



iJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: II Month of publication: February 2023

DOI: <https://doi.org/10.22214/ijraset.2023.48991>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Covid-19 for India: A Data Analytics Approach

Riddhi Chatterjee¹, Ritwik Mukherjee², Soumik Podder³

Department of Electronics and Communication Engineering, Guru Nanak Institute of Technology

Abstract: The COVID-19 epidemic has caused a large number of human losses and havoc in the economic, social, societal, and health system around the world. India also severely impacted throughout 34 states. First case reported on: 30 January 2020. As of 10th June Ministry of Health and Family Welfare reported a total of 276,804 cases, 134,843 recoveries and 7,751 deaths. In this project, we aim to analyse COVID-19 data in India and find the key insights on patient demographics, patient clusters, state and district level spread. To be able to foresee the local transmission rate, top affected districts and predict the saturation point of the disease spread. In this work some datasets for each requirement were collected and organized those as per our requirements. All redundant data were eradicated and made it useful for each requirement. These data were grouped using Hadoop. After this query was written using Hive. To improve this planning Power BI was used for enhancing the visions on COVID-19 as per the said requirements. The collected data were used for state purpose to understand how COVID-19 spreads, the severity of disease it causes, how to treat it, and how to stop it and update the strategy based on evidence.

Keywords: COVID-19 data set, Hive Tools, BI generator, Big data, Data Analytics, Hadoop, Distributed File System

I. INTRODUCTION

In the current pandemic situation where COVID-19 has spread across the world impacting 150+ countries, India too is severely impacted throughout 34 states. So, now analysing COVID-19 data for India using Big Data is become the most popular and important topic for understanding patient clusters, state and district level spread and find key insights on patient demographics and be able to foresee the local transmission rate, top affected districts and predict the saturation point of the disease spread. In this project the patient demographics were analysed to understand the spread and trend in daily cases treatment. As India is a vast country with a geographic area of 3,287,240 square of about 1.3 billion so, we consider state wise to identify the top 6 affected districts in India. Big Data is not only a broad term but also a latest approach to analyze a complex and huge amount of data; there is no single accepted definition for Big Data. The challenge of Big Data is how to use it to create something that is value to the user. How to gather it, store it, process it and analyze it to turn the raw data information to support decision making.

Hadoop allows to store and process Big Data in a distributed environment across group of computers using simple programming models. It is intended to scale up starting with solitary machines and will be scaled to many machines. In this paper Hive tool is used. The primary goal of Hive is to provide answers about business functions, system performance, and user activity. To meet these needs strongly dumping the data into MySQL data set, but now since huge amount of data in Terabytes which is injected into Hadoop Distributed File System files and processed by Hive Tool.

Power BI is a Data Visualization and Business Intelligence tool that converts data from different data sources to interactive dashboards and BI reports. In this project we also use power BI for data visualization. In the present work we are identifying the top 6 affected districts in India in order to improve the planning and strategy. Then we are identifying the patient demographics to understand the spread and trend in daily cases treatment. After that we will analyse the key insights for new policy formation like lockdown extension and also predict the saturation point for the spread. Customer can even get information about average time taken by patients to recover and average time a patient stays in hospital. Customer can also track load on healthcare facility and get the predicted number of ventilators required in future.

II. SIGNIFICANCE

Hadoop is a highly scalable storage platform, because it can store and distribute very large data sets across hundreds of inexpensive servers that operate in parallel so that thousands of terabytes of data and information regarding COVID-19 could easily be accessed. The motivation behind the use of Big data analytics derives innovative solutions and it helps in understanding and targeting customers. It helps in optimizing business processes and also helps in improving science and research. Here we identify patient demographics to understand the spread and trend in daily cases treatment so it improves healthcare and public health with availability of record of patients. Here the number of ventilators are predicted in order to satisfy the requirement in future, so using this information many hospitals can understand the number of mechanical ventilators they needed to have available.

It is a cost effective storage solution for exploring data sets. This provides reliable prediction of cases and performs even better when modified to include more states and parameters. Straight forward and simple to use. Every second additions are made. One platform carry unlimited information. Anyone can access vast information via surveys and deliver answer of any query.

III.METHODOLOGY

Hadoop is used for this data analytics. Hive is mainly used for structured data assuming all the Hadoop tools have been installed and having semi structured information on COVID-19 data. Power BI is used for COVID-19 data visualization.

Methodology used are as follows:

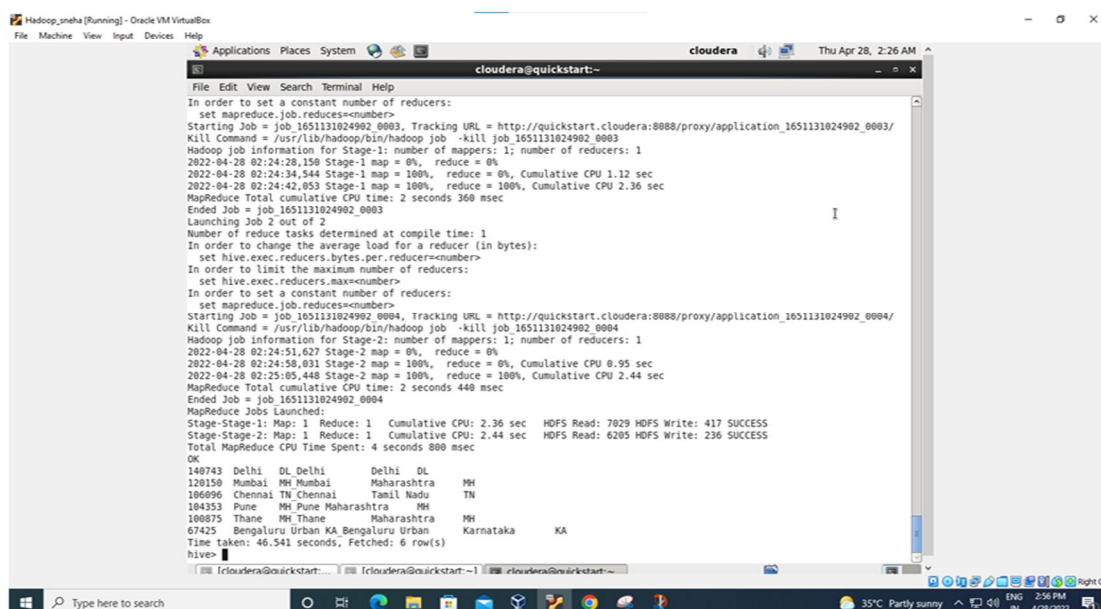
- 1) Create tables with required attributes
- 2) Extract semi structured data into table using the load a command
- 3) Analyse data for the following queries
 - a) List of top 6 affected districts in India Improves planning and strategy .
 - b) List of patient numbers to understand the spread and trend in daily cases treatment.
 - c) Average time taken by patients to recover.
 - d) The saturation point for the spread .

In the current pandemic situation where COVID-19 has spread across the world impacting 150+ countries, India too is severely impacted: 34 states / UT affected. First Case Reported on: 30 January 2020 as of 10th June, Ministry of Health and Family Welfare reported a total of 276,804 cases, 134,843 recoveries and 7,751 deaths the infection rate of COVID-19 in India is: 1.7 (significantly lower than in the worst affected countries.

TABLE I

Req_ID	Requirements
1	Data preparation: Loading into Hadoop Quality data preparation Identifying tool for BI
2	Identifies the top 6 affected districts in India Improves planning and strategy to provide
3	Identifies patient demographics to understand the spread and trend in daily cases treatment
4	Predicts the saturation point for the spread
5	Average time taken by patients to recover, average time a patient stays in hospital – hospital occupancy

IV.RESULT AND DISCUSSION

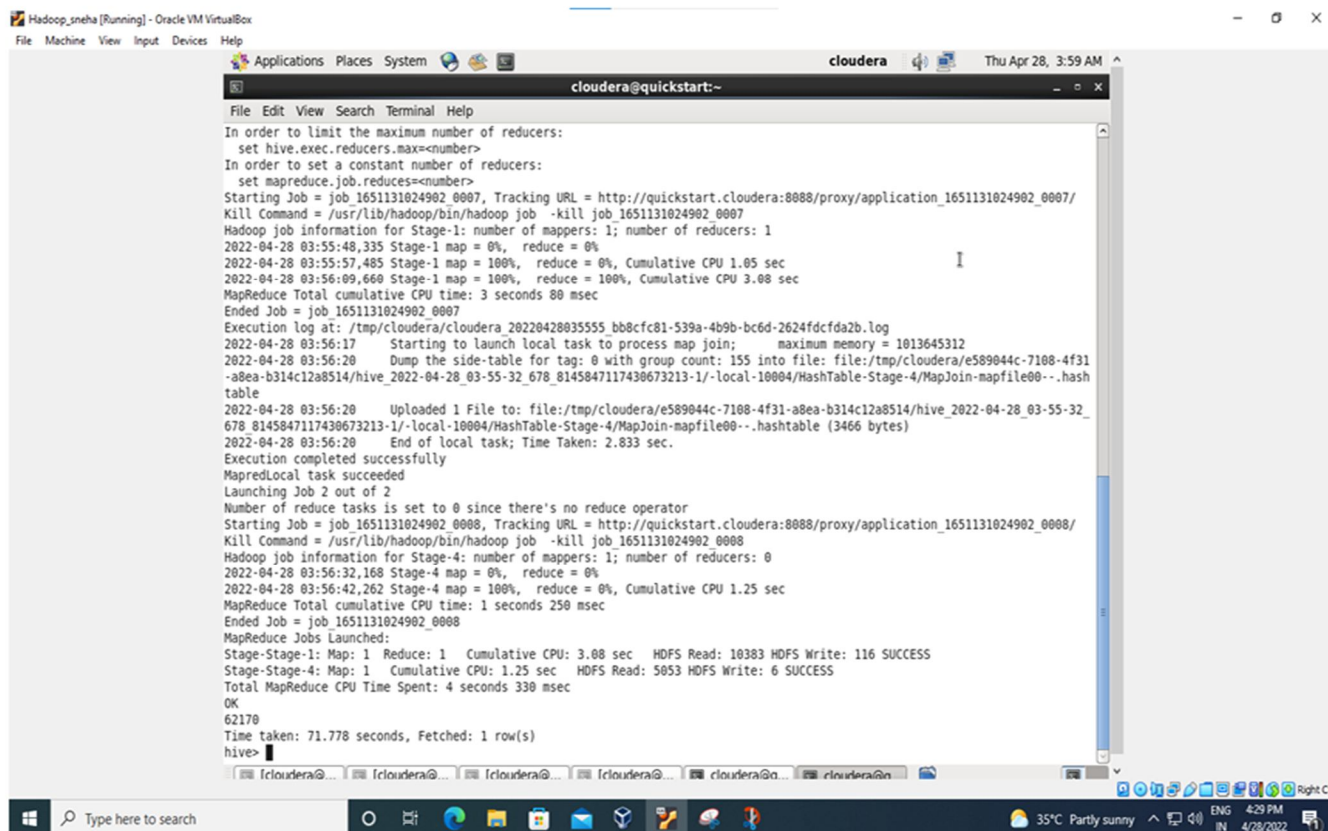


```

Hadoop_sneha [Running] - Oracle VM VirtualBox
File Edit View Search Terminal Help
cloudera@quickstart:~$
In order to set a constant number of reducers:
set mapreduce.job.reduces=<number>
Starting Job = job_1651131024902_0003, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1651131024902_0003/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1651131024902_0003
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-04-28 02:24:28,150 Stage-1 map = 0%, reduce = 0%
2022-04-28 02:24:34,544 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.12 sec
2022-04-28 02:24:42,053 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 2.36 sec
MapReduce Total cumulative CPU time: 2 seconds 360 msec
Ended Job = job_1651131024902_0003
Launching Job 2 out of 2
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
set mapreduce.job.reduces=<number>
Starting Job = job_1651131024902_0004, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1651131024902_0004/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1651131024902_0004
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2022-04-28 02:24:51,627 Stage-2 map = 0%, reduce = 0%
2022-04-28 02:24:58,031 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 0.95 sec
2022-04-28 02:25:05,440 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 2.44 sec
MapReduce Total cumulative CPU time: 2 seconds 440 msec
Ended Job = job_1651131024902_0004
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 2.36 sec HDFS Read: 7029 HDFS Write: 417 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 2.44 sec HDFS Read: 6205 HDFS Write: 236 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 800 msec
OK
140743 Delhi DL Delhi Delhi DL
120150 Mumbai MH Mumbai Maharashtra MH
106096 Chennai TN Chennai Tamil Nadu TN
104353 Pune MH Pune Maharashtra MH
100075 Thane MH Thane Maharashtra MH
67425 Bengaluru Urban KA Bengaluru Urban Karnataka KA
Time taken: 46.541 seconds, Fetched: 6 row(s)
hive>

```

Fig 1: Identification of the top 6 affected districts in India Improves planning and strategy

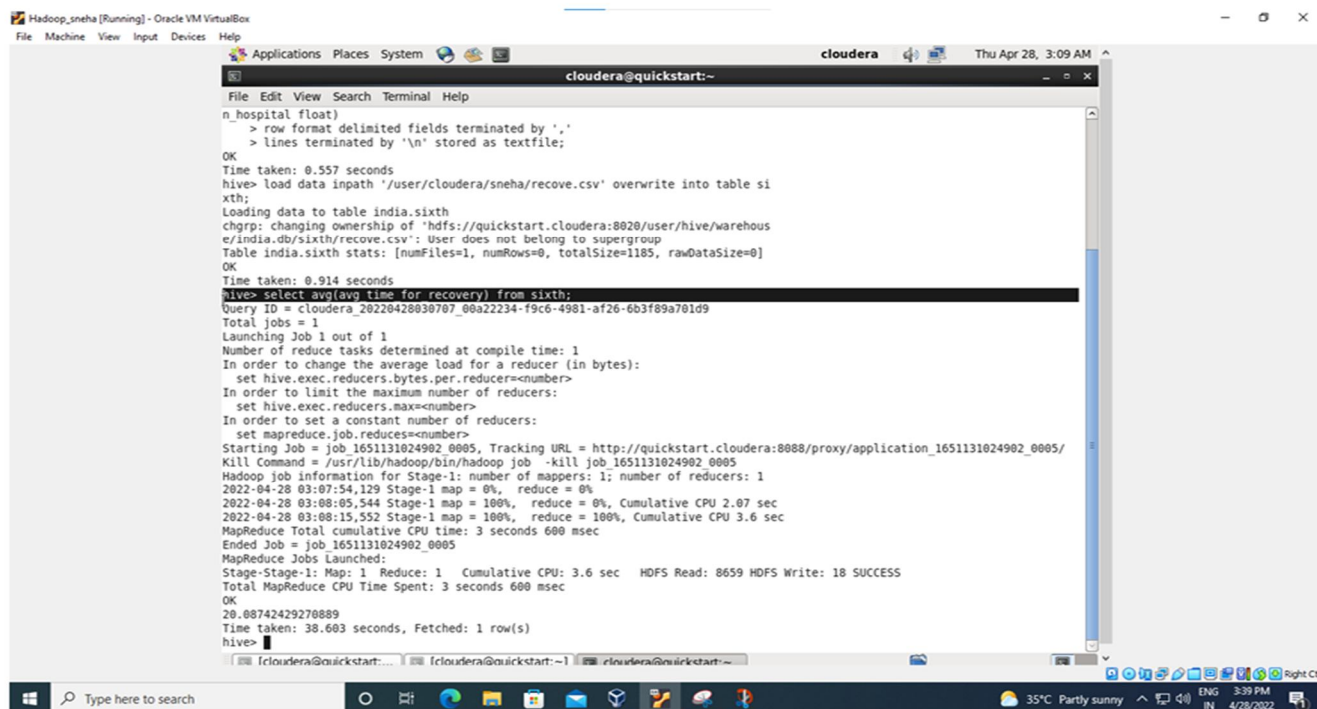


```

cloudera@quickstart:~$
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1651131024902_0007, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1651131024902_0007/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1651131024902_0007
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-04-28 03:55:48,335 Stage-1 map = 0%, reduce = 0%
2022-04-28 03:55:57,485 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.05 sec
2022-04-28 03:56:09,660 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.08 sec
MapReduce Total cumulative CPU time: 3 seconds 80 msec
Ended Job = job_1651131024902_0007
Execution log at: /tmp/cloudera/cloudera_20220428035555_bb8cfc81-539a-4b9b-bc6d-2624fcd42b.log
2022-04-28 03:56:17 Starting to launch local task to process map join; maximum memory = 1013645312
2022-04-28 03:56:20 Dump the side-table for tag: 0 with group count: 155 into file: file:/tmp/cloudera/e589044c-7108-4f31-a8ea-b314c12a8514/hive_2022-04-28_03-55-32_678_8145847117430673213-1/-local-10004/HashTable-Stage-4/MapJoin-mapfile000...hash
table
2022-04-28 03:56:20 Uploaded 1 File to: file:/tmp/cloudera/e589044c-7108-4f31-a8ea-b314c12a8514/hive_2022-04-28_03-55-32_678_8145847117430673213-1/-local-10004/HashTable-Stage-4/MapJoin-mapfile000...hash
table (3466 bytes)
2022-04-28 03:56:20 End of local task; Time Taken: 2.833 sec.
Execution completed successfully
MapredLocal task succeeded
Launching Job 2 out of 2
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1651131024902_0008, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1651131024902_0008/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1651131024902_0008
Hadoop job information for Stage-4: number of mappers: 1; number of reducers: 0
2022-04-28 03:56:32,168 Stage-4 map = 0%, reduce = 0%
2022-04-28 03:56:42,262 Stage-4 map = 100%, reduce = 0%, Cumulative CPU 1.25 sec
MapReduce Total cumulative CPU time: 1 seconds 250 msec
Ended Job = job_1651131024902_0008
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.08 sec HDFS Read: 10383 HDFS Write: 116 SUCCESS
Stage-Stage-4: Map: 1 Cumulative CPU: 1.25 sec HDFS Read: 5053 HDFS Write: 6 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 330 msec
OK
62170
Time taken: 71.778 seconds, Fetched: 1 row(s)
hive>

```

Fig 2: Identification of patient demographics to understand the spread and trend in daily cases treatment



```

cloudera@quickstart:~$
n_hospital float)
> row format delimited fields terminated by ','
> lines terminated by '\n' stored as textfile;
OK
Time taken: 0.557 seconds
hive> load data inpath '/user/cloudera/sneha/recove.csv' overwrite into table si
xth;
Loading data to table india.sixth
chgrp: changing ownership of 'hdfs://quickstart.cloudera:8020/user/hive/warehouse
e/india.db/sixth/recove.csv': User does not belong to supergroup
Table india.sixth stats: [numFiles=1, numRows=0, totalSize=1185, rawDataSize=0]
OK
Time taken: 0.914 seconds
hive> select avg(avg time for recovery) from sixth;
Query ID = cloudera_20220428030707_00a22234-f9c6-4981-af26-6b3f89a701d9
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1651131024902_0005, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1651131024902_0005/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1651131024902_0005
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-04-28 03:07:54,129 Stage-1 map = 0%, reduce = 0%
2022-04-28 03:08:05,544 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.07 sec
2022-04-28 03:08:15,552 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.6 sec
MapReduce Total cumulative CPU time: 3 seconds 600 msec
Ended Job = job_1651131024902_0005
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.6 sec HDFS Read: 8659 HDFS Write: 18 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 600 msec
OK
20.08742429270889
Time taken: 38.603 seconds, Fetched: 1 row(s)
hive>

```

Fig 3: Identification of the Average time taken by patients to recover

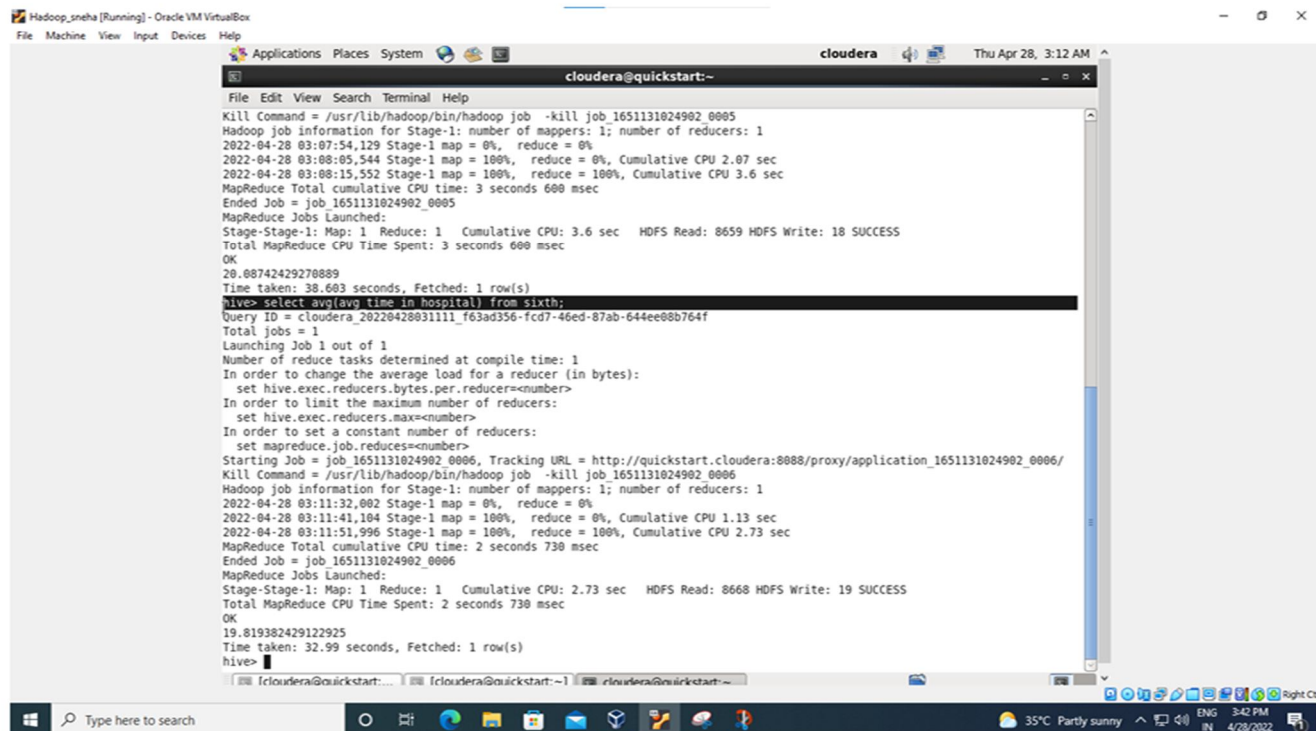
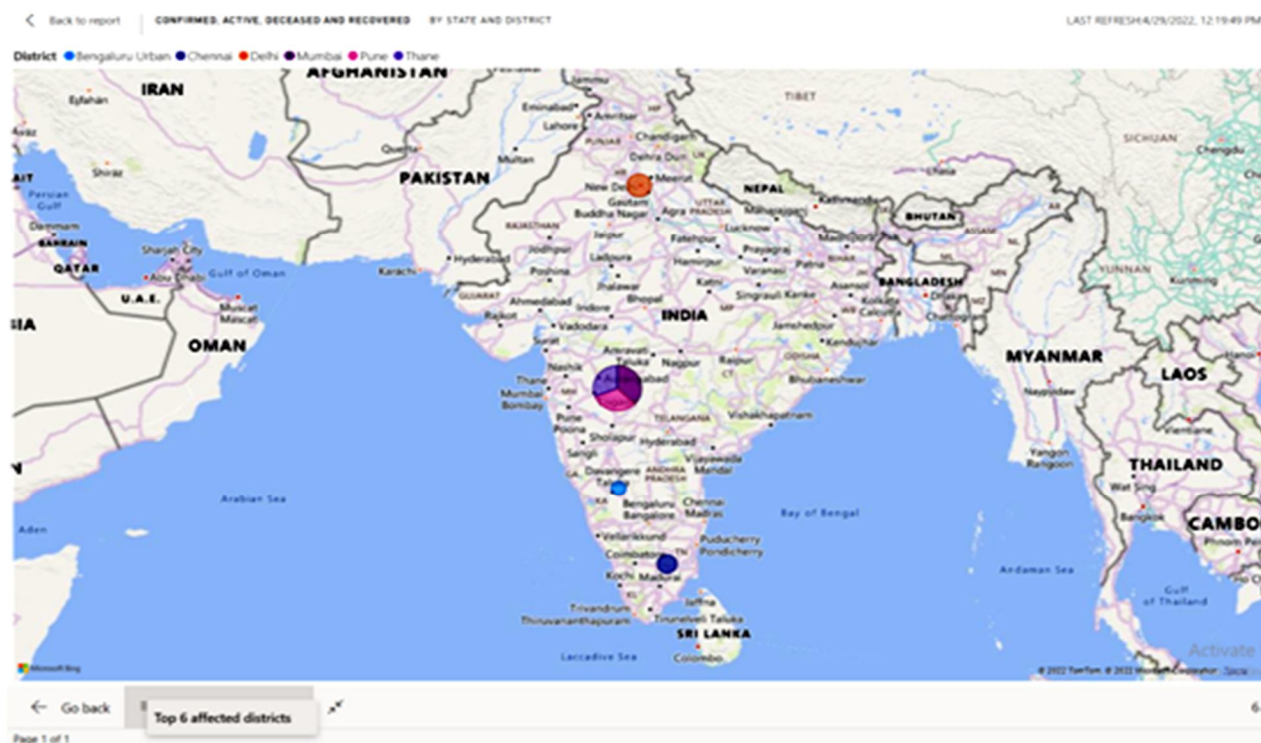


Fig 4: Identification of the average time a patient stays in hospital

A. Visualization of COVID-19 in India

1) Requirement - 2

Identification of patient demographics to understand the spread and trend in daily cases treatment



A Power BI Dataset can work as a collection of data for use in Power BI reports, and can either be connected to or imported into a Power BI Report. A Dataset can be connected to and get their source data through one or more Data flows.

The tool we used to develop this visualisation is power bi. To identify the top 6 districts we used map view as a visual from the entire state data we got this total 6 districts as most increased cases.

From the top 6 lists



Delhi has cases more cases compare to the other districts

Total cases : 140743

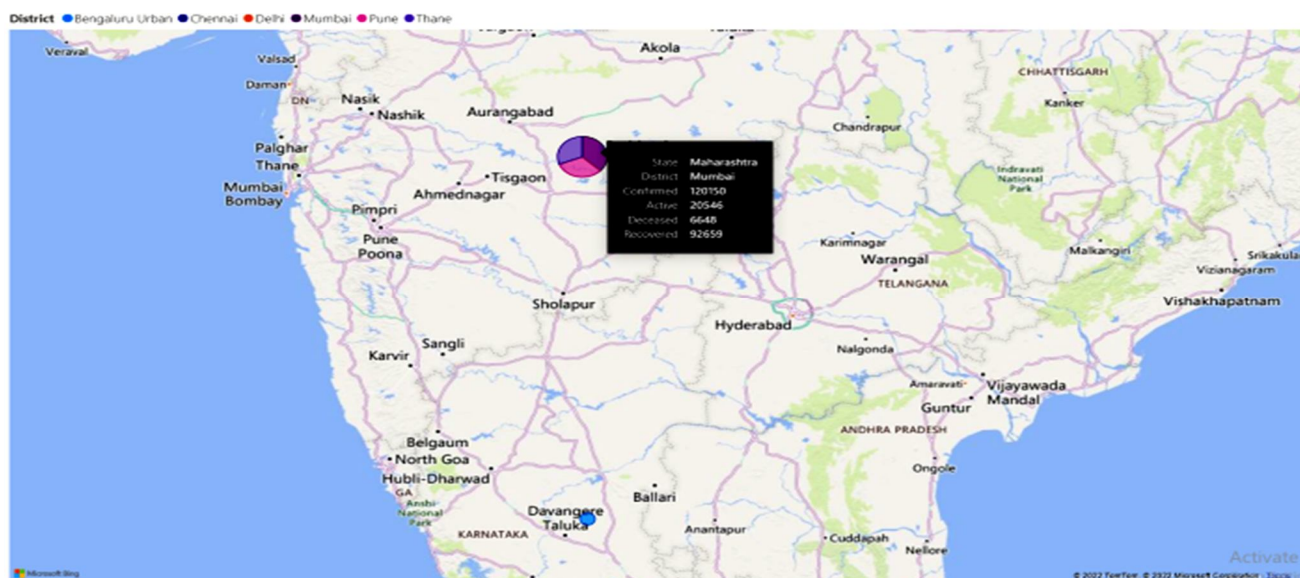
Present active cases are 9561

Decreased cases are 4058

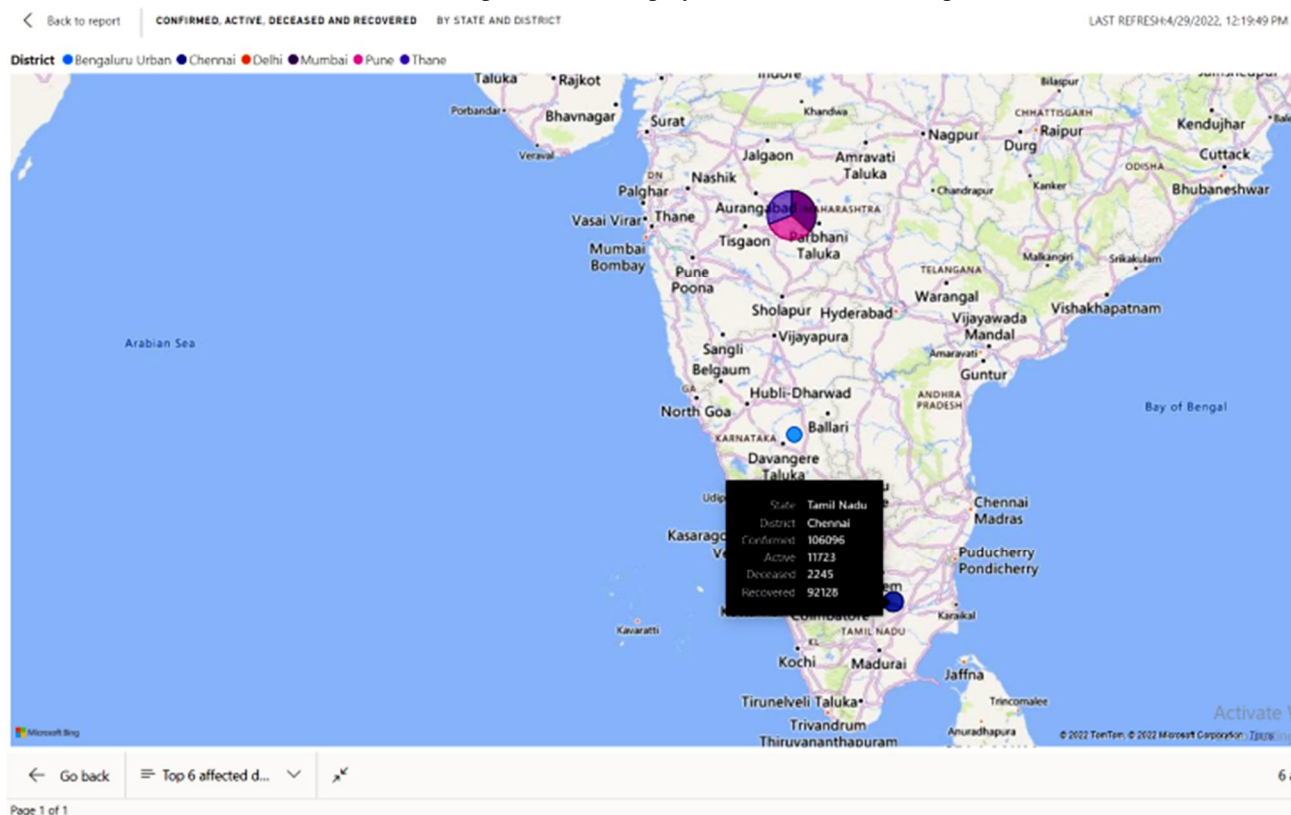
Total recovered cases are 127124

Delhi in district made up 22.00% of confirmed cases out of 6 districts.

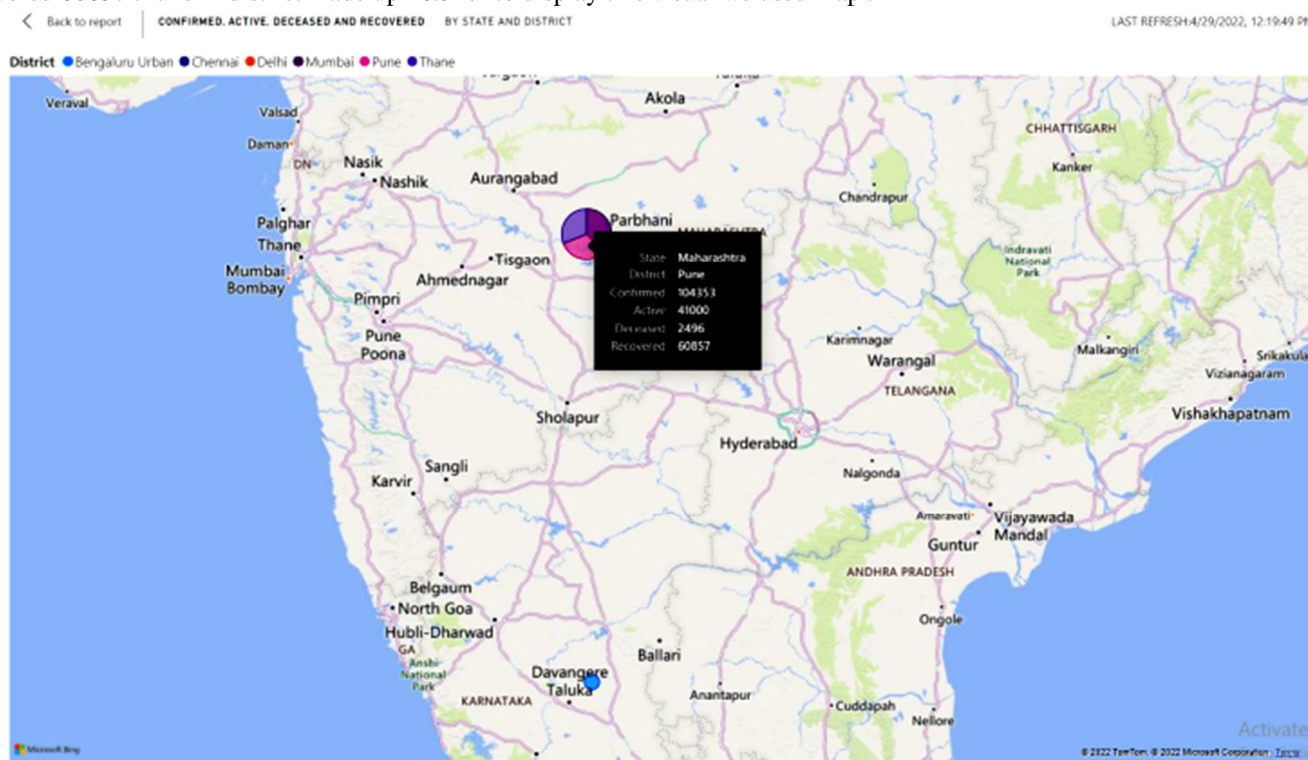
2nd highest cases are in Maharashtra state in Mumbai district with the confirmed cases of 120150. Total active are 20546, decreased 6648, recovered 92659 Mumbai in district made up 18.78% to display this visual we used map .



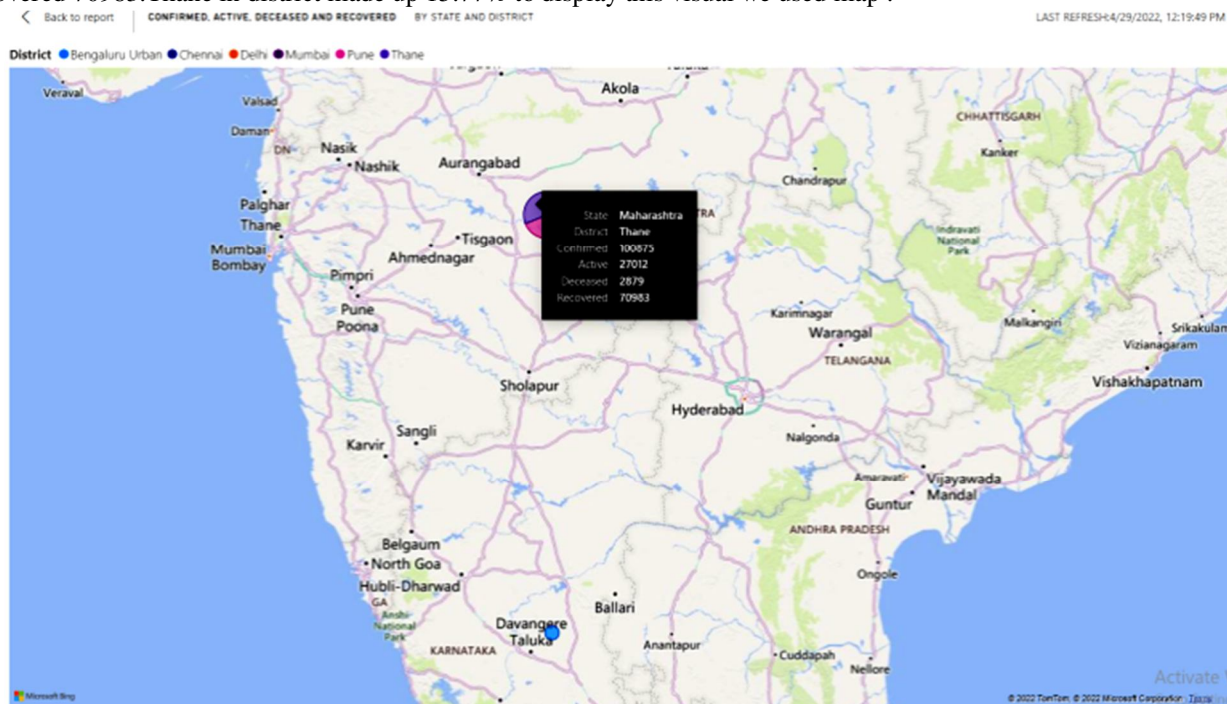
3rd highest cases are in Tamil nadu state in Chennai district with the confirmed cases of 106096. Total active are 11723 , decreased 2245, recovered 92128 .Chennai in district made up 16.59% to display this visual we used map.



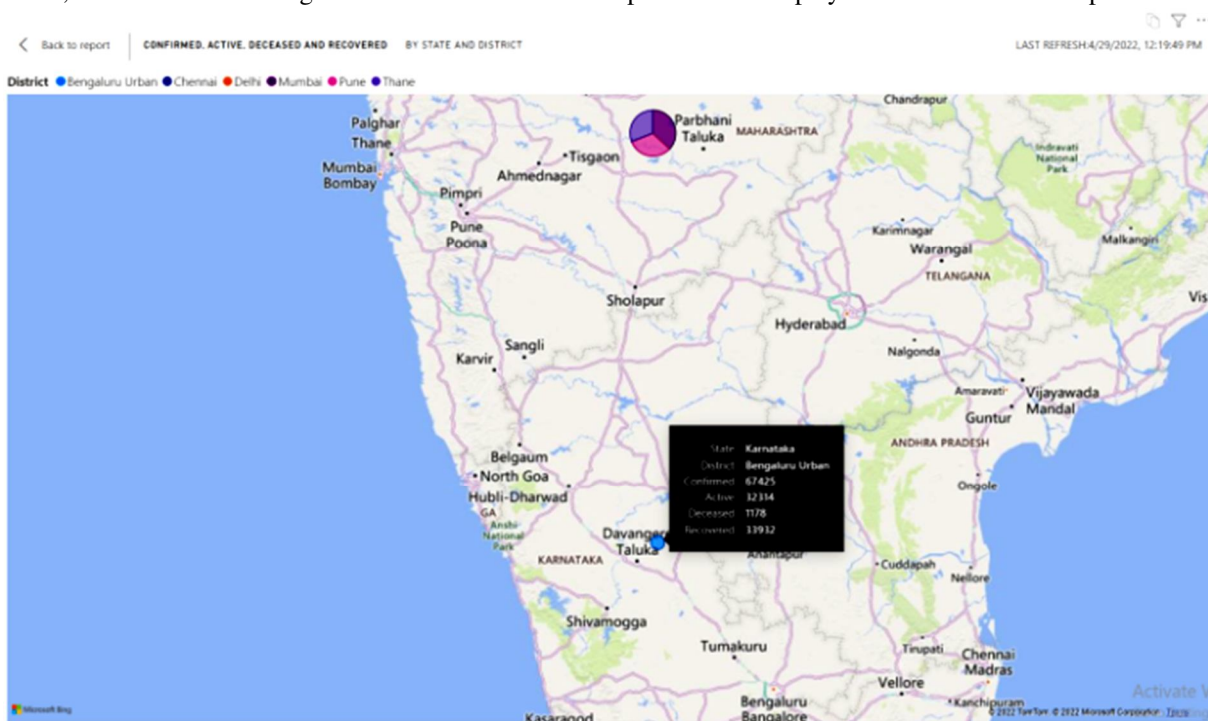
4th highest cases are in Maharashtra state in Pune district with the confirmed cases of 104353. total active are 41000 , decreased 2496, recovered 60857 .Pune in district made up 16.31% to display this visual we used map .



5th highest cases are in Maharashtra state in Thane district with the confirmed cases of 100875, total active are 27012, decreased 2879, recovered 70983. Thane in district made up 15.77% to display this visual we used map.



6th highest cases are in Karnataka state in bengaluru urban district with the confirmed cases of 67425, total active are 32314, decreased 1178, recovered 70983. bengaluru urban in district made up 15.77% to display this visual we used map.



The total cases in these districts are 639642, active 142156, decreased 19504, recovered 477683.

To improve this planning the visions on COVID-19 were improved as per the said requirements. The collected data for state is employed to understand the top 6 affected districts. One of the main things we've learned is that the faster all cases are found, tested

and isolated, the harder we make it for this virus to spread. This principle will save lives and mitigate the economic impact of the pandemic. The strategy requires a whole-of-government and whole-of-society response. The resolve and sacrifice of frontline health workers must be matched by every individual and every political leader to put in place the measures to end the pandemic.

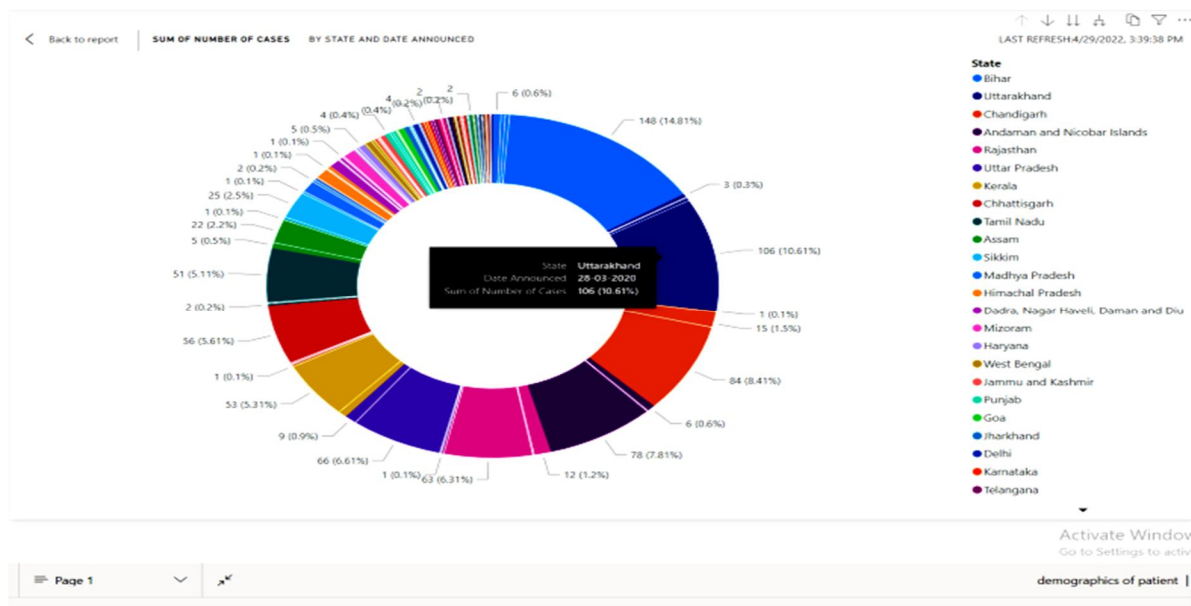
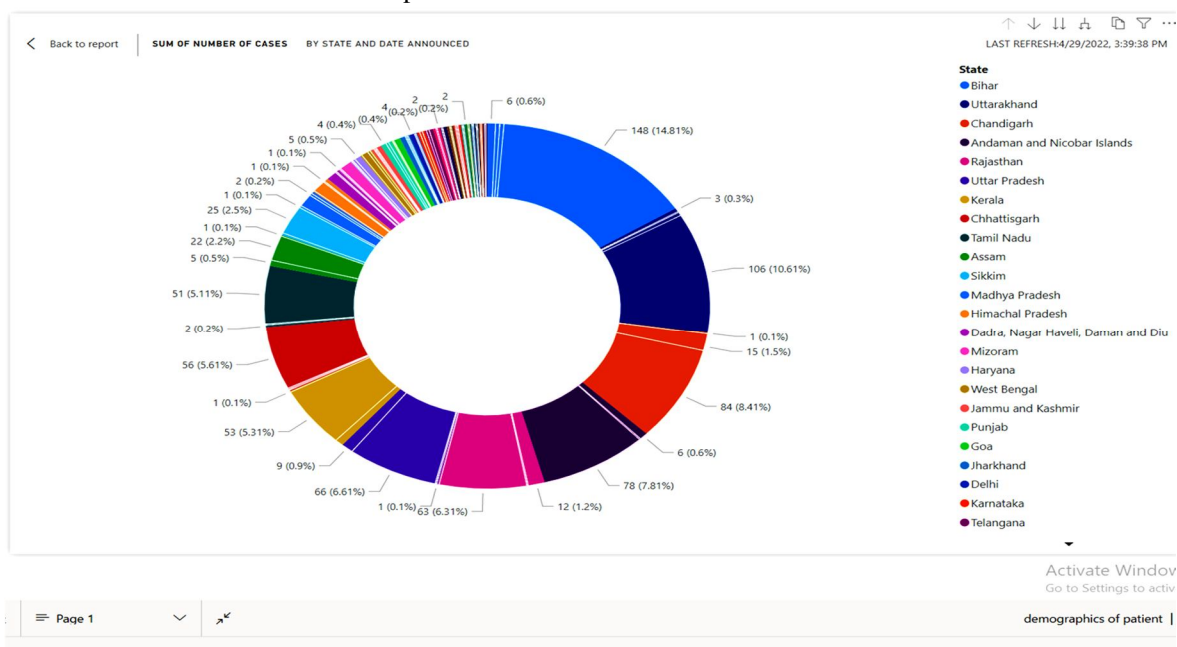
2) Requirement - 3

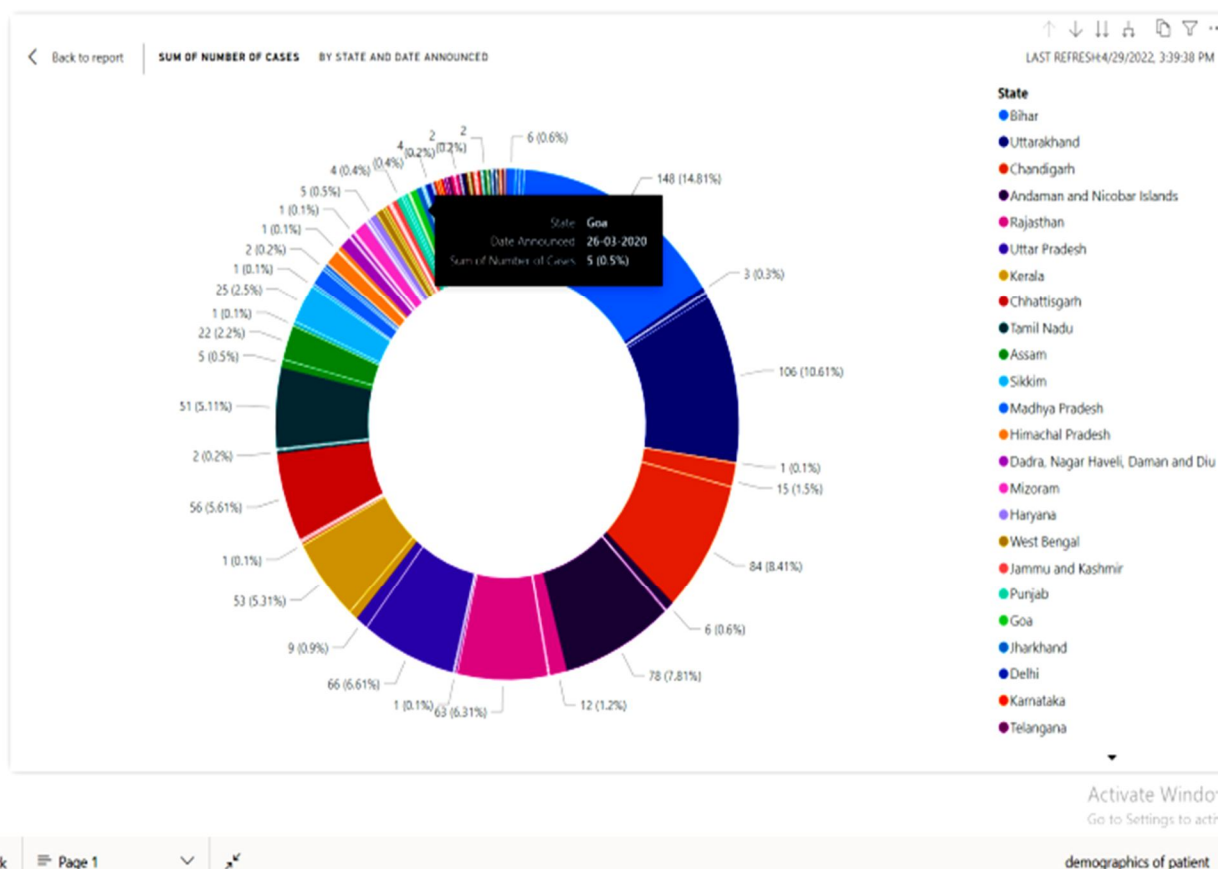
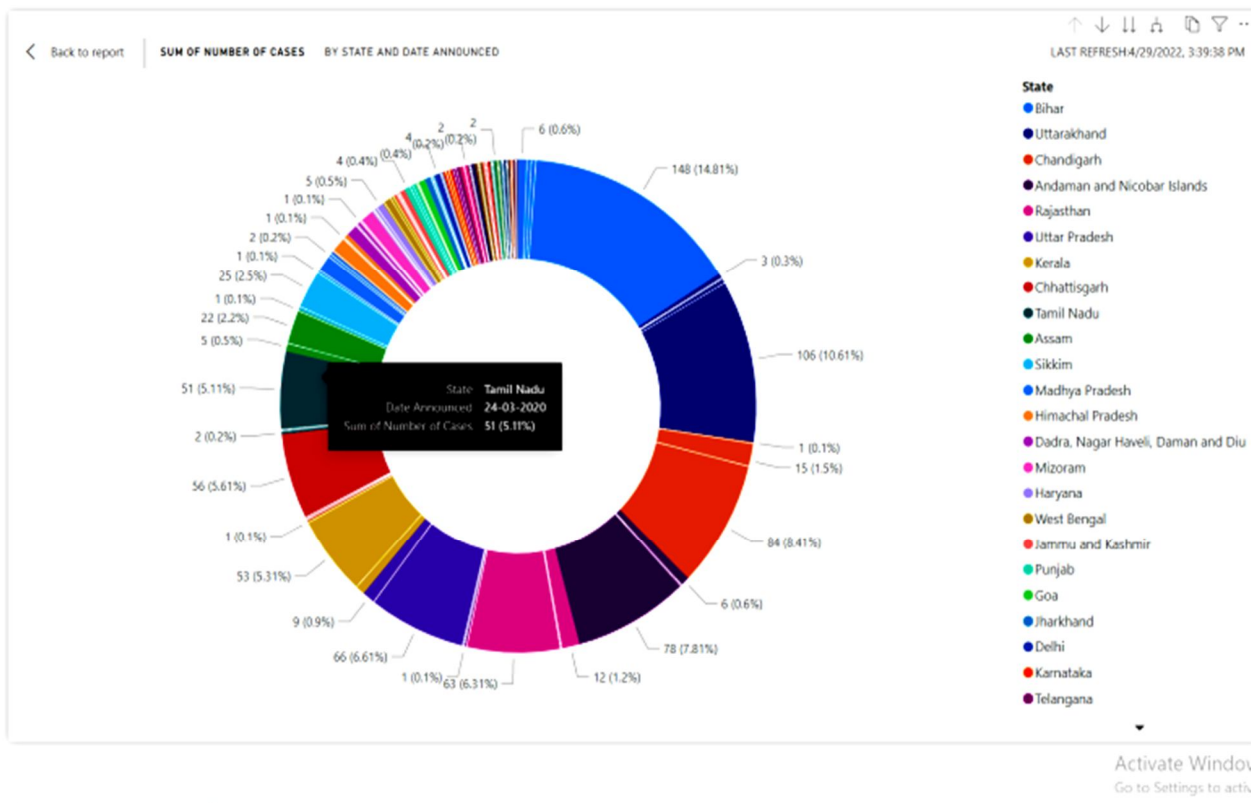
Identification of patient demographics to understand the spread and trend in daily cases treatment

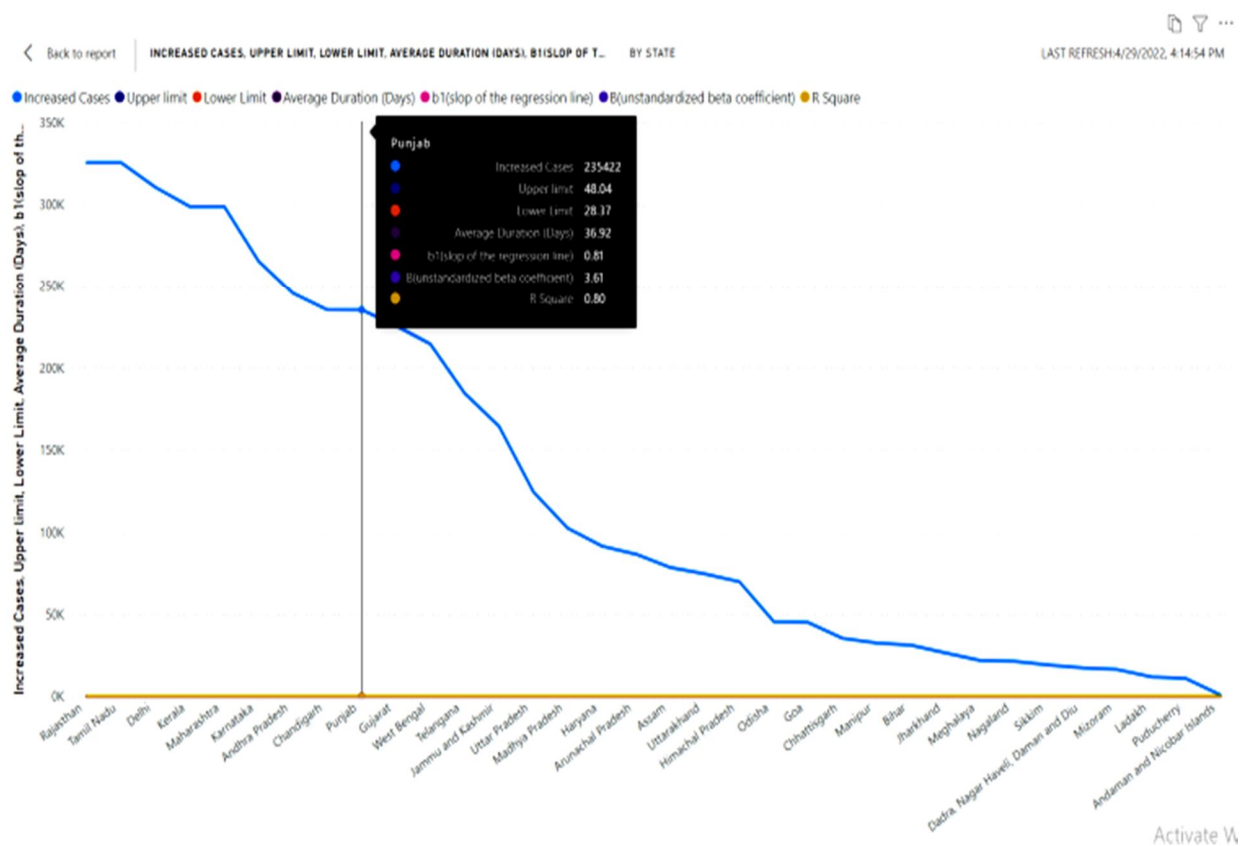
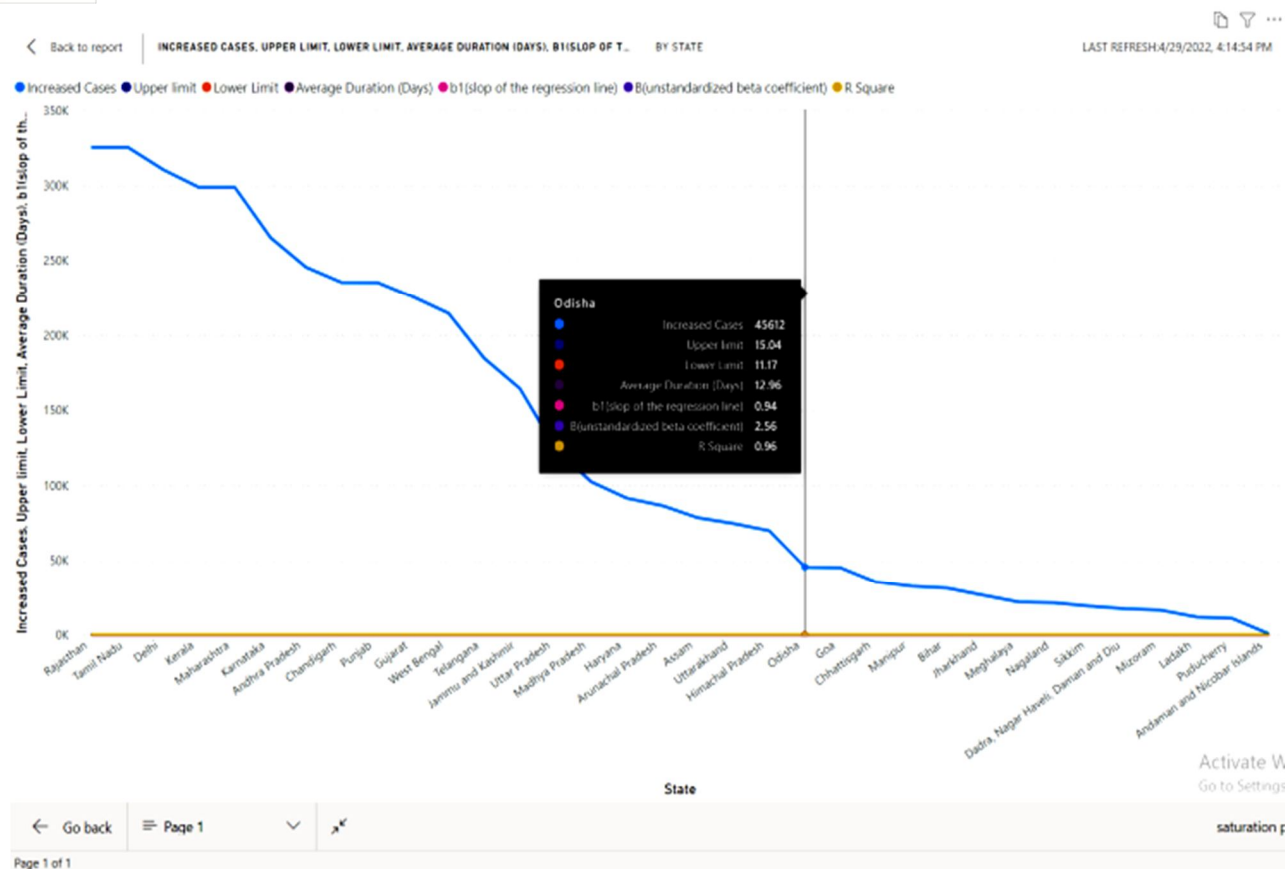
This visualization was developed by using Donut chart for clear visual of states. To prevent the spread and trend in the daily cases treatment, we need to minimize social and economic impact through multisectoral partnerships. communicate critical risk and event information to all communities and counter misinformation.

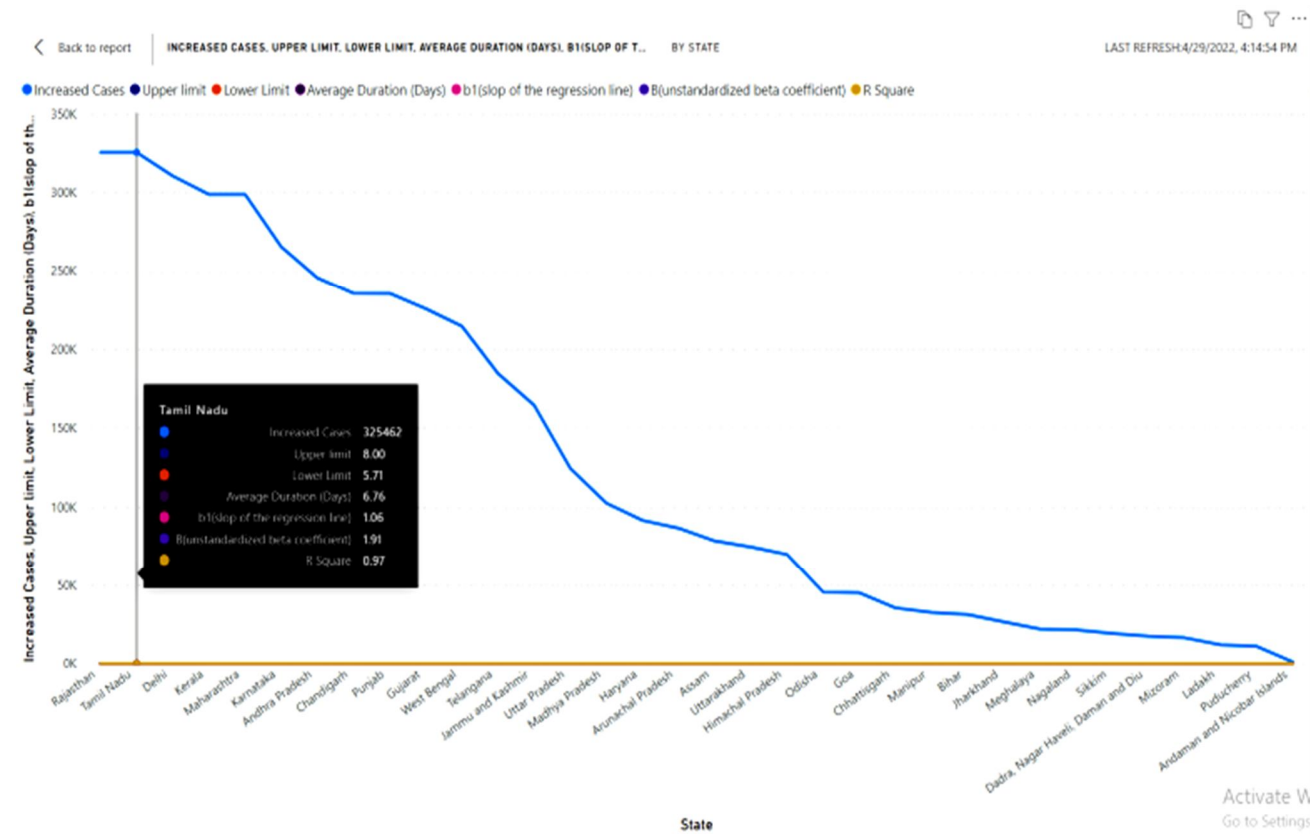
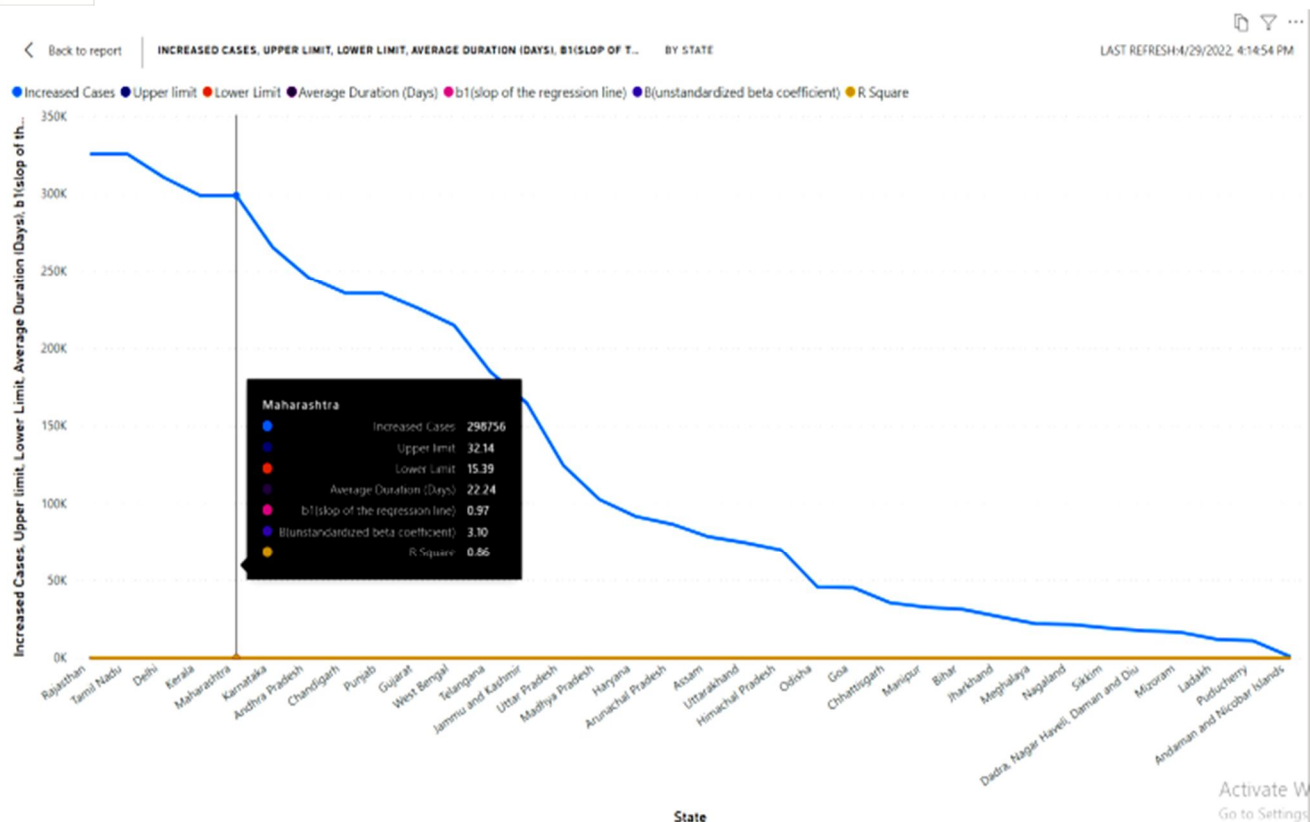
This can be achieved through a combination of public health measures, such as rapid identification, diagnosis and management of the cases, identification and follow up of the contacts, infection prevention and control in health care settings, implementation of health measures for travellers, awareness-raising in the population and risk communication.

Here in the above picture it shows the data visualization which is made.







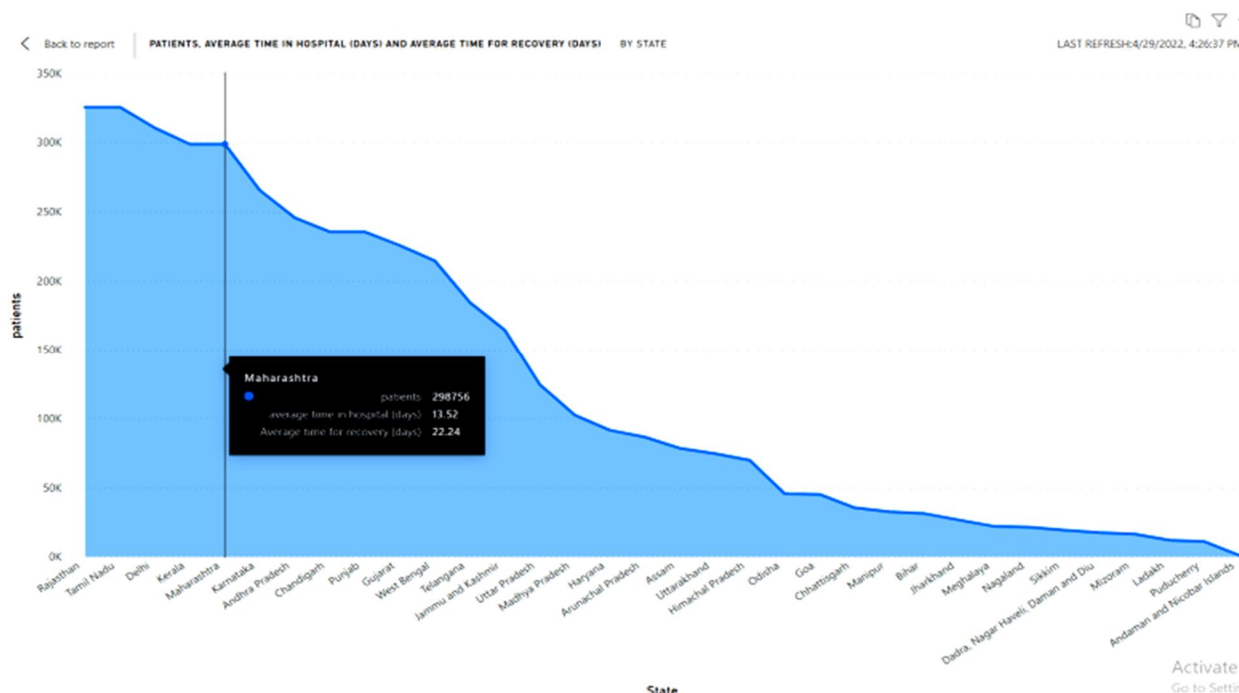
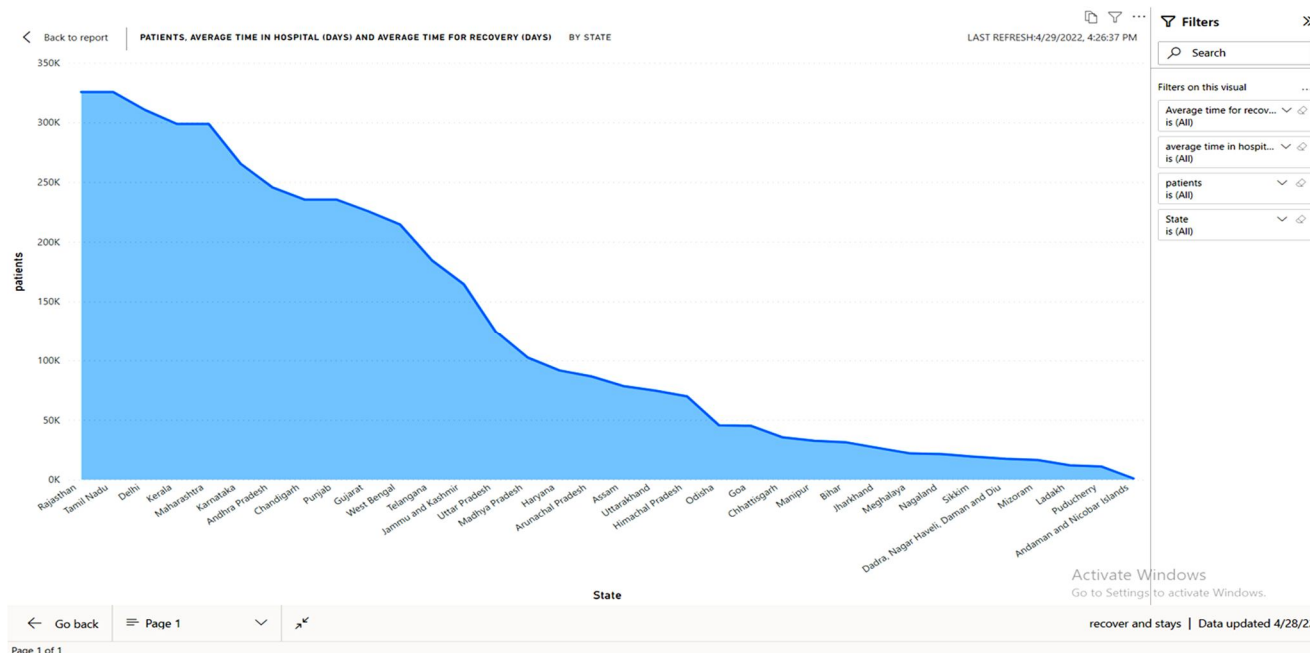


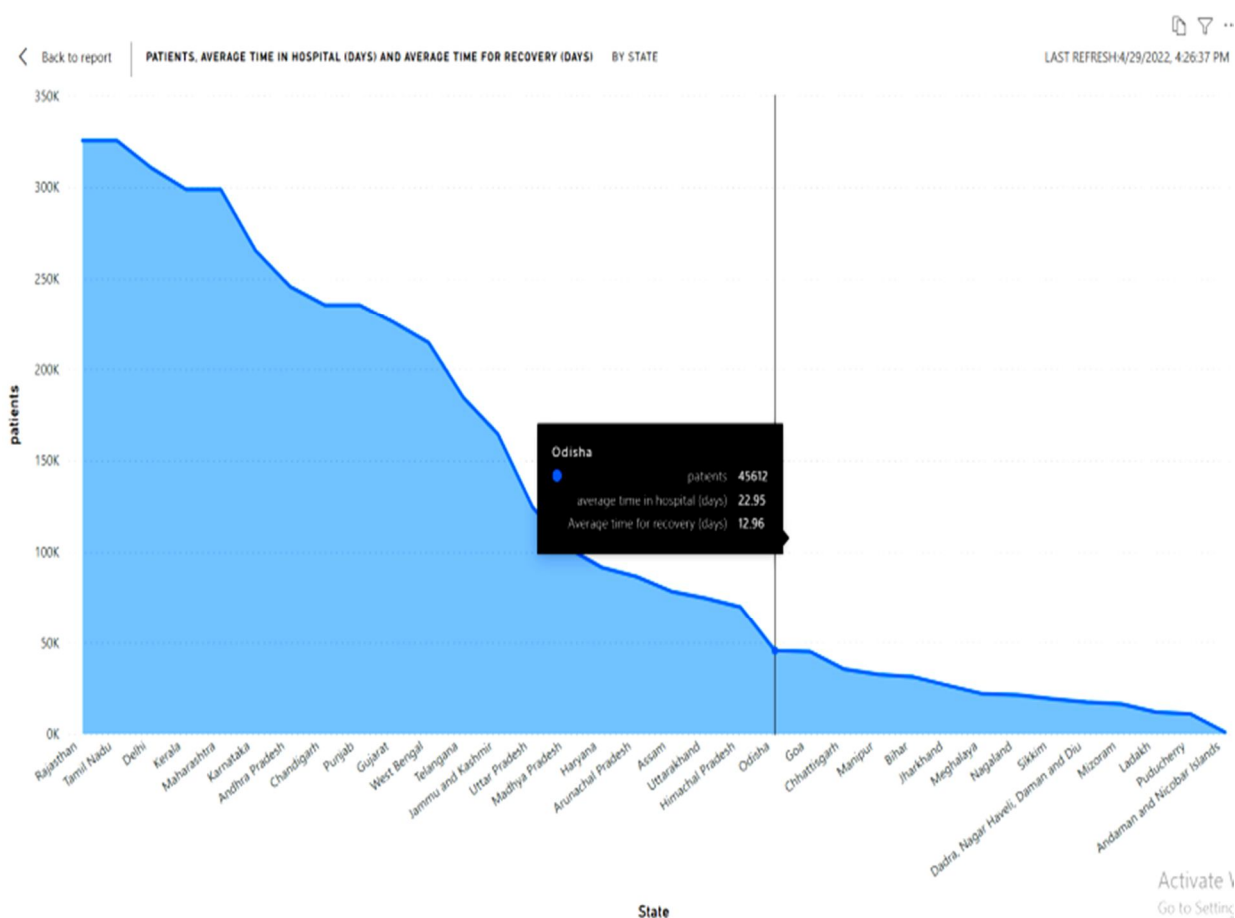
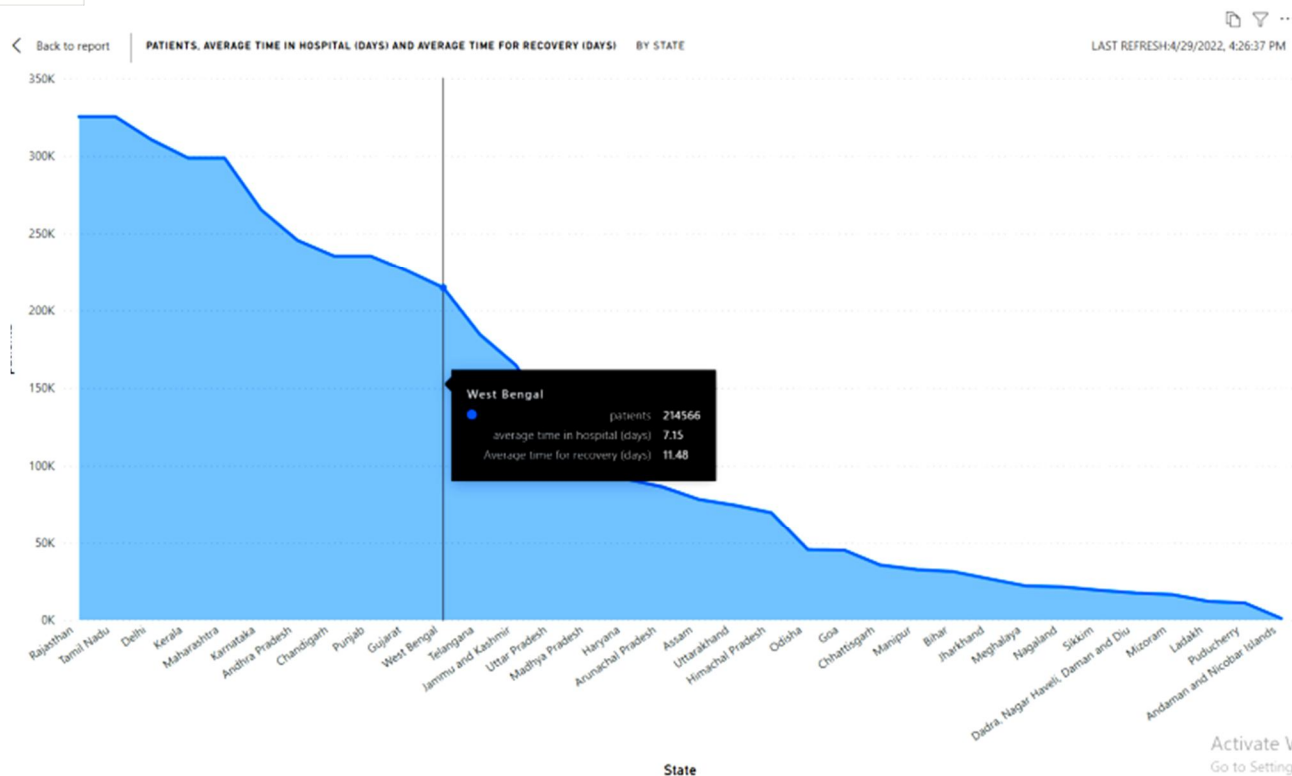
4) Requirement 5

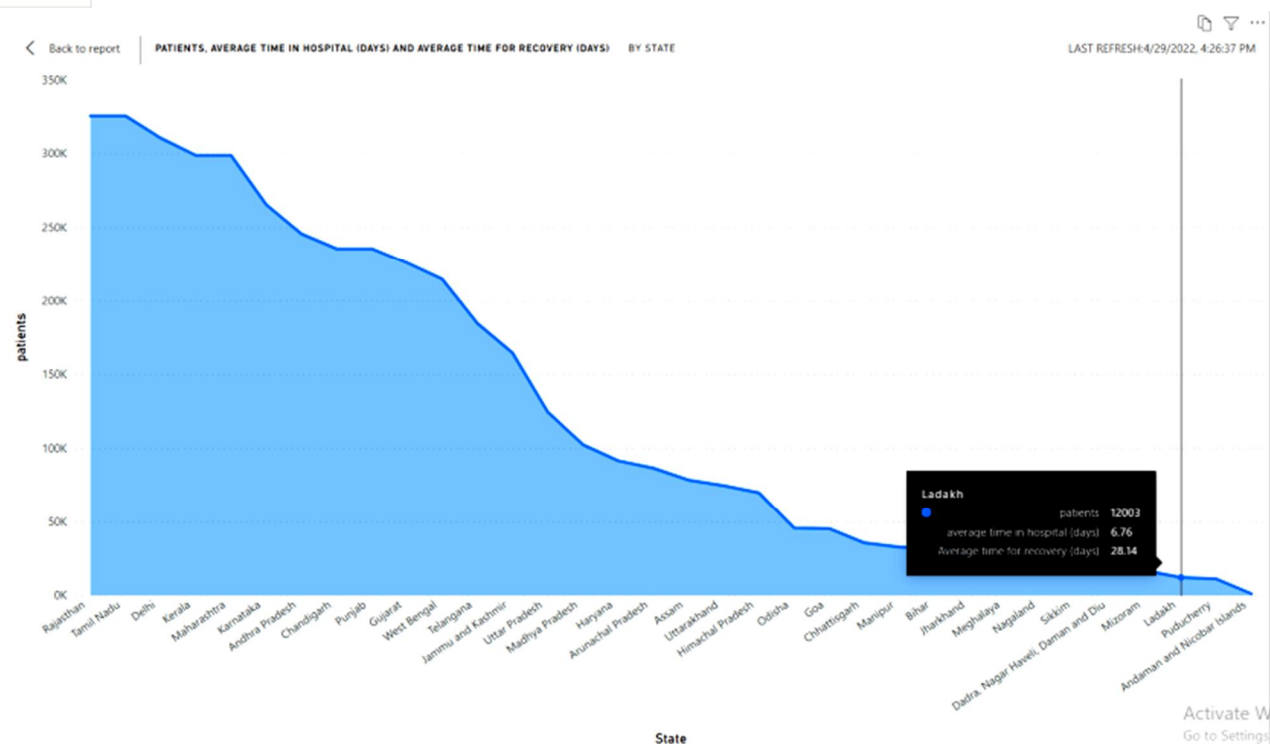
Average time taken by patients to recover, average time a patient stays in hospital-hospital occupancy.

The COVID-19 pandemic has placed an unprecedented strain on health systems, with rapidly increasing demand for healthcare in hospitals and intensive care units (ICUs) worldwide. As the pandemic escalates, determining the resulting needs for healthcare resources (beds, staff, equipment) has become a key priority for many countries. Projecting future demand requires estimates of how long patients with COVID-19 need different levels of hospital care.

We made a visual report to understand how much time patients took their time to recover and who stays in the hospital .average time of a patient is different in every state because it depends on the patients majority. The area chart shows the patients data that how many patients are staying in hospital and how many are recovered all these are calculated by the patients average rate in per state.







V. CONCLUSIONS

The volume of data increases dramatically over time, especially data generated on the global pandemic caused by COVID-19. Such volume of data requires utilizing big data analytics tools along with BI tool to make sense of the pandemic and its spread in a timely manner. In this study, we presented a review of several data analysis applications for COVID-19 to find key insights on patient demographics, patient clusters, state and district level spread and also to find transmission rate, top affected districts and predict the saturation point of the disease spread. Finally, we highlighted and discussed about track load on healthcare facility and predict number of ventilators required in future. The pandemic has exposed India's cherished inadequate medical infrastructure. India is a country of 1.3 billion people but around 75 percent of the healthcare infrastructure is focused in urban areas and making basic facilities inaccessible to rural areas. In this regards the data analytics can play a crucial role. It can bridge the gap between healthcare access and affordability across the country. So using this information, people of India can be prepared with a complete reformation within the healthcare sector led by digital technologies and the pandemic became a perfect time to effect this change.

REFERENCES

- [1] <https://www.kaggle.com/datasets>
- [2] https://www.kaggle.com/datasets/imdevskp/covid19-corona-virus-india-dataset?select=district_level_latest.csv
- [3] P. Ghosh, R. Ghosh, B. Chakraborty, "COVID-19 in India: Statewise Analysis and Prediction", *JMIR Public Health Surveill*, vol. 6(3), pp. e20341, 2020
- [4] A. Haleem, M. Javaid, I. Haleem Khan, and R. Vaishya., "Significant Applications of Big Data in COVID-19 Pandemic", *Indian J Orthop*. Vol. 54(4), pp. 526-528, 2020
- [5] S. J. Alsunaidi, A.M. Almuhaideb, N. M. Ibrahim, F. S. Shaikh, K. S. Alqudaihi, F. A. Alhaidari, I. Ullah Khan, N. Aslam, and M.d S. Alshahrani, "Applications of Big Data Analytics to Control COVID-19 Pandemic", 21(7): 2282, 2021.
- [6] A R Pradana, S R Madjid, H J Prayitno, R D Utami and Y Dharmawan , "Potential Applications of Big Data for Managing the COVID-19 Pandemic", *DOI* 10.1088/1742-6596/1720/1/012002.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)