Exploring Covid-19 trends in India: Vulnerabilities and Mitigation Strategies(A Pilot Study) By Swayamjit Saha

Table of Contents

[Project Goal 2](#_Toc152623450)

[Step 1: Setting Working Directory for Experiment 2](#_Toc152623451)

[Step 2: Installing necessary libraries 2](#_Toc152623452)

[Step 3: Load the necessary libraries 2](#_Toc152623453)

[Step 4: Cleaning the datasets and making it ready for processing 2](#_Toc152623454)

[Read data from CSV file 3](#_Toc152623455)

[Remove blank spaces from column names 3](#_Toc152623456)

[Save the modified data to the existing CSV file 3](#_Toc152623457)

[Libraries and their functions 3](#_Toc152623458)

[Explore the structure of the raw data 3](#_Toc152623459)

[Step 5: Tidying the data 3](#_Toc152623460)

[Tidying the covid\_vaccine\_statewise.csv data 4](#_Toc152623461)

[Finally exploring the structure of the new tidy data 4](#_Toc152623462)

[Save the tidy data to a new CSV file 4](#_Toc152623463)

[Step 6: Creating plots for visualization 4](#_Toc152623464)

[Example 1. Visualizing Male vs. Female Vaccine Doses Administered for a particular date of the month 4](#_Toc152623465)

[Load libraries 4](#_Toc152623466)

[Read data from CSV file 5](#_Toc152623467)

[Convert Date to a Date object 5](#_Toc152623468)

[Specify the particular date (replace ‘2022-01-15’ with your desired date) 5](#_Toc152623469)

[Filter data for the selected date 5](#_Toc152623470)

[Create a bar graph 5](#_Toc152623471)

[Example 2. Time series of Initial Cases in Kerala from 1/30/2020 to 3/1/2020 5](#_Toc152623472)

[Filter data for the specified date range 5](#_Toc152623473)

[Example 3. Time series of First Dose Administered Vaccination Initially in Random states 6](#_Toc152623474)

[Example 4. Graph plot to visualize females vaccination status from West Bengal in the period 27/06/2021 to 07/07/2021 6](#_Toc152623475)

[Example 5. Graph plot to visualize the installation of vaccine sites throughout 2020 to 2021 6](#_Toc152623476)

[Example 6. Graph plot to visualize the state of Odisha in India has made an exponential growth (mostly) in developing vaccine sites for conducting vaccination sessions 7](#_Toc152623477)

[Discussion 7](#_Toc152623478)

[Conclusion 7](#_Toc152623479)

[Future Scope 7](#_Toc152623480)

[References 8](#_Toc152623481)

# Project Goal

We intend to visualize the Covid 19 infection and vaccination patterns across various geographical states of India from the years 2020 through 2021 using the various plots discussed in class. We also preprocess and clean the data by removing any inconsistencies, null values and outliers so that it can be visualized properly. Overall, we provide valuable insights that can inform public health measures, resource allocation, and strategies for managing and controlling the spread of the virus.

We follow the following control map:

A diagram of a model

Description automatically generated

# Step 1: Setting Working Directory for Experiment

Firstly, we set the working directory under Sessions Tab to access the corresponding CSV files for data processing purpose.

# Step 2: Installing necessary libraries

install.packages(“tidyverse”) install.packages(“ggplot2”) install.packages(“maps”)

# Step 3: Load the necessary libraries

library(tidyverse) library(ggplot2) library(maps)

# Step 4: Cleaning the datasets and making it ready for processing

As an essential step of cleaning the dataset, we remove the blank spaces and replace them with dots in the column names for example “First Doses Administered” to “First.Doses.Administered”. This was basically done so that fetching the column names becomes easy during function calling. We save the new data in a new CSV and name the file as dataset1.csv.

In order to do the above steps, the following code chunk has been implemented:

# Read data from CSV file

covid\_data <- read.csv(“covid\_vaccine\_statewise.csv”, stringsAsFactors = FALSE)

# Remove blank spaces from column names

colnames(covid\_data) <- gsub(” “,”“, colnames(covid\_data))

# Save the modified data to the existing CSV file

write.csv(covid\_data, “dataset1.csv”, row.names = FALSE)

# Libraries and their functions

We know that data visualization is accomplished by a package called ggplot2. This package is part of a larger group of related packages called tidyverse. Hence, we need to install the package tidyverse.

# Explore the structure of the raw data

It is essential to know the structure of the raw data before processing is done. To know the structure of the raw data, we implement str() function. str(covid\_data)

The above function returns 7845 records (rows) and 24 variables (columns).

A screenshot of a computer

Description automatically generated

It is essential to know that the blank values in the attribute fields has been substituted with NA values for our convenience. In later times, we can safely discard the NA values and deal with the original integer or character values whenever necessary.

# Step 5: Tidying the data

Next step, we try to tidy the data. Tidy data in R refers to a specific structure of data that makes it easier to work with and analyze using various R packages, particularly those in the tidyverse (e.g., dplyr, ggplot2). Tidy data principles, as introduced by Hadley Wickham, promote a standardized way of organizing data that facilitates efficient analysis and visualization.

# Tidying the covid\_vaccine\_statewise.csv data

covid\_data <- read.csv(“dataset1.csv”, stringsAsFactors = FALSE) # Loading the dataset1.csv to the covid\_data variable

tidy\_covid\_data <- covid\_data %>% # Handle missing values if necessary # Example: drop rows with any missing values drop\_na() %>%

# Convert Date to a Date object mutate(Date = as.Date(Date, format = “%d/%m/%Y”))

Here the dataset1.csv is assigned to the variable covid\_data. In return the covid\_data variable is reassigned to the variable tidy\_covid\_data and undergone through drop\_na() function which drops the NA values for the corresponding data frames. The variable also undergoes through the mutate() function, which is used for creating or modifying variables (columns) in a data frame. For example here the mutate() function changes the Date format to Day/Month/Year.

We can also use pivot functions for reshaping or transforming data from a wide format to a long format or vice versa. There are different functions for pivoting, for example the pivot\_longer and pivot\_wider functions from the tidyverse package, part of the tidyverse, are commonly used for these tasks. We essentially did not use pivot functions here but a general example is given below:

# Example: If you have separate columns for confirmed, recovered, and deaths, pivot to long format pivot\_longer(cols = c(Confirmed, Deaths), names\_to = “Status”, values\_to = “Count”)

# Finally exploring the structure of the new tidy data

str(tidy\_covid\_data)

# Save the tidy data to a new CSV file

write.csv(tidy\_covid\_data, “tidy\_dataset1.csv”, row.names = FALSE)

# Step 6: Creating plots for visualization

# Example 1. Visualizing Male vs. Female Vaccine Doses Administered for a particular date of the month

# Load libraries

library(ggplot2) library(tidyverse)

# Read data from CSV file

vaccine\_data <- read.csv(“dataset2.csv”, stringsAsFactors = FALSE)

# Convert Date to a Date object

vaccine\_dataUpdated.On, format = “%d/%m/%Y”)

# Specify the particular date (replace ‘2022-01-15’ with your desired date)

selected\_date <- as.Date(‘16/01/2021’)

# Filter data for the selected date

filtered\_data <- vaccine\_data %>% filter(Updated.On == selected\_date)

# Create a bar graph

ggplot(filtered\_data, aes(x = Gender, y = Total.Individuals.Vaccinated., fill = Gender)) + geom\_bar(stat = “identity”, position = “dodge”) + labs(title = paste(“Individuals Vaccinated by Gender on”, format(selected\_date, “%d/%m/%Y”)), x = “Gender”, y = “Total Individuals Vaccinated”, fill = “Gender”) + theme\_minimal()

The output of the above code is given below:

A graph with a green and blue bar

Description automatically generated

# Example 2. Time series of Initial Cases in Kerala from 1/30/2020 to 3/1/2020

# Filter data for the specified date range

covid\_data <- read.csv(“covid\_19\_india.csv”) filtered\_covid\_data1 <- covid\_data %>% filter(Sno >= 1, Sno <=32)

ggplot(data = filtered\_covid\_data1, mapping = aes(x = Sno, y = Confirmed)) + geom\_point(mapping = aes(color = “Red”))

The output of the above code is given below:

A graph with red dots

Description automatically generated

# Example 3. Time series of First Dose Administered Vaccination Initially in Random states

covid\_data <- read.csv(“covid\_vaccine\_statewise.csv”)

filtered\_covid\_data2 <- covid\_data %>% filter(State == “India”)

ggplot(data = filtered\_covid\_data2, mapping = aes(x = FirstDoseAdministered)) + geom\_point(mapping = aes(color = “Red”))

The output of the above code is given below:

A graph with a line drawn on it

Description automatically generated

# Example 4. Graph plot to visualize females vaccination status from West Bengal in the period 27/06/2021 to 07/07/2021

covid\_data3 <- read.csv(“dataset1.csv”) filtered\_covid\_data3 <- covid\_data3 %>% filter(State == “West Bengal”, FemaleIndividualsVaccinated >= 1) ggplot(data = filtered\_covid\_data3, mapping = aes(x = Female.Individuals.Vaccinated., y = Sno)) + geom\_point(mapping = aes(color = "Red"))+ labs(title = paste("Female Individuals Vaccinated in the period 27/06/2021 to 07/07/2021"))

In this case it does not show any plotting in the graph. This is because data is missing in this region of the dataset. Hence, there is some level of inconsistency in the dataset.

The output of the above code is given below:

A white rectangular object with text

Description automatically generated

# Example 5. Graph plot to visualize the installation of vaccine sites throughout 2020 to 2021

Now, let’s try to visualize how the vaccine sites accomodation has changed over the time with respect to the sessions of vaccine doses.

covid\_data3 <- read.csv(“covid\_vaccine\_statewise.csv”)

ggplot(data = covid\_data3, mapping = aes(x = Sessions, y= Sites)) + geom\_point(alpha = 1/3) + geom\_line(alpha = 1/3) + geom\_smooth()

This graph plot shows that India has made an exponential growth (mostly) in developing vaccine sites for conducting vaccination sessions on the common populi.

The output of the above code is given below:

A graph with a curve and lines

Description automatically generated with medium confidence

The same data if produced in a histogram produces the result:

ggplot(data = covid\_data3) + geom\_histogram(mapping = aes(x = Sessions), bins = 100)

The output of the above code is given below:

A graph with a number of numbers

Description automatically generated with medium confidence

# Example 6. Graph plot to visualize the state of Odisha in India has made an exponential growth (mostly) in developing vaccine sites for conducting vaccination sessions

This graph plot shows for the state of Odisha in India has made an exponential growth (mostly) in developing vaccine sites for conducting vaccination sessions on the common populi after making a slight degradation it has made a steep positive slope indicating the success in setting up new vaccination sites for vaccination purpose.

covid\_data4 <- filter(covid\_data3, State == “Odisha”) ggplot(data = covid\_data4, mapping = aes(x = Sessions, y = Sites)) + geom\_point(mapping = aes(size = Sessions), alpha = 1/3) + geom\_smooth()

The output of the above code is given below:

A graph with dots and lines

Description automatically generated

# Discussion

Our plots suggest that India had tactfully increased the number of vaccination sites across all states to expedite the process of curbing the spread of Covid-19 virus all over the country. The increased number of vaccination sites allowed more vaccination sessions to be conducted thereby making the general population of the people vaccinated to stop the spread of the virus. We also discussed the sudden spike in covid cases initially when Covid-19 hit the country. The spike was found in the state of Kerala and it was essential to know how the cases across Kerala and then further.

# Conclusion

India, faced with the formidable challenge of the COVID-19 pandemic, exhibited resilience and determination in its fight against the virus. The nation implemented a series of stringent measures, including increase of vaccination sites to carry out successful vaccination drives, to curb the spread of the virus and protect its citizens. Our plots show how the nation has exponentially grown the installation of vaccination so that proper doses can be administered to those needed. The successful management of the pandemic underscored the importance of effective public health strategies, global collaboration, and the resilience of the Indian people in overcoming adversity.

# Future Scope

The pilot study can be expanded to study the vaccination records and mitigation strategies for other countries.

# References

<https://www.kaggle.com/datasets/sudalairajkumar/covid19-in-india/> <https://www.kaggle.com/code/evangower/visualizing-covid-19-using-r/input> <https://dplyr.tidyverse.org/reference/mutate.html> <https://r4ds.had.co.nz/tidy-data.html> <https://www.mohfw.gov.in/> <https://www.covid19india.org/> <https://www.isibang.ac.in/~athreya/incovid19/>