Parshvanath Charitable Trust's
A. P. SHAH INSTITUTE OF TECHNOLOGY, THANE
(All Programs Accredited by NBA)
Department of Information Technology

Academic Year: 2021-22

Semester: VIII

Class / Branch: BE IT

Subject: R PROGRAMMING LAB

Subject In-charge: Shafaque Fatma Syed

STUDENT TEAM:

Akshata Gawas (18104039)

Siddhesh Gaikwad (18104069)

Krishita Tolia (18104021)

MINI-PROJECT REPORT

ON

"COVID-19 World Vaccination Progress"

CONTENTS OF THE REPORT

INTRODUCTION

To bring this pandemic to an end, a large share of the world needs to be immune to the virus. The safest way to achieve this is with a vaccine. Vaccines are a technology that humanity has often relied on in the past to bring down the death toll of infectious diseases.

Within less than 12 months after the beginning of the pandemic, several research teams rose to the challenge and developed vaccines that protect from COVID-19.

Now the challenge is to make these vaccines available to people around the world. It will be key that people in all countries — not just in rich countries — receive the required protection. It is important to track the progress of vaccinations. With the help of our proposed idea we can track the progress of vaccinations but also predict the number of vaccinated people in future using Machine Learning algorithm 'Linear Regression'. We can also find out the number of vaccines of certain brand are taken people world wide.

OBJECTIVES

The key objectives include:

- To find vaccination progress across different countries.
- To find total people vaccinated around the world.
- To visualize vaccination progress in India.
- To predict number of vaccinated people in India after 500 days.

SCOPE

The scope of the project include:

- The visualization can be used to understand the actual progress of vaccination not just for country but also worldwide.
- The vaccine companies can also find out the count of vaccines used all over the world as well as in a particular country.

SUMMARIZATION OF THE DATASET

The dataset is named "country_vaccinations.csv" was downloaded from Kaggle having data consisting of country, date, daily vaccination, daily vaccinations per million, vaccines, source, etc. The dataset had 15 columns, and 74961 rows having data of different countries. The range of date for which the data was recorded was "2021-01-16" to "2022-02-04" i.e 385 days.

Filter

	country	iso_code	date	total_vaccinations	people_vaccinated	people_fully_vaccinated	daily_vaccinations_raw	da
1	Afghanistan	AFG	2021-02-22	0	0	NA	NA	
2	Afghanistan	AFG	2021-02-23	NA	NA	NA	NA	
3	Afghanistan	AFG	2021-02-24	NA	NA	NA	NA	
4	Afghanistan	AFG	2021-02-25	NA	NA	NA	NA	
5	Afghanistan	AFG	2021-02-26	NA	NA	NA	NA	
6	Afghanistan	AFG	2021-02-27	NA	NA	NA	NA	
7	Afghanistan	AFG	2021-02-28	8200	8200	NA	NA	
8	Afghanistan	AFG	2021-03-01	NA	NA	NA	NA	
9	Afghanistan	AFG	2021-03-02	NA	NA	NA	NA	
10	Afghanistan	AFG	2021-03-03	NA	NA	NA	NA	
11	Afghanistan	AFG	2021-03-04	NA	NA	NA	NA	
12	Afghanistan	AFG	2021-03-05	NA	NA	NA	NA	

Showing 1 to 13 of 74,961 entries, 15 total columns

Fig. 1 Dataset

VISUALIZATION OF THE DATASET

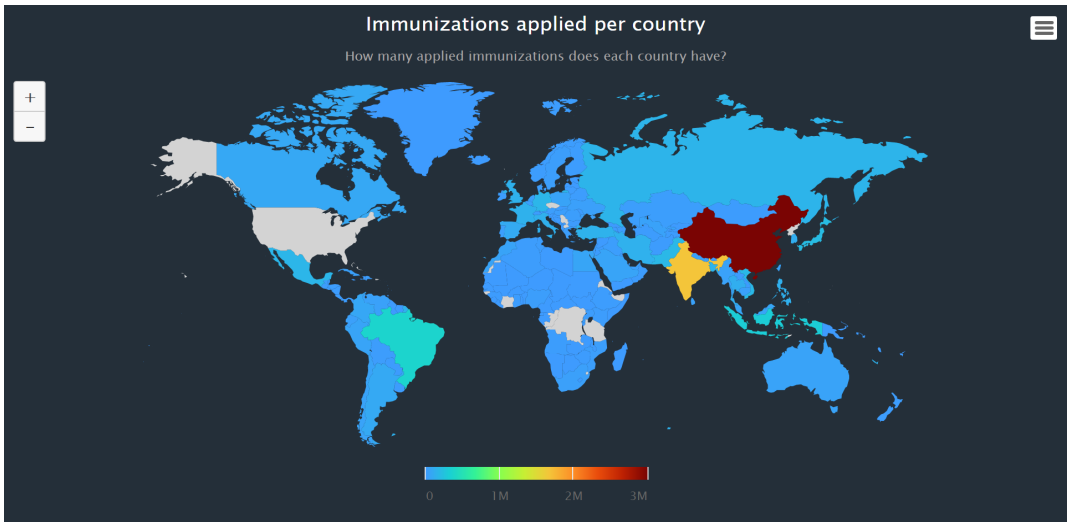


Fig. 2: World vaccination progress

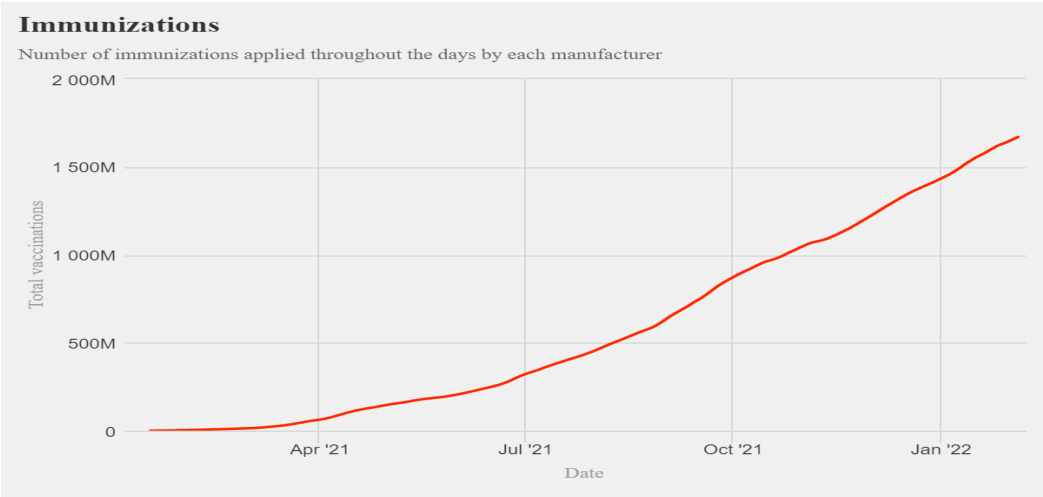


Fig. 3: India vaccination progress

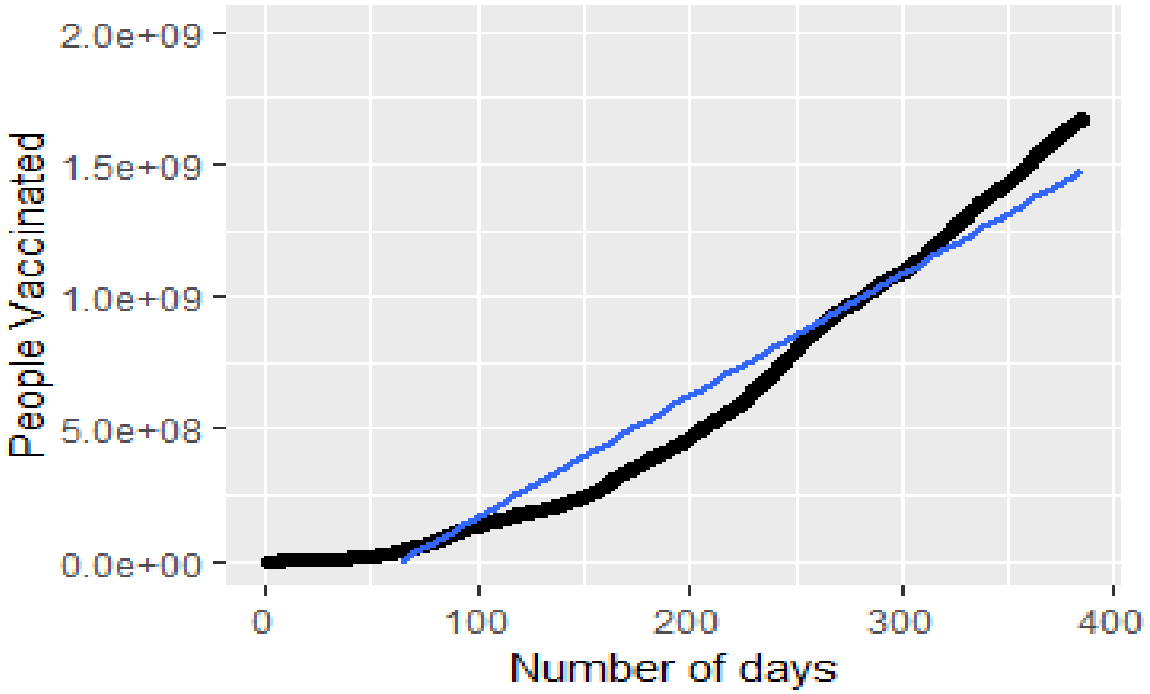


Fig.4: Plotting the regression line

ALGORITHM

The algorithm used for prediction of number of vaccinated people in future is **Linear Regression**.

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between dependent and independent variables. Different regression models differ based on – the kind of relationship between dependent and independent variables they are considering, and the number of independent variables getting used.

Linear regression typically takes the form

$$y = \beta X + \epsilon$$

where 'y' is a vector of the response variable, 'X' is the matrix of our feature variables or called the 'design' matrix, and β is a vector of parameters that we want to estimate. ϵ is the error term.

We will need is a vector of our response variable, typically called 'y' and 'X' or 'design' matrix of the input features.

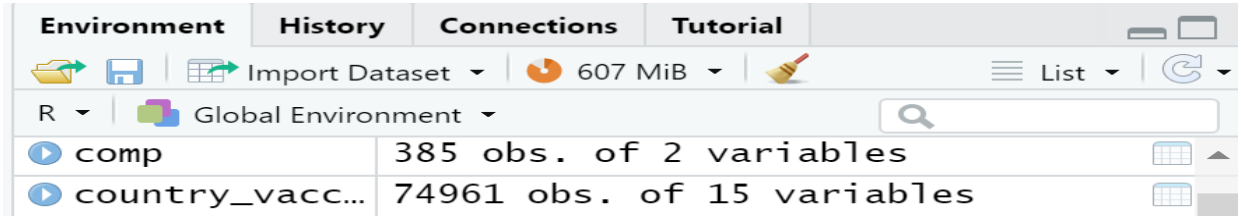
$$\beta = (X^T X)^{-1} X^T y$$

By calculating β, we get the parameter vector. It contains the linear relationships between the response variable 'y' and each of the features in X.

IMPLEMENTATION

1. Initially we installed and imported all the necessary libraries for our project. Then we imported the csv file containing our data set into a variable as a data frame.

```
library(funModeling)
library(tidyverse)
library(Hmisc)
library(ggplot2)
library(readr)
library(dplyr)
library(janitor)
library(bit64)
library(highcharter)


country_vaccinations <- read_csv("C:/Users/Akshata/Documents/R_Project/country_vaccinations.csv")
View(country_vaccinations)
```



2. Then we converted the numerical values to integer and checked if any duplicates were found.

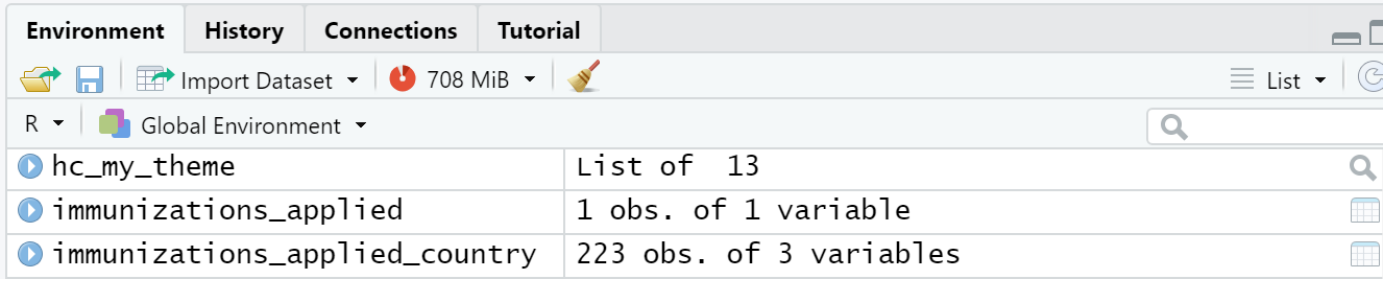
```
country_vaccinations.formatted <- country_vaccinations %>%
  mutate(daily_vaccinations = as.integer64(daily_vaccinations)) %>%
  mutate_at(c('people_vaccinated',
              'daily_vaccinations_raw',
              'total_vaccinations',
              'daily_vaccinations_per_million'), as.integer)


anyDuplicated(country_vaccinations)
```

```
> anyDuplicated(country_vaccinations)
[1] 0
>
```

3. We created a new data frame where we added the daily vaccination of all the countries to find the total people vaccinated till "2022-02-04" and visualized the data.

```
immunizations_applied_country <- country_vaccinations.formatted %>%
  group_by(country, iso_code) %>%
  summarise(total_immunizations = sum(daily_vaccinations, na.rm = TRUE), .groups = 'drop') %>%
  arrange(desc(total_immunizations))
View(immunizations_applied_country)
```

	country	iso_code	total_immunizations
1	China	CHN	2999723926
2	India	IND	1670383729
3	United States	USA	541538267
4	Brazil	BRA	364113021
5	Indonesia	IDN	310440640
6	Japan	JPN	205875784

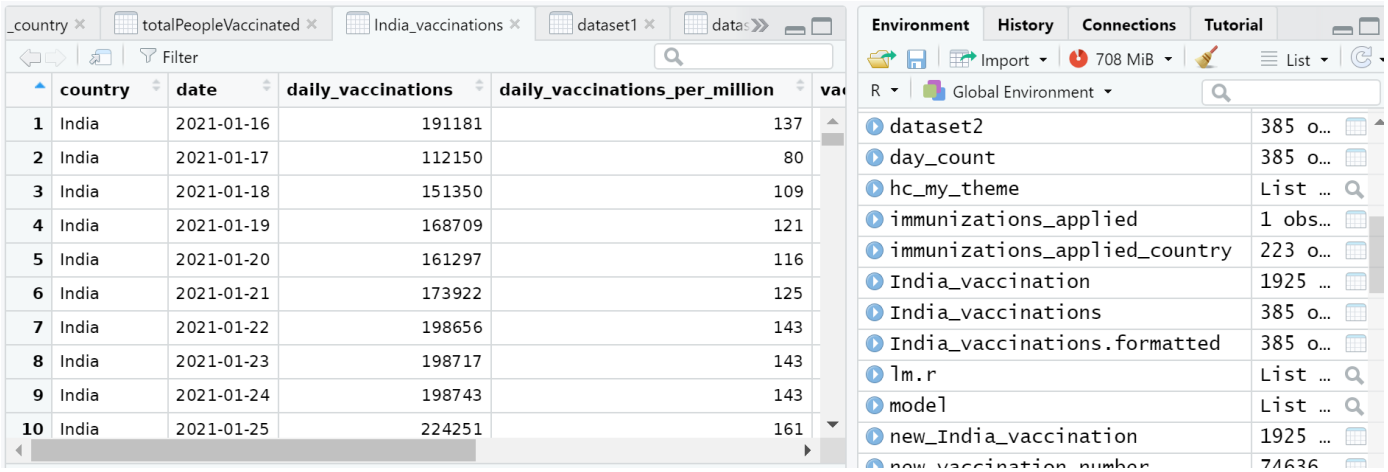


4. Similarly we did the above step for country “India”.

```
totalPeopleVaccinated <- c()
num <- 0
num <- as.double(num)
find_sum <- as.double(India_vaccinations$daily_vaccinations)
find_sum <- drop_na(find_sum)
for (i in find_sum) {
  num <- num + i
  totalPeopleVaccinated <- append(totalPeopleVaccinated,num)

}
totalPeopleVaccinated <- data.frame(totalPeopleVaccinated)
View(totalPeopleVaccinated)
India_vaccinations <- cbind(India_vaccinations, totalPeopleVaccinated)
```

	totalPeopleVaccinated
1	191181
2	303331
3	454681
4	623390
5	784687
6	958609
7	1157265
8	1355982
9	1554725
10	1778976
11	1972497

Showing 1 to 11 of 385 entries, 1 total columns



_country | totalPeopleVaccinated | India_vaccinations | dataset1

Filter

	country	date	daily_vaccinations	daily_vaccinations_per_million	va
1	India	2021-01-16	191181	137	
2	India	2021-01-17	112150	80	
3	India	2021-01-18	151350	109	
4	India	2021-01-19	168709	121	
5	India	2021-01-20	161297	116	
6	India	2021-01-21	173922	125	
7	India	2021-01-22	198656	143	
8	India	2021-01-23	198717	143	
9	India	2021-01-24	198743	143	
10	India	2021-01-25	224251	161	

Environment | History | Connections | Tutorial

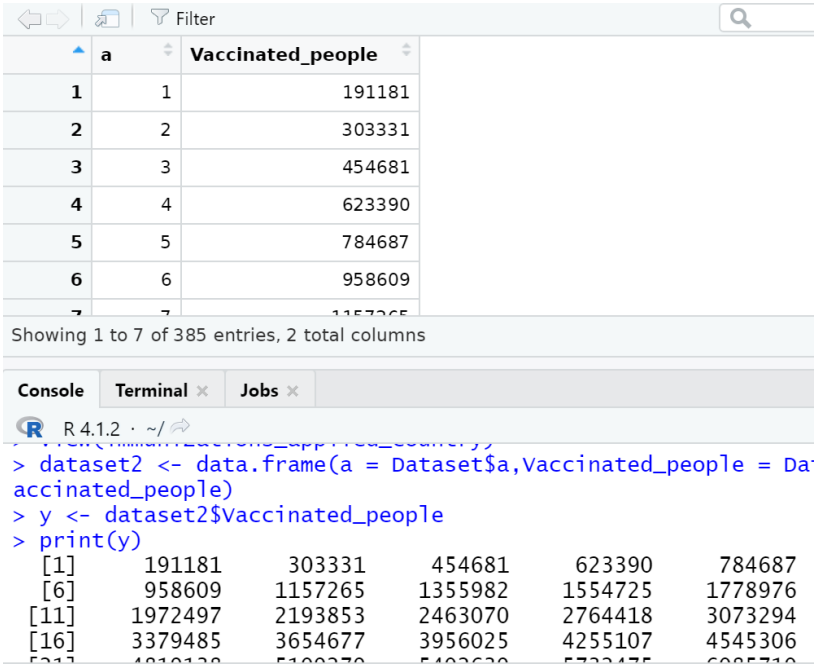
Import | 708 MiB | List

R | Global Environment

dataset2	385 o...
day_count	385 o...
hc_my_theme	List ...
immunizations_applied	1 obs...
immunizations_applied_country	223 o...
India_vaccination	1925 ...
India_vaccinations	385 o...
India_vaccinations.formatted	385 o...
lm.r	List ...
model	List ...
new_India_vaccination	1925 ...

5. For predicting number of vaccinated people in future we applied simple linear regression by using number days column ‘a’ and number of people vaccinated column ‘Vaccinated_people’.

```
######
dataset2 <- data.frame(a = Dataset$a,Vaccinated_people = Dataset$Vaccinated_people)
y <- dataset2$Vaccinated_people
print(y)
View(dataset2)
X <- as.matrix(dataset2[-ncol(dataset2)])
int <- rep(1, length(y))
X <- cbind(int, X)
```



Filter

	a	Vaccinated_people
1	1	191181
2	2	303331
3	3	454681
4	4	623390
5	5	784687
6	6	958609

Showing 1 to 7 of 385 entries, 2 total columns

Console Terminal Jobs

R 4.1.2 · ~/

```
> dataset2 <- data.frame(a = Dataset$a,Vaccinated_people = Da
accinated_people)
> y <- dataset2$Vaccinated_people
> print(y)
  [1]     191181     303331     454681     623390     784687
  [6]     958609    1157265    1355982    1554725    1778976
 [11]    1972497    2193853    2463070    2764418    3073294
 [16]    3379485    3654677    3956025    4255107    4545306
```

```
betas <- solve(t(X) %*% X) %*% t(X) %*% y
betas <- round(betas, 2)
print(betas)
```

```
> print(betas)
             [,1]
int -294090435
a       4591291
```

Here, we can see that the betas values i.e. the parameter, intercept is -294090435 and the value of a represents the relationship between the column ‘a’ and ‘Vaccinated_people’.

RESULTS AND CONCLUSION

In the screenshot below, we have used the formula y = β*X + intercept to find out the number of people vaccinated on the 500th day using the data from the dataset.

```
> prediction <- betas[2]*500 + betas[1]
> print(prediction)
[1] 2001555090
>
```

Here, we can see the comparison of predicted values and the actual values.

comp <- cbind(Dataset$Vaccinated_people, data.frame(result))

View(comp)

Filter

	Dataset$Vaccinated_people	result
277	974834903	977697185
278	979334144	982288476
279	984311607	986879767
280	989851924	991471058
281	995702108	996062349
282	1002262559	1000653640
283	1008719018	1005244931
284	1014967165	1009836222
285	1020562790	1014427513
286	1026170539	1019018804
287	1032136678	1023610095

Showing 276 to 287 of 385 entries, 2 total columns

Root mean squared error:

```
> rmse <- 1/385*sum(a)
> print(rmse)
[1] 2.03388e+18
```

Correlation coefficient:

```
> cor(Dataset$Vaccinated_people, Dataset$a)
[1] 0.9727252
>
```

We can also find count of particular vaccines taken by people through out the world.

```
> length(grep("Johnson&Johnson", country_vaccinations$vaccines))
[1] 28580
```

REFERENCES

1. https://www.kaggle.com/datasets/gpreda/covid-world-vaccination-progress
2. library(ggplot2) : https://www.rdocumentation.org/packages/ggplot2/versions/3.3.5
3. library(highcharter) : https://www.rdocumentation.org/packages/highcharter/versions/0.9.4
4. library(dpylr) : https://www.rdocumentation.org/packages/dplyr/versions/0.7.8
5. library(Hmisc) : https://www.rdocumentation.org/packages/Hmisc/versions/4.7-0
6. library(tidyverse) : https://www.rdocumentation.org/packages/tidyverse/versions/1.3.1
7. library(janitor) : https://www.rdocumentation.org/packages/janitor/versions/2.1.0
8. library(funModeling) : https://www.rdocumentation.org/packages/funModeling/versions/1.9.4
9. Healy, Kieran. *Data visualization: a practical introduction*. Princeton University Press, 2018.
10. Pearson, Ronald K. *Exploratory data analysis using R*. Chapman and Hall/CRC, 2018.