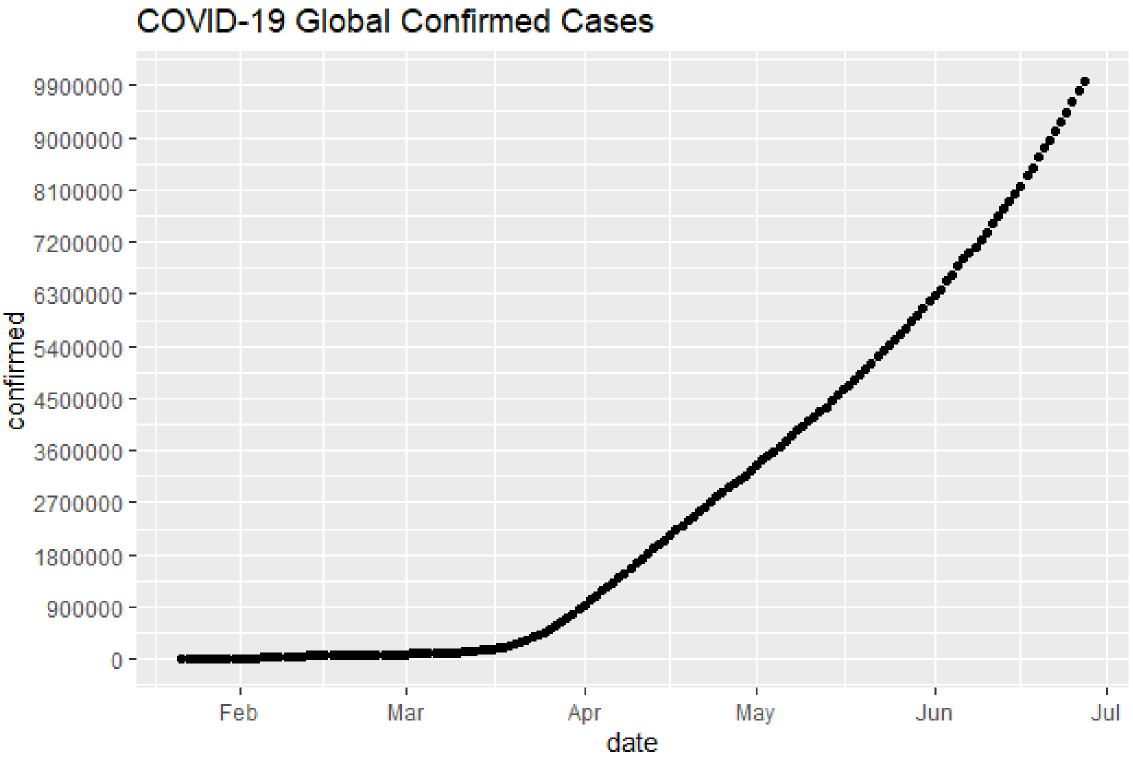


Figure 9 Plot of Global confirmed cases

We can clearly see that the total number of infected cases has reached above 1 crore, the situation is getting worse as the virus spreads exponentially.

```
# Confirmed cases Dataframe for modeling
  ds <- coviddata.world$date
  y <- coviddata.world$confirmed
  df <- data.frame(ds,y)

  # Prophet modeling
  m <- prophet(df)
  future <- make_future_dataframe(m,periods = 40)
  forecast <- predict(m,future)

    plot(m,forecast)+
    scale_y_continuous(breaks = seq(0, 17000000, by = 900000))
    dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED - GLOBAL
CONFIRMED CASES")%>%
      dyOptions(maxNumberWidth = 20)
```

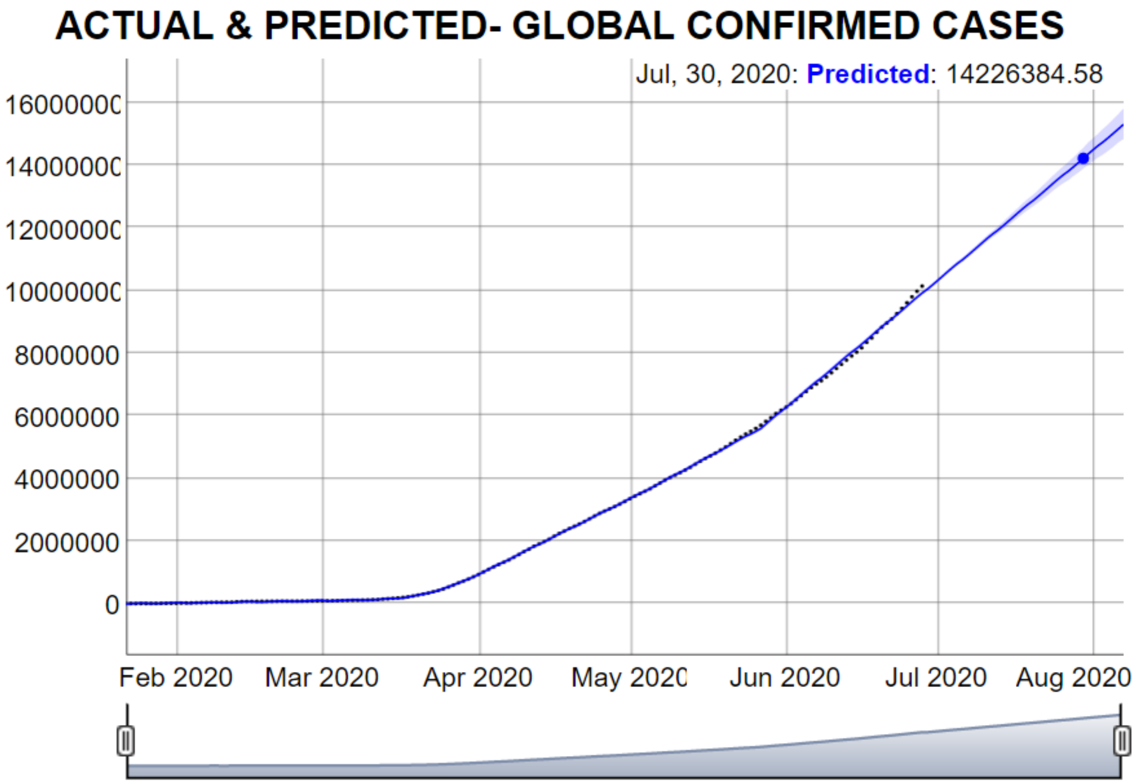


Figure 10 Dyplot of Global actual & predicted confirmed cases

This is the dyplot for global confirmed cases. Here we can see the actual number of covid-19 infected people and predicted number of infected people by dragging the cursor. The prophet model is forecasting that the total number of infected people would be around 14226385 on 30th July 2020.

```
prophet_plot_components(m,forecast)
```

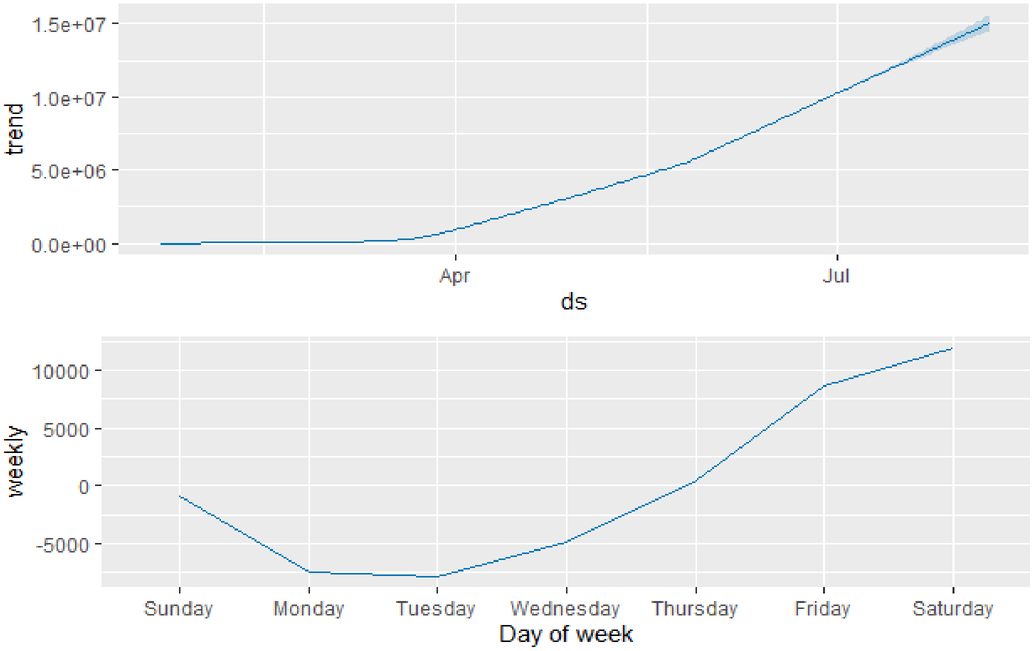


Figure 11 TS components of global data

The above figure shows the components of the time series dataset. It has a trend component and weakly seasonality component. We can see that the trend line is growing exponentially. Since we have short range data here it is showing weekly seasonality only. As per the weekly seasonality component it is meant that more people are infected on Fridays and Saturdays , but we cannot interpret like this because virus does not any days of the week. So this seasonality can be due to some data entry errors or we can say that may be more tests will be conducted on Fridays and Saturdays . There are several external factors that will affect this weekly seasonality component.

```
#Actual vs Predicted plot
predicted <- forecast$yhat[1:158]
actual <- m$history$y

plot(actual,predicted)+
abline(lm(predicted~actual),col='red')

summary(lm(predicted~actual))
Accuracymodel1 <- accuracy((lm(predicted~actual)))
 Accuracymodel1
 str(forecast)
```

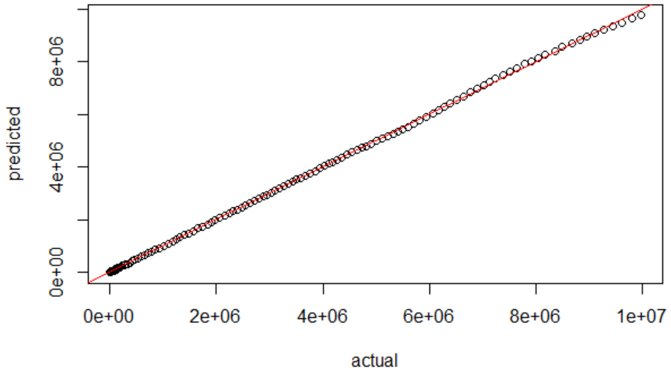


Figure 12 Fit line of global actual & predicted cases

Global Recovered Cases – Forecasting

```
Coviddata.world
ds <- coviddata.world$date
y <- coviddata.world$recovered
df <- data.frame(ds,y)
m <- prophet(df)
future <- make_future_dataframe(m,periods = 40)
forecast <- predict(m,future)
plot(m,forecast)
dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED GLOBAL RECOVERED CASES ")%>%dyOptions(maxNumberWidth = 20)
```

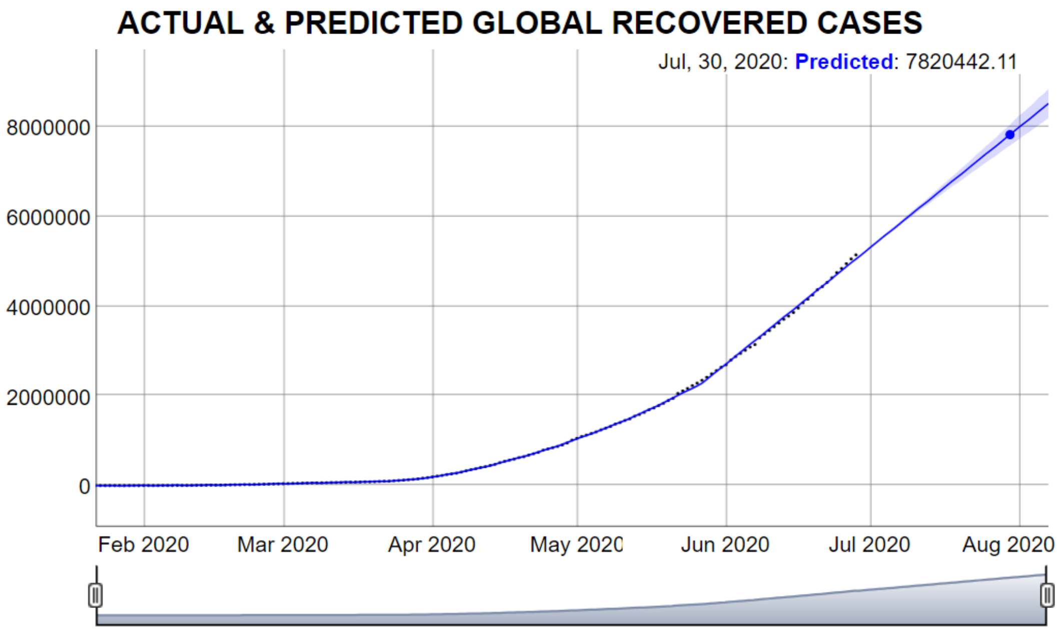


Figure 13 Dyplot of global recovered cases

Global Deceased Cases – Forecasting

```r
ds <- coviddata.world$date
y <- coviddata.world$deaths
df <- data.frame(ds,y)
m <- prophet(df)
future <- make_future_dataframe(m,periods = 40)
forecast <- predict(m,future)
plot(m,forecast)
dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED GLOBAL DECEASED CASES")%>%dyOptions(maxNumberWidth = 20)
```

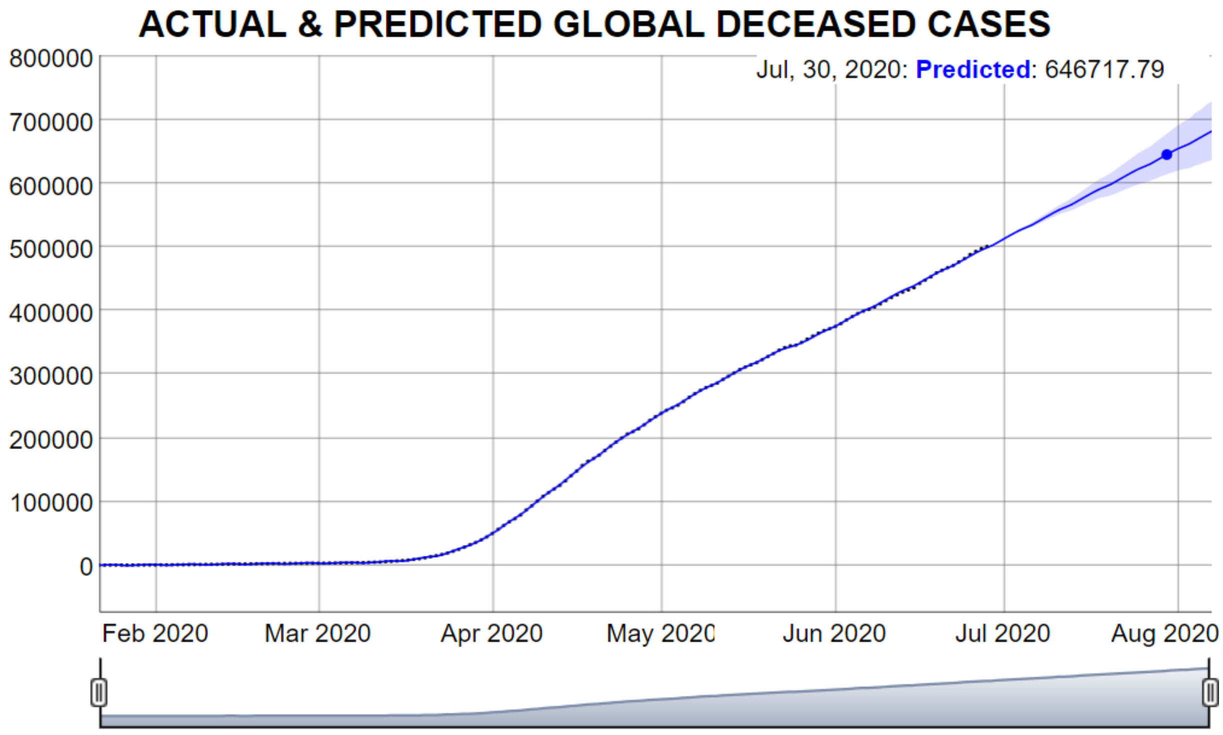


Figure 14 Dyplot of global deceased cases

India Confirmed Cases – Forecasting

```r
coviddata.India <- COVID19GlobalData%>%filter(Country=="India")
  qplot(date,confirmed,data=coviddata.India,main = 'COVID-19 CONFIRMED CASES IN INDIA') +
    scale_y_continuous(breaks = seq(0, 900000, by = 50000))
```

```r
# New Dataframe
  ds <- coviddata.India$date
  y <- coviddata.India$confirmed
  df <- data.frame(ds,y)
  df
  #Using Prophet
  m <- prophet(df)
  future <- make_future_dataframe(m,periods = 40)
  forecast <- predict(m,future)
   # Plot
   plot(m,forecast)
   dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED CONFIRMED CASES IN INDIA") %>%dyOptions(maxNumberWidth = 20)
   prophet_plot_components(m,forecast)
```

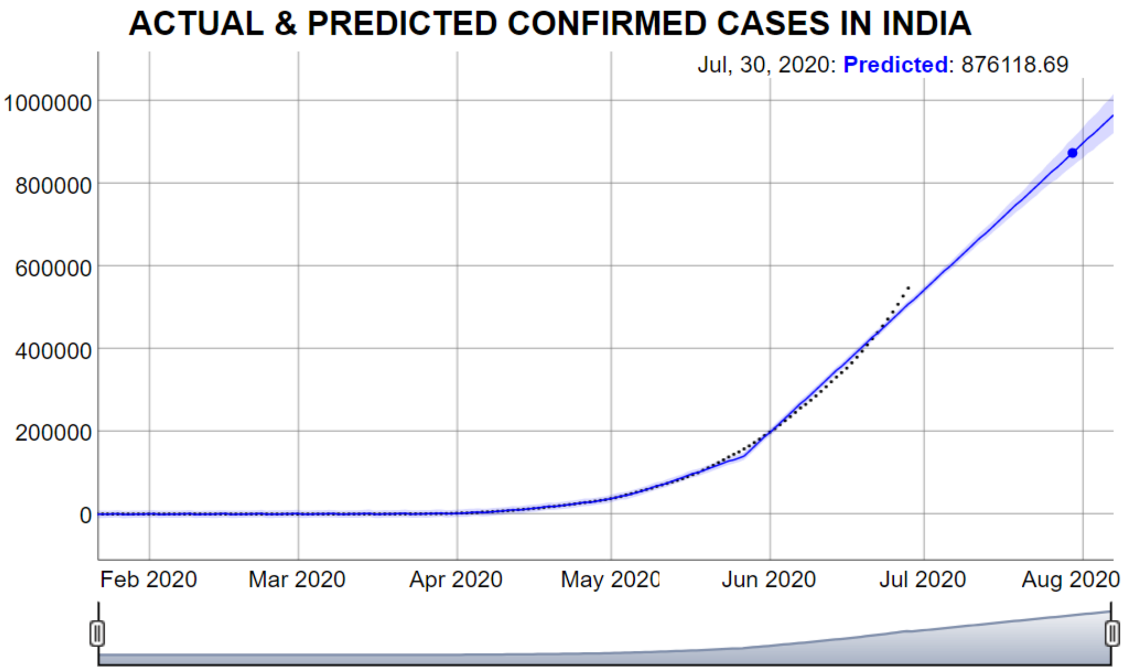


Figure 15 Dyplot of actual and predicted confirmed cases in India

India Recovered Cases – Forecasting

```r
ds <- coviddata.India$date
y <- coviddata.India$recovered
df <- data.frame(ds,y)
m <- prophet(df)
future <- make_future_dataframe(m,periods = 40)
forecast <- predict(m,future)
plot(m,forecast)
dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED RECOVERED CASES IN INDIA")%>%dyOptions(maxNumberWidth = 20)
```

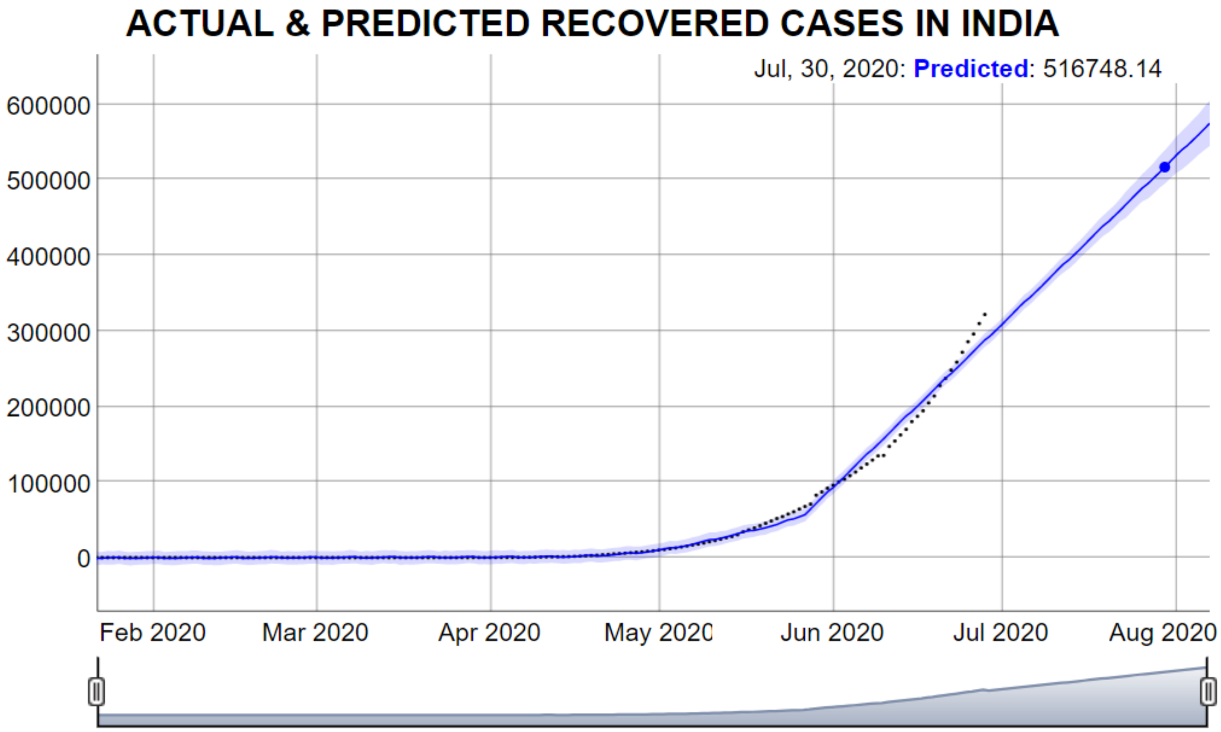


Figure 16 Dyplot of actual & predicted recovered cases in India

India Deceased Cases – Forecasting

```
ds <- coviddata.India$date
y <- coviddata.India$deaths
df <- data.frame(ds,y)
m <- prophet(df)
future <- make_future_dataframe(m,periods = 40)
forecast <- predict(m,future)
plot(m,forecast)
dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED DECEASED CASES IN INDIA")%>%dyOptions(maxNumberWidth = 20)xNumberWidth = 20)
```

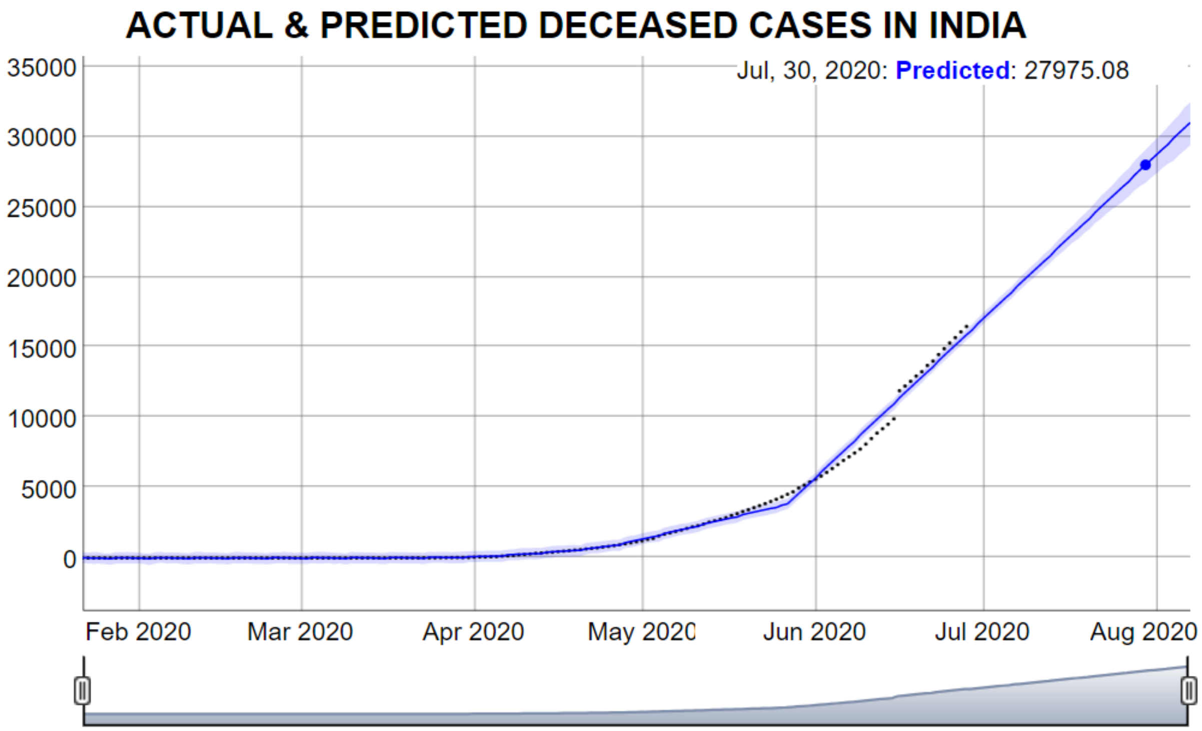


Figure 17 Dyplot of actual and predicted deceased cases in India

US Confirmed Cases – Forecasting

```r
coviddata.US <- COVID19GlobalData%>%filter(Country=="US")
  qplot(date,confirmed,data=coviddata.US,main = 'COVID-19 CONFIRMED CASES IN THE US')
```

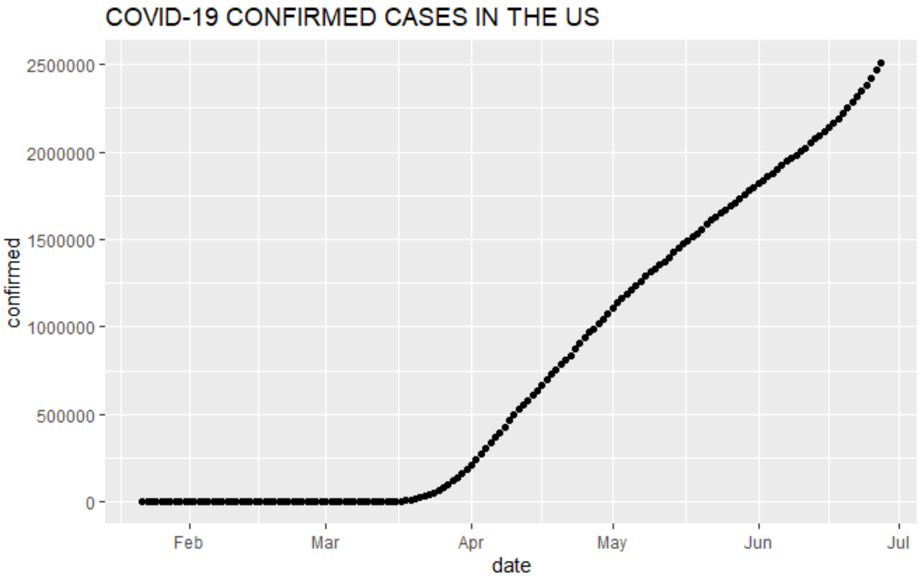


Figure 18 Plot of confirmed cases in the US

```r
# New Dataframe
  ds <- coviddata.US$date
  y <- coviddata.US$confirmed
  df <- data.frame(ds,y)
  df
  #Using Prophet
  m <- prophet(df)
  future <- make_future_dataframe(m,periods = 40)
  forecast <- predict(m,future)
  # Plot
  plot(m,forecast)
  dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED CONFIRMED CASES IN THE US") %>%dyOptions(maxNumberWidth = 20)
```

```r
  predicted <- forecast$yhat[1:158]
  actual <- m$history$y
  plot(actual,predicted)
  abline(lm(predicted~actual),col='red')
  AccuracyUS <- accuracy((lm(predicted~actual)))
  AccuracyUS
  summary(lm(predicted~actual))
  str(forecast)
```

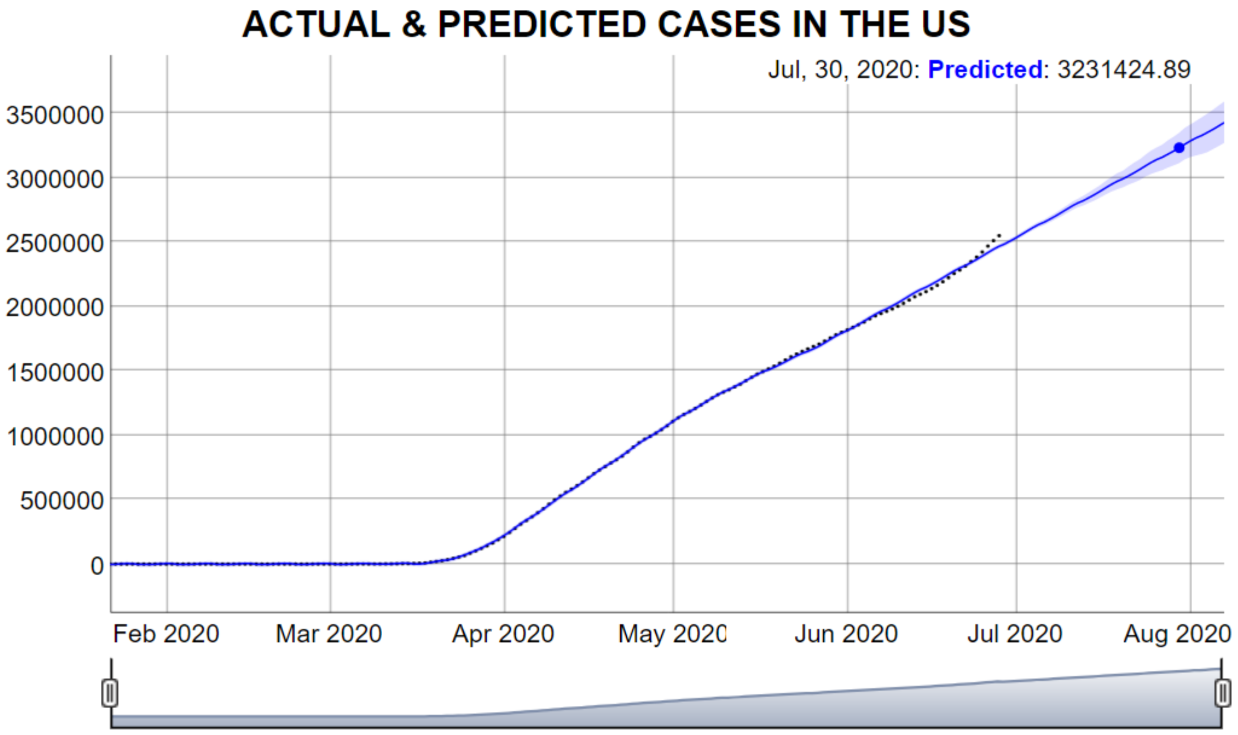


Figure 19 Dyplot of actual & predicted confirmed cases in the US

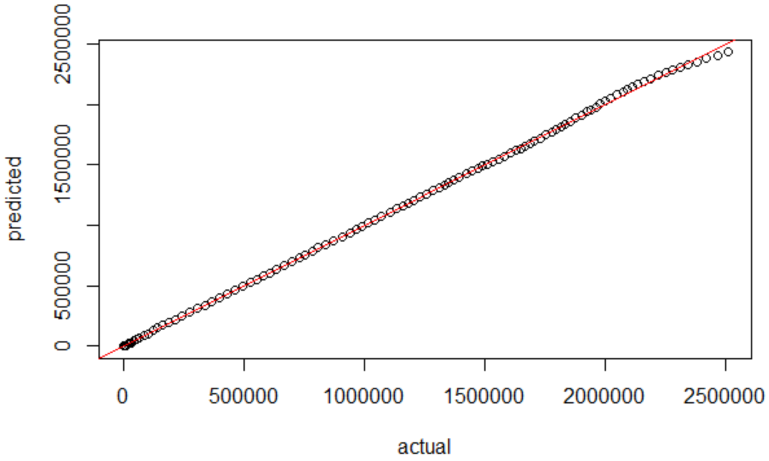


Figure 20 Fit line of confirmed cases in the US

US Recovered Cases – Forecasting

```
ds <- coviddata.US$date
y <- coviddata.US$recovered
df <- data.frame(ds,y)
df
m <- prophet(df)
future <- make_future_dataframe(m,periods = 40)
forecast <- predict(m,future)
# Plot
plot(m,forecast)
dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED RECOVERED CASES IN THE US")%>%dyOptions(maxNumberWidth = 20)
```

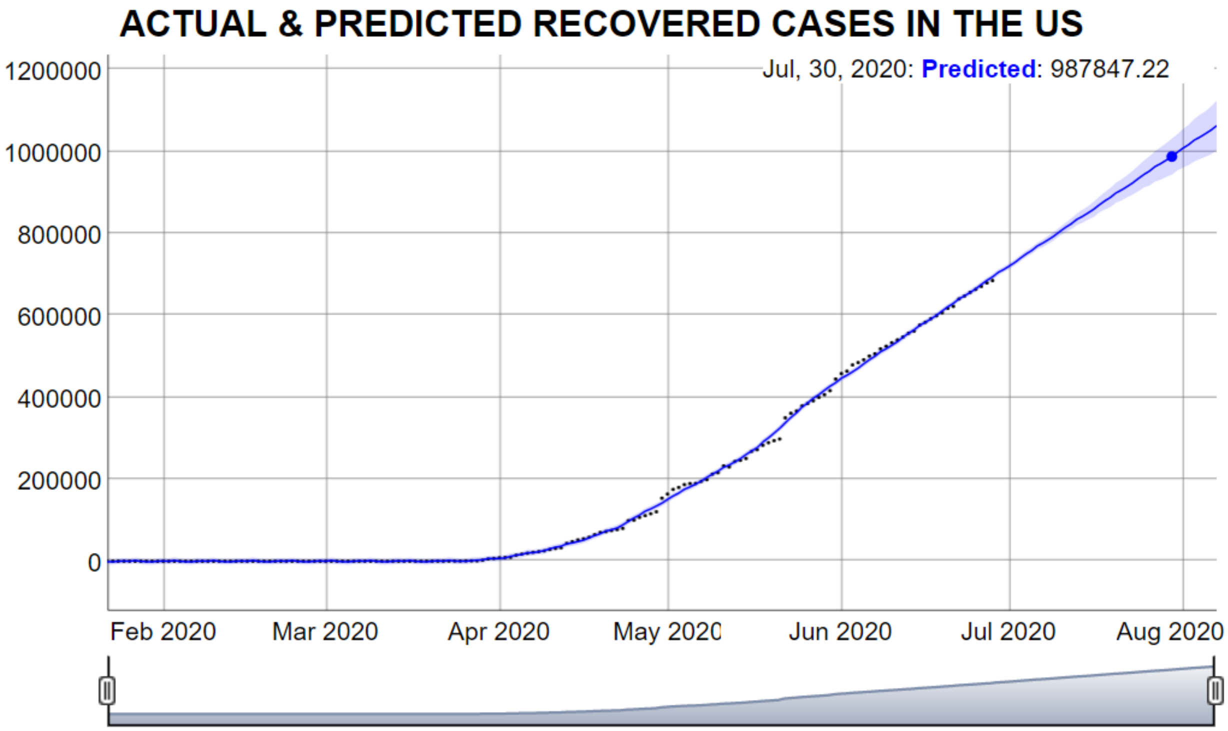


Figure 21 Dyplot of actual & predicted recovered cases in the US

US Deceased Cases – Forecasting

```
ds <- coviddata.US$date
y <- coviddata.US$deaths
df <- data.frame(ds,y)
df
m <- prophet(df)
future <- make_future_dataframe(m,periods = 40)
forecast <- predict(m,future)
plot(m,forecast)
```

```
dyplot.prophet(m,forecast,main="ACTUAL & PREDICTED DECEASED CASES IN
THE US")%>%dyOptions(maxNumberWidth = 20)
```

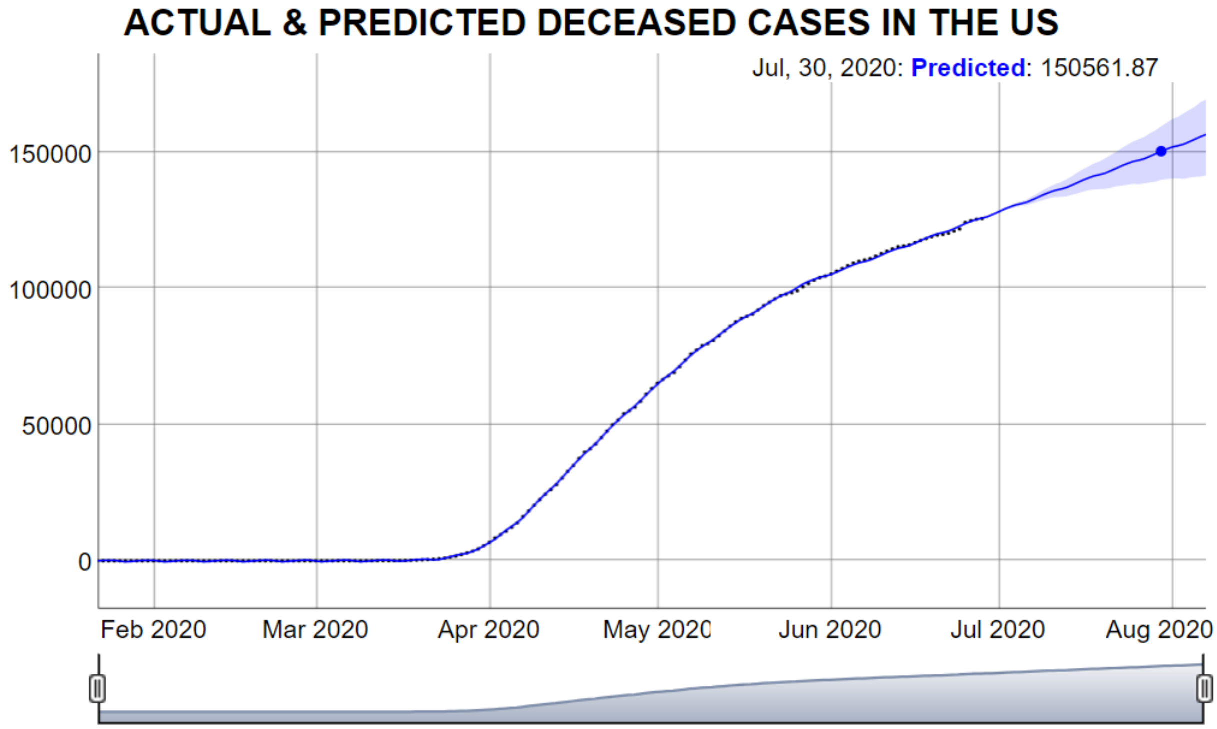


Figure 22 Dyplot of actual & predicted deceased cases in the US

3.3.2 ARIMA

An autoregressive integrated moving average (ARIMA) model is a generalization of an autoregressive moving average (ARMA) model. Both of these models are fitted to time series data either to better understand the data or to predict future points in the series (forecasting). ARIMA models are applied in some cases where data show evidence of non-stationarity, where an initial differencing step (corresponding to the "integrated" part of the model) can be applied one or more times to eliminate the non-stationarity.

The AR part of ARIMA indicates that the evolving variable of interest is regressed on its own lagged (i.e., prior) values. The MA part indicates that the regression error is actually a linear combination of error terms whose values occurred contemporaneously and at various times in the past. The I (for "integrated") indicates that the data values have been replaced with the difference between their values and the previous values (and this differencing process may have been performed more than once). The purpose of each of these features is to make the model fit the data as well as possible. Non-seasonal ARIMA models are generally denoted ARIMA (p, d, q) where parameters p, d, and q are non-negative integers.

- p is the order (number of time lags) of the autoregressive model.
- d is the degree of differencing (the number of times the data have had past values subtracted).
- q is the order of the moving-average model.

Seasonal ARIMA models are usually denoted ARIMA (p, d, q) (P, D, Q) m. Where, m refers to the number of periods in each season. When two out of the three terms are zeros, the model may be referred to based on the non-zero parameter, dropping "AR", "I" or "MA" from the acronym describing the model. For example, ARIMA (1,0,0) is AR(1), ARIMA(0,1,0) is I(1),and ARIMA(0,0,1) is MA(1).

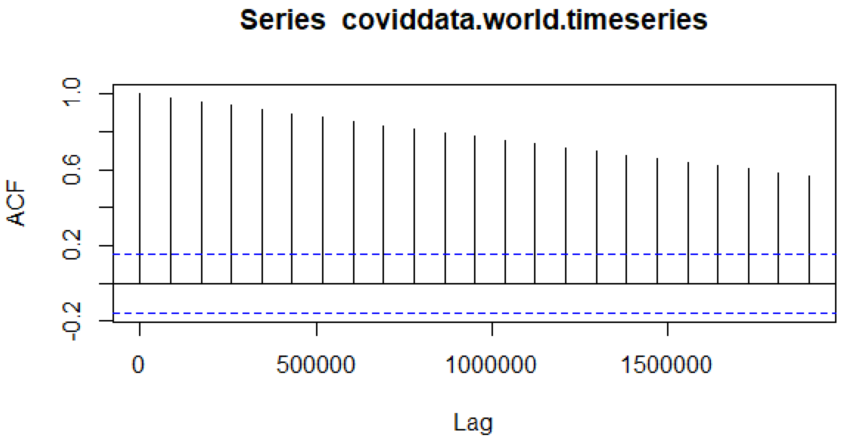
The ARIMA model includes autoregressive (AR) model, moving average (MA) model, and seasonal autoregressive integrated moving average (SARIMA) model. The Augmented Dickey-Fuller (ADF) unit-root test helps in estimating whether the time series is stationary. Log transformation and differences are the preferred approaches to stabilize the time series. Seasonal and nonseasonal differences were used to stabilize the term trend and periodicity.

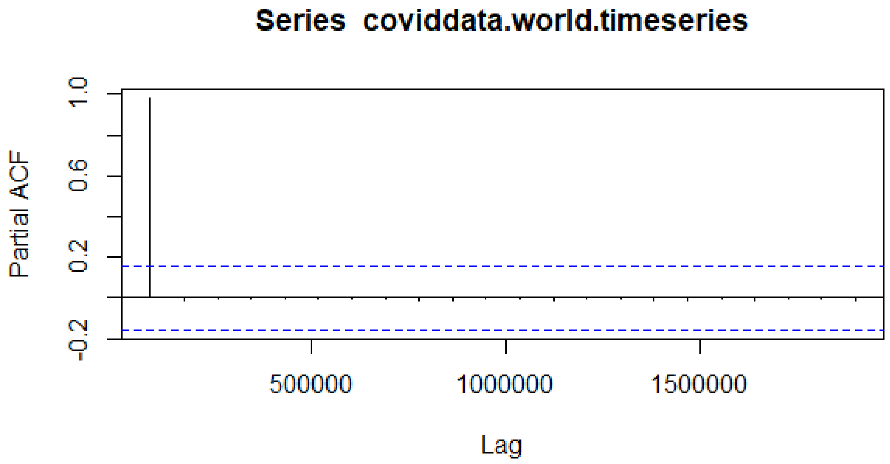
Parameters of the ARIMA model were estimated by autocorrelation function (ACF) graph and partial autocorrelation (PACF) correlogram. To determine the prevalence of COVID-19, ARIMA (1,0,1) was selected as the best ARIMA model, while ARIMA (1,0,1) was selected as the best ARIMA model for determining the incidence of COVID-19.Auto.arima function was used to perform statistical analysis on the prevalence and incidence datasets. A previous study was considered as reference for the methodology of the analysis.

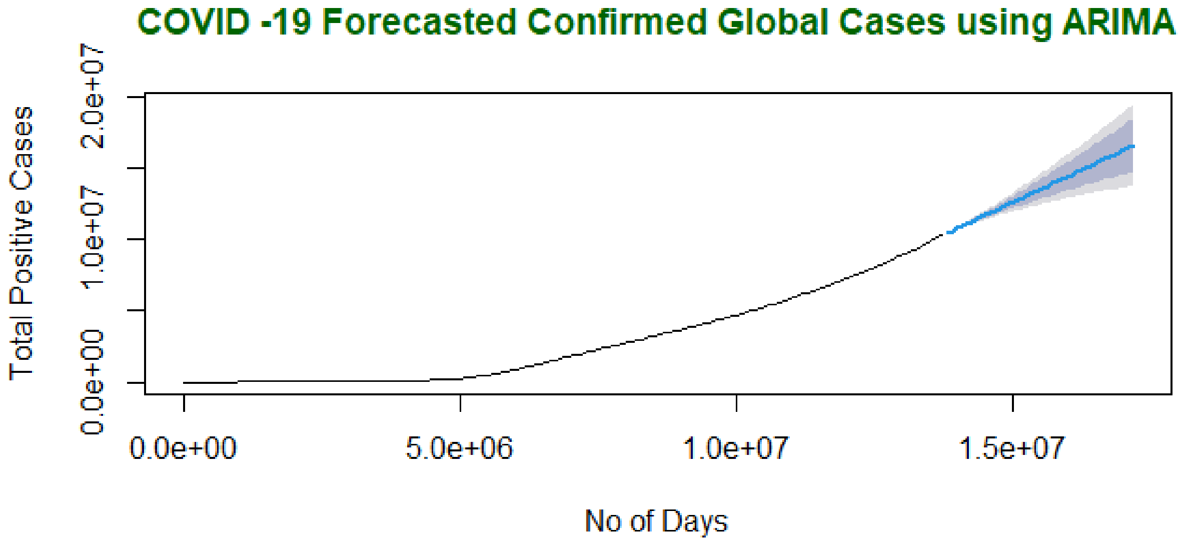
Logarithmic transformation was performed to evaluate the influence of seasonality on the forecast. The correlogram reporting the ACF and PACF showed that both prevalence and incidence of COVID-19 are not influenced by the seasonality. The forecast of prevalence and incidence data with relative 95% confidence intervals are reported.

ARIMA forecasts on its previous past values

```
#ARIMA Model---------------------------
  coviddata.world.timeseries <- xts(coviddata.world$confirmed,order.by = coviddata.world$date)
  acf(coviddata.world.timeseries)
  pacf(coviddata.world.timeseries)

  fit_arima <- arima(coviddata.world.timeseries,order=c(0, 2, 0))
  # Next 40 days forecasted values
  forecast(fit_arima, 40)
  # plotting the graph with Next 40 Days forecasted values
   plot(forecast(fit_arima, 40), xlab ="No of Days",
        ylab ="Total Positive Cases",
        main ="COVID -19 Forecasted Confirmed Global Cases using ARIMA", col.main ="darkgreen")
   AIC(fit_arima)
 BIC(fit_arima)
   summary(fit_arima)
   accuracy(fit_arima)

 #AUTO ARIMA--------------------
 #creating TS object for ARIMA Model

   coviddata.world.timeseries <- xts(coviddata.world$confirmed,order.by = coviddata.world$date)
   fit_auto.arima <- auto.arima(coviddata.world.timeseries)
   # Next 40 days forecasted values
   forecast(fit_auto.arima, 40)
   # plotting the graph with Next 40 Days forecasted values
   plot(forecast(fit_auto.arima, 40), xlab ="No of Days",
        ylab ="Total Positive Cases",
        main ="COVID -19 Forecasted Confirmed Global Cases using AUTO ARIMA", col.main ="darkgreen")
   acf(coviddata.world.timeseries)
   pacf(coviddata.world.timeseries)

   summary(fit_auto.arima)
   accuracy(fit_auto.arima)
```







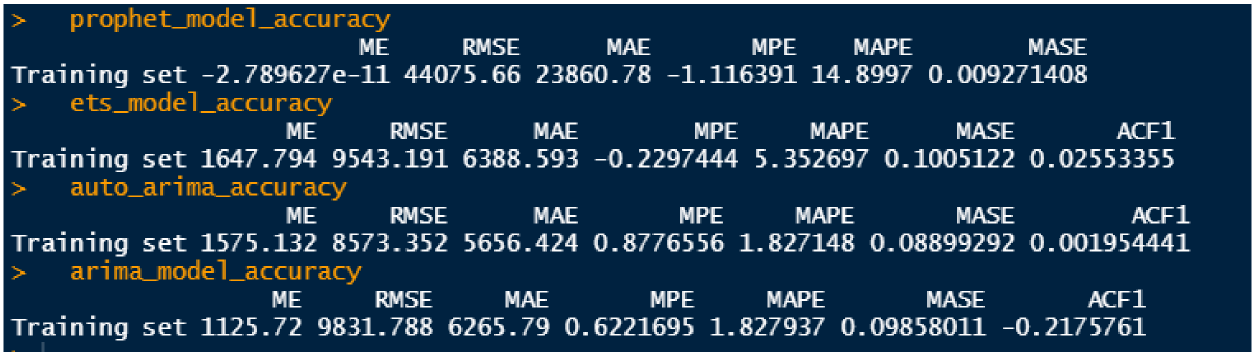
3.3.3 COMPARISON OF TS MODELS

There are many forecasting techniques developed based on different logics such as Prophet, moving average, exponential smoothing and ARIMA are used for forecasting before selecting the best model. The model selection may depend on the chosen forecasting accuracy measure. In this section we are trying to compare the accuracy of time series models which we have built. The frequently used forecasting accuracy measures are Mean absolute error, mean absolute percentage error, mean squared error, Root mean square error, AIC and BIC.

MAE is the average absolute error and should be calculated on the validation data set. MAPE is the average of absolute percent error. MAPE is one of the popular forecasting accuracy measures used by analysts since it expresses the average error in percentage terms and is easy to interpret. Since MAPE is dimensionless it can be used for comparing different models with varying scales. MSE is the average of squared error calculated over the validation data set. Lower MSE implies better prediction. However, it depends on the range of time-series data. RMSE is the square root of mean square error. RMSE and MAPE are the two most popular accuracy measures of forecasting. AIC & BIC are measures of distance from the actual values to the forecasted values.

Here we are comparing prophet model, arima model, automated arima model and exponential smoothing model. To compare the accuracy metrics, we are using accuracy () function from the forecast package. We have got lower MAPE values for all the four models.

```
#---------------------------COMPARISON OF TIME SERIES MODELS------------
#GLOBAL CONFIRMED CASES
#-------------------------MODEL 1 : PROPHET----------------------------
    prophet_model_accuracy <- accuracy((lm(predicted~actual)))
  prophet_model_accuracy

#-------------------------MODEL 2 : EXPONENTIAL SMOOTHING--------------
#CREATING TS OBJECT
  coviddata.world.timeseries <- xts(coviddata.world$confirmed,order.by = coviddata.world$date)
    ets(coviddata.world.timeseries)
  ets_model_accuracy <- accuracy( ets(coviddata.world.timeseries))
  ets_model_accuracy

 #-------------------------MODEL 3 : AUTO ARIMA MODEL------------------
    auto.arima(coviddata.world.timeseries)
  auto_arima_accuracy <- accuracy(auto.arima(coviddata.world.timeseries))
  auto_arima_accuracy

 #---------------------------MODEL 4 : ARIMA---------------------------
    arima(coviddata.world.timeseries, order=c(0, 2, 0))
  arima_model_accuracy <- accuracy(arima(coviddata.world.timeseries, order=c(0, 2, 0)))

 #-----Comparing MAPE,ACF,RMSE,MASE VALUES OF EACH MODELS--------------
    prophet_model_accuracy
  ets_model_accuracy
  auto_arima_accuracy
  arima_model_accuracy
```



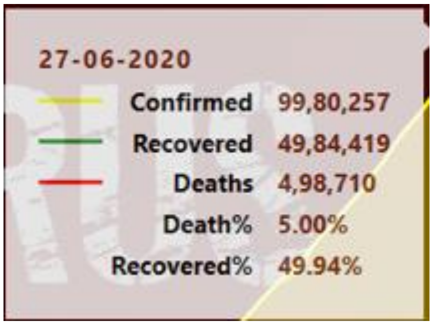
```
>   prophet_model_accuracy
                        ME     RMSE      MAE       MPE    MAPE        MASE
Training set -2.789627e-11 44075.66 23860.78 -1.116391 14.8997 0.009271408
>   ets_model_accuracy
                   ME     RMSE      MAE        MPE     MAPE      MASE       ACF1
Training set 1647.794 9543.191 6388.593 -0.2297444 5.352697 0.1005122 0.02553355
>   auto_arima_accuracy
                   ME     RMSE      MAE       MPE     MAPE       MASE        ACF1
Training set 1575.132 8573.352 5656.424 0.8776556 1.827148 0.08899292 0.001954441
>   arima_model_accuracy
                  ME     RMSE     MAE       MPE     MAPE       MASE       ACF1
Training set 1125.72 9831.788 6265.79 0.6221695 1.827937 0.09858011 -0.2175761
```

4.SUMMARY

The novel corona virus spread so rapidly that it has changed the rhythm of the globe. Whether from the perspective of a single country or multilateral levels, the solidity of international relations has been put under test. The most obvious consequences include economic recession, a crisis of global governance, trade protectionism and increasing isolationist sentiment. People-to-people, cultural and travel exchanges have all been restricted. Nonetheless, this is just a tip of the iceberg.

We have analyzed, visualized, modeled and forecasted the Covid-19 data set. The results and insights we have got from the project is as follows:

- From the EDA, we could understand that around 49.94% of total corona infected people has recovered. And 5% overall death rate.
- We could understand that US is the worst affected country with highest mortality rate.
- In the Power BI dashboard, we can clearly see the confirmed, recovered, deaths, percentage of deceased cases and percentage of recovered cases country wise. It helps in analyzing and studying the spread of this virus.



27-06-2020

Confirmed	99,80,257
Recovered	49,84,419
Deaths	4,98,710
Death%	5.00%
Recovered%	49.94%

- We successfully build time series models and forecasted the future values. The forecasted values is as shown in the table below:

TIME SERIES FORECASTED VALUES: 30th July 2020			
COUNTRY	CONFIRMED	RECOVERED	DECEASED
GLOBAL	15608413	7820442	646718
INDIA	876119	516748	27975
US	3231425	987847	150562

Table 2 Forecasted value matrix

As per our time series models these are the approximate number of confirmed, recovered, and deceased cases by the end of July 2020. These are huge numbers and we should take necessary precautions to stop the spread and to reduce the number.

TOTAL NO: OF CONFIRMED CASES				
Date	Prophet	Arima	Auto Arima	Actual
29-06-2020	10039570	10311325	10316875	1024460
30-07-2020	14226385	15608413	15831730	---

Table 3 Comparison of forecasted values

The above table shows the forecasted confirmed cases by three models. From this it is clear that ARIMA forecasting is closest to the actual confirmed case. There are several external factors that will affect this predicted values such as increasing the number of tests, maintaining proper social distancing, extending the lockdown, proper use of personal protection measures (masks, sanitizer), increasing covid specialized hospitals, giving proper treatment to the infected people. By doing all the mentioned methods we can reduce the number of infected cases, reduce the death rates and we can win this health war.

5.INFERENCE

According to the current data, Almost every region in the world is affected by COVID-19.As per our analysis from ARIMA model and PROPHET model, we came to this conclusion that the trend of confirmed cases are still going up but death rates are comparatively low and fertility rate is increasing.

Differences results for the countries have been observed since they have different epidemic exposure dates and social and technological developments such as health policies, preliminary measures and economic levels. In this study, the models, which are established by using the number of COVID-19 pandemic cases of the countries, provide information about the estimated number of cases that may be for the future days. The measures taken by countries such as the individual attitudes of the societies towards the specified measures and the number of virus tests to be performed are factors that may affect the number of cases. Since this study was conducted with the current measures, the forecasts obtained may differ from the number of cases that occur in the future. The more precautions are taken, the fewer the number of cases.

6.KEYWORDS

- COVID-19,
- ARIMA Model,
- Time Series,
- Short term prediction,
- Dashboard
- COVID-19 Outbreak
- Forecasting
- Lock down

7.REFERENCES

1. Johns Hopkins University for making the data available for educational and academic research purposes
2. https://www.r-bloggers.com/
3. https://rpubs.com/
4. https://stackoverflow.com/
5. https://facebook.github.io/prophet/
6. https://facebook.github.io/prophet/docs/quick_start.html#r-api
7. World Health Organization (WHO): https://www.who.int/