Modern Big Data analysis with SQL

Coursera

A Training Report

Submitted is partial fulfilment of the requirements for the award of the degree of B. Tech Computer Science and Engineering (Hons.).

Submitted to

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA, PUNJAB

From 27/04/21 to 31/05/21

SUBMITTED BY

Yarragunta Sai Charan Reddy

Regn.no- 11905039

Student Declaration

To whom so ever it may concern

1.Yarragunta Sai Charan Reddy, Regn.no 11905039, Yarragunta Sai Charan Reddy, hereby declare that the work done by me on" Modern Big Data analysis with SQL" from 27/04/2021 to 31/05/21 is a record of original work for the partial fulfilment of the requirements for the award of the degree B. Tech Computer Science and Engineering (Hons.).

Name: Yarragunta Sai Charan Reddy

Regn.no- 11905039

Yarragunta Sai Charan Reddy

3 Courses

Foundations for Big Data Analysis with SQL

Analyzing Big Data with SQL

Managing Big Data in Clusters and Cloud Storage

CLOUDERA

May 31, 2021

Yarragunta Sai Charan Reddy

has successfully completed the online, non-credit Specialization

Modern Big Data Analysis with SQL

In this Specialization, learners acquired essential knowledge and skills for data analysis with SQL using open source distributed big data systems. Through a sequence of three courses, learners gained knowledge of the fundamental concepts behind relational databases, SQL, and big data; learned how to write and run SQL queries using query engines including Apache Hive and Apache Impala; and learned how to manage large-scale data in clusters and cloud storage using the Hadoop Distributed File System (HDFS) and Amazon Simple Storage Service (S3).

Glynn Durham
Senior Instructor
Cloudera

Ian Cook
Staff Curriculum Developer
Cloudera

Verify this certificate at:
coursera.org/verify/specialization/6X9GMGGU5PKH

ACKNOWLEDGEMENT

The opportunity of attaining a course based on Big Data Analysis through Coursera, provided by Cloudera was worth learning. It was a prestige for me to be a part of it. During the period of my course, I received tremendous knowledge related to big data, Analysis of Data and Structured Query Language (SQL).

Pre-eminently, I would like to express my deep gratitude and special thanks to my course instructors Prof. Glynn Durham and Prof. Ian Cook for their theoretical knowledge and encouragement on this course and for their valuable guidance and affection for the successful completion of this work.

Secondly, I would like to thank Coursera Staff for giving me an opportunity to learn this course by providing me financial aid.

Lastly, I would like to thank LPU for making this happen by organising the summer training project for us.

Sincerely,

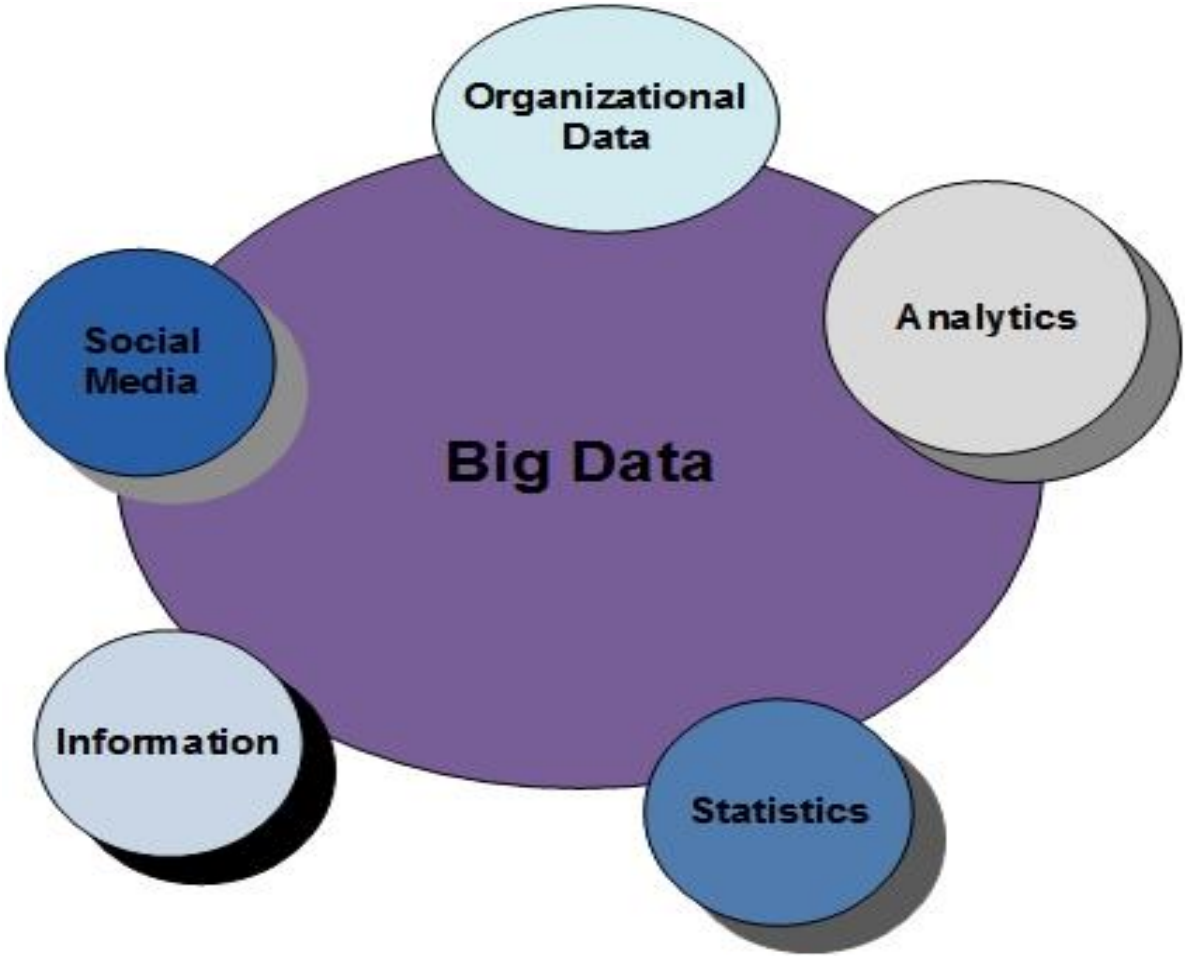
Yarragunta Sai Charan Reddy

Regn.no: 11905039

TABLE OF CONENTS

INTRODUCTION

What is Big Data?



We produce a massive amount of data every day, whether we know about it or not. Every click on the internet, every bank transaction, every video we watch on YouTube, every mail we sent, every like on our Instagram post makes up data for tech companies. With such a massive amount of data being collected, it only makes sense for companies to use this data to understand their customers and their behaviour better. This is the reason why the popularity of big data has grown manifold over the last few years

Big data is a collection of massive and complex data sets and data volume that include the huge quantities of data, data management capabilities, social media analytics and real-time data. Big data analytics is the process of examining large amounts of data. There exist large amounts of heterogeneous digital data. Big data is about data volume and large data set's measured in terms of terabytes or petabytes. This phenomenon is called Bigdata. After examining of Bigdata, the data has been launched as Big Data analytics. In this paper, presenting the 5Vs characteristics of big data and the technique and technology used to handle big data.

The challenges include capturing, analysis, storage, searching, sharing, visualization, transferring and privacy violations. It can neither be worked upon by using traditional SQL queries nor can the relational database management system (RDBMS) be used for storage.

Though, a wide variety of scalable database tools and techniques has evolved. Hadoop is an open-source distributed data processing is one of the prominent and well-known solutions.

Big Data is something so huge and complex that it is impossible for traditional data warehousing tools to process and work on them. Data (Big Data) is generated machines, generated by humans, and generated by mother nature. Within the growth of technologies and services, this large data is produced that can be structed, semi structured and unstructured from the different sources. Big data can neither be worked upon by using traditional SQL like queries nor can the relational database management system (RDBMS) be used for storage. So that a wide variety of scalable database tools and techniques have evolved. Hadoop, an open-source distributed data processing system is one of the prominent and well-known solutions.

Types of Big Data:

Big Data is classified into 3 types. They are

1.Structured Data

2.Unstructured Data

3.Semi-structured Data

Structured Data

Any data that can be stored, accessed, and processed in the form of fixed format is termed as a 'structured' data. Structured data includes quantitative data that is stored in an organized manner. It consists of numerical and text data. It is easy to analyse and process structured data. It is generally stored in a relational database and can be queried using Structured Query Language (SQL). Examples are dates, phone numbers, addresses, transaction information etc

Unstructured Data

Any data with unknown form or the structure is classified as unstructured data. In addition to the size being huge, un-structured data poses multiple challenges in terms of its processing for deriving value out of it. A typical example of unstructured data is a heterogeneous data source containing a combination of simple text files, images, videos etc. Unstructured data is said to lack "structure". It is stored in a non-relational database and can be queried using NoSQL. Examples are text files, audio files, video files etc.

Semi-structured Data

Semi-structured data can contain both the forms of data. We can see semi-structured data as a structured in form, but it is actually not defined with e.g., a table definition in relational DBMS. Example of semi-structured data is a data represented in an XML file. Examples are server logs, email sorting by folders etc.

Characteristics of Big Data:

Big data can be described by the following characteristics:

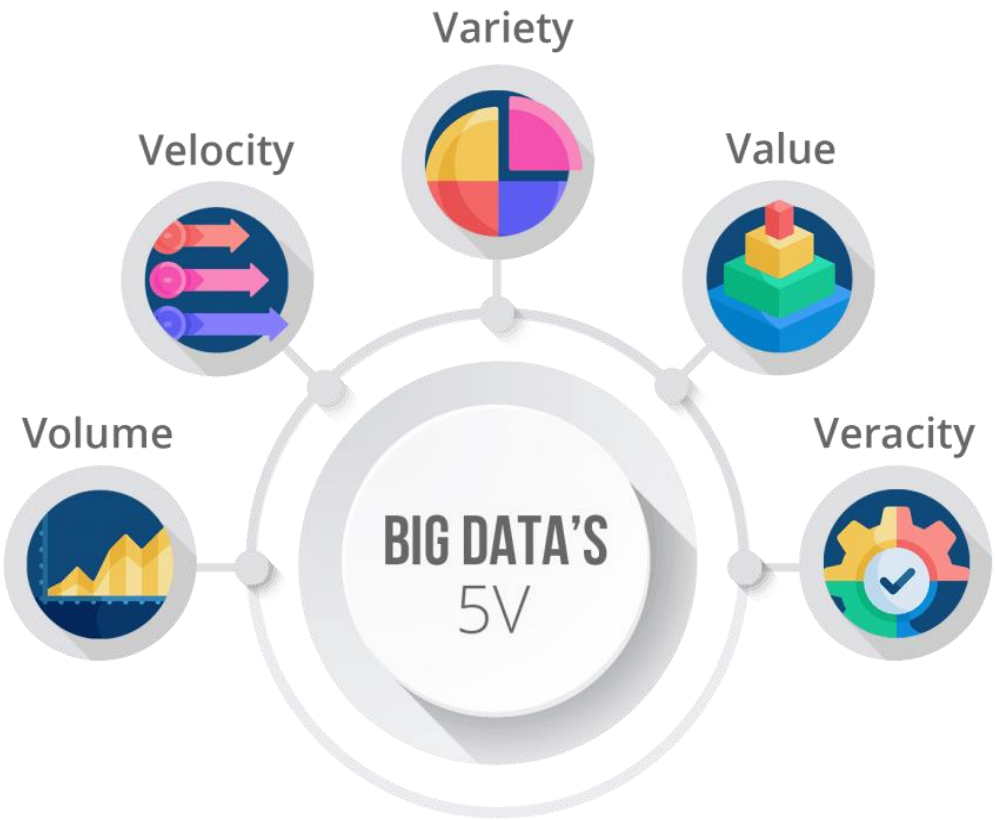
Volume refers to the amount of data that is being collected. The data could be structured or unstructured.

Velocity refers to the rate at which data is coming in. the speed of generation of data. How fast the data is generated and processed to meet the demands, determines real potential in the data.

Variety refers to the different kinds of data (data types, formats, etc.) that is coming in for analysis. Data in the form of emails, photos, videos, monitoring devices, PDFs, audio, etc. are also being considered in the analysis applications This variety of unstructured data poses certain issues for storage, mining and analysing data.

Value refers to the usefulness of the collected data.

Veracity refers to the quality of data that is coming in from different sources. data in the form of emails, photos, videos, monitoring devices, PDFs, audio, etc. are also being considered in the analysis applications.



Examples of Big data:

1.The **New York Stock Exchange** is an example of Big Data that generates about one terabyte of new trade data per day.

2.The statistic shows that 500+terabytes of new data get ingested into the databases of social media site **Facebook**, every day. This data is mainly generated in terms of photo and video uploads, message exchanges, putting comments etc.

3. A single **Jet engine** can generate 10+terabytes of data in 30 minutes of flight time. With many thousand flights per day, generation of data reaches up too many Petabytes.

How Big Data Works

Big data analytics refers to collecting, processing, cleaning, and analyzing large datasets to help organizations operationalize their big data.

1. Collect Data

Data collection looks different for every organization. With today's technology, organizations can gather both structured and unstructured data from a variety of sources — from cloud storage to mobile applications to in-store IoT sensors and beyond. Some data will be stored in data warehouses where business intelligence tools and solutions can access it easily. Raw or unstructured data that is too diverse or complex for a warehouse may be assigned metadata and stored in a data lake.

2. Process Data

Once data is collected and stored, it must be organized properly to get accurate results on analytical queries, especially when it's large and unstructured. Available data is growing exponentially, making data processing a challenge for organizations. One processing option is batch processing, which looks at large data blocks over time. Batch processing is useful when there is a longer turnaround time between collecting and analyzing data. Stream processing looks at small batches of data at once, shortening the delay time between collection and analysis for quicker decision-making. Stream processing is more complex and often more expensive.

3. Clean Data

Data big or small requires scrubbing to improve data quality and get stronger results; all data must be formatted correctly, and any duplicative or irrelevant data must be eliminated or accounted for. Dirty data can obscure and mislead, creating flawed insights.

4. Analyze Data

Getting big data into a usable state takes time. Once it's ready, advanced analytics processes can turn big data into big insights. Some of these big data analysis methods include:

- Data mining sorts through large datasets to identify patterns and relationships by identifying anomalies and creating data clusters.
- Predictive analytics uses an organization's historical data to make predictions about the future, identifying upcoming risks and opportunities.
- Deep learning imitates human learning patterns by using artificial intelligence and machine learning to layer algorithms and find patterns in the most complex and abstract data.

Benefits and Advantages of Big Data

1. Risk Management

Banco de Oro, a Philippine banking company, uses Big Data analytics to identify fraudulent activities and discrepancies. The organization leverages it to narrow down a list of suspects or root causes of problems.

2. Product Development and Innovations

Rolls-Royce, one of the largest manufacturers of jet engines for airlines and armed forces across the globe, uses Big Data analytics to Analyze how efficient the engine designs are and if there is any need for improvements.

3. Quicker and better decision making within Organisation

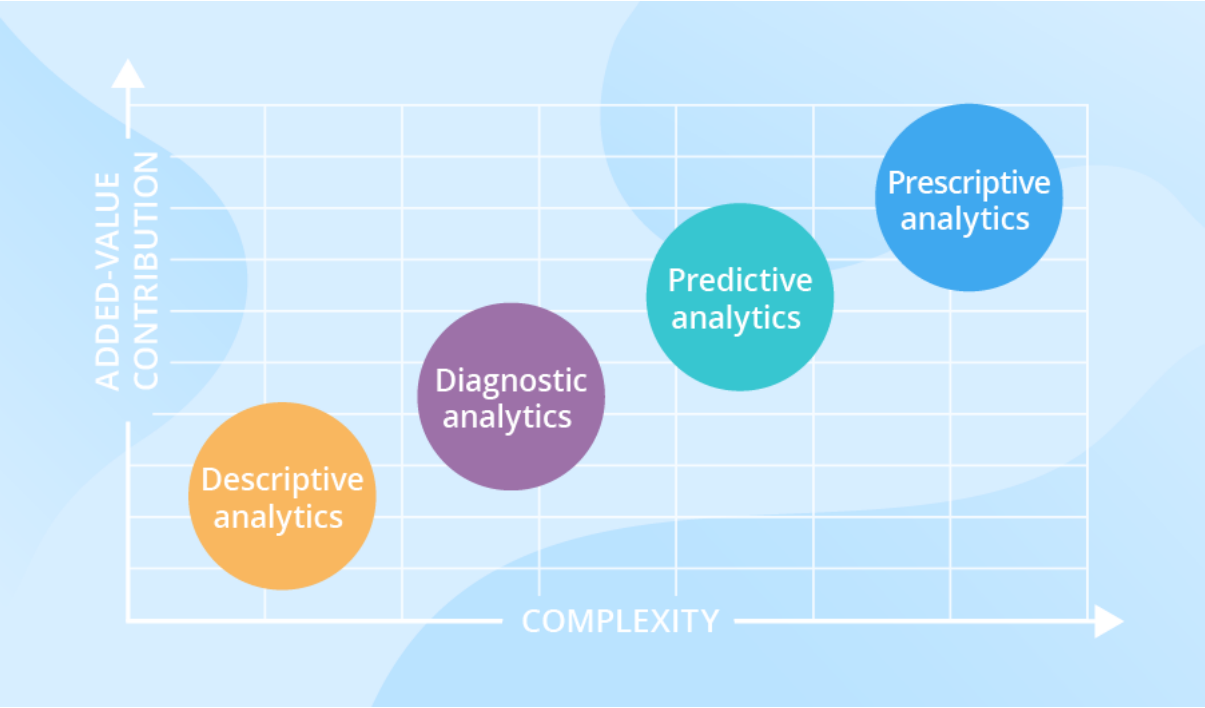
Starbucks uses Big Data analytics to make strategic decisions. For example, the company leverages it to decide if a particular location would be suitable for a new outlet or not. They will Analyze several different factors, such as population, demographics, accessibility of the location, and more.

4.Improve Customer Experience

Delta Air Lines uses Big Data analysis to improve customer experiences. They monitor tweets to find out their customers' experience regarding their journeys, delays, and so on. The airline identifies negative tweets and does what's necessary to remedy the situation. By publicly addressing these issues and offering solutions, it helps the airline build good customer relations.

Different Types of Big Data Analytics

Here are the four types of Big Data analytics:



1.Descriptive Analytics

This summarizes past data into a form that people can easily read. This helps in creating reports, like a company's revenue, profit, sales, and so on. Also, it helps in the tabulation of social media metrics.

Use Case: The Dow Chemical Company analysed its past data to increase facility utilization across its office and lab space. Using descriptive analytics, Dow was able to identify underutilized space. This space consolidation helped the company save nearly US $4 million annually.

2. Diagnostic Analytics

This is done to understand what caused a problem in the first place. Techniques like drill-down, data mining, and data recovery are all examples. Organizations use diagnostic analytics because they provide an in-depth insight into a particular problem.

Use Case: An e-commerce company's report shows that their sales have gone down, although customers are adding products to their carts. This can be due to various reasons like the form didn't load correctly, the shipping fee is too high, or there are not enough payment options available. This is where you can use diagnostic analytics to find the reason.

3. Predictive Analytics

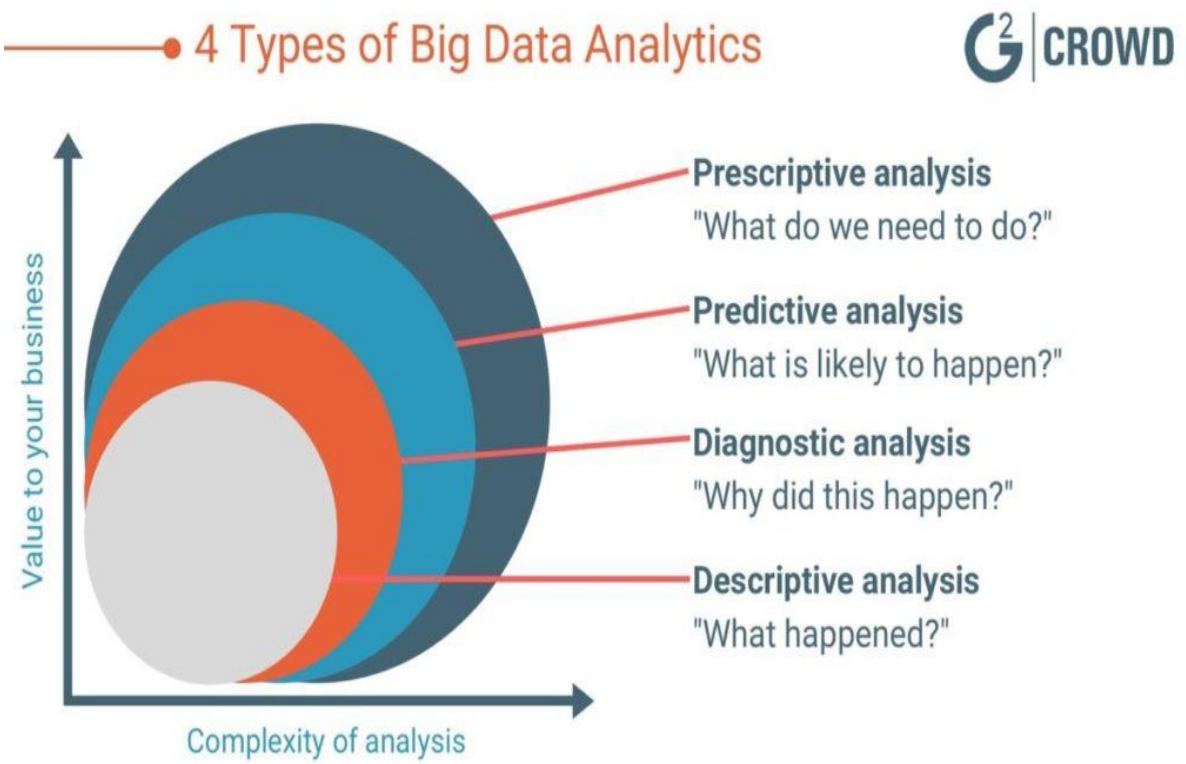
This type of analytics investigates the historical and present data to make predictions of the future. Predictive analytics uses data mining, AI, and machine learning to analyse current data and make predictions about the future. It works on predicting customer trends, market trends, and so on.

Use Case: PayPal determines what kind of precautions they must take to protect their clients against fraudulent transactions. Using predictive analytics, the company uses all the historical payment data and user behaviour data and builds an algorithm that predicts fraudulent activities.

4. Prescriptive Analytics

This type of analytics prescribes the solution to a particular problem. Perspective analytics works with both descriptive and predictive analytics. Most of the time, it relies on AI and machine learning.

Use Case: Prescriptive analytics can be used to maximize an airline's profit. This type of analytics is used to build an algorithm that will automatically adjust the flight fares based on numerous factors, including customer demand, weather, destination, holiday seasons, and oil prices.



Objectives of Big Data

The importance of big data doesn't revolve around how much data you have, but what you do with it. You can take data from any source and analyse it to find answers that enable

1) cost reductions

2) time reductions

3) new product development and optimized offerings

4) smart decision making

To make use of the humongous amount of data available to us to make the working and projecting of our day-to-day activities more efficient and fruitful and rightly so big data has indeed provided us with some extremely helpful predictions. Be it climate change, disaster management or marketing, big data has taken the centre stage.

1.Big data helps align and organize a company better.

2.Big data helps to take decision based on some pattern and model.

3.Big data helps data analytics become more accurate.

4.The objective of big data is to challenge the traditional data analytic tools and say “Hey you don’t have ability to analyse me” AND thereby giving opportunities to software companies, opensource communities to develop cutting edge tools.

Scope Of Big Data

With the increasing size of data every day, all companies have started using Big Data technologies to analyse their data and get insights from it. As the data size will increase in future as well, so undoubtedly big data is going to provide you future proof career and will have huge scope ahead as well.

The scope of Big Data today is inexplicable, given that it has delivered proven results in decision making and managing the deluge of information that piles up in organizations.

The scope of big data lies in its telling potential to reduce complexities in the organization’s infrastructure and production systems.

Big data is influencing the IT industry like few technologies or trends have done so before. If analysed effectively, massive information caches can help companies improve their decision-making and compete on another level. However, managing big data is a difficult endeavour, according to a recent report by Microsoft.

"Big data absolutely has the potential to change the way governments, organizations, and academic institutions conduct business and make discoveries, and its likely to change how everyone lives their day-to-day lives

he future of big data is bright. Per IBM, 90% of the data that we have in the world today has been generated in last 2 years!! Every day we are generating 2.5 Quintilian Bytes (2,500,000 Terabytes) of data**.** This data comes in from all over the place such as social media, sensors, transactions, pictures, videos and so on. The growth of this data is expected to be even faster in coming decades.

Business**:** big data provide the ability to pursue new business models or to achieve a significant competitive advantage on the company's traditional business.
Finance**:** several cases in which big data was used show that it brings economic advantages to the companies that have adopted such solutions. It is necessary, however, to assess in advance the costs of implementing these solutions. An important element to consider is the availability, on the one hand of technologies which use low-cost hardware, and on the other of cloud solutions which allow to minimize the initial investment. We will address these issues in the section about big data ROI.

Software Requirement Analysis

This Big Data Analysis course is organized by the Cloudera platform. And they have their unique set of software of their own for the big data analysis which are open source and used widely all over the world.

In this course we were taught Impala and Hive in the unique querying environment called "Hue" which is developed by the Cloudera.

So, throughout this project I have used:

- Hue
  - Impala

After querying, I had exported the results in the form of CSV file and made the graphs in

- Excel

SQL

SQL (Structured Query Language) is a standardized programming language that's used to manage relational databases and perform various operations on the data in them. Initially created in the 1970s, SQL is regularly used not only by database administrators, but also by developers writing data integration scripts and data analysts looking to set up and run analytical queries.

The uses of SQL include modifying database table and index structures; adding, updating and deleting rows of data; and retrieving subsets of information from within a database for transaction processing and analytics applications. Queries and other SQL operations take the form of commands written as statements -- commonly used SQL statements include select, add, insert, update, delete, create, alter, and truncate.

- SQL lets you access and manipulate databases
- SQL is used to communicate with Data Base.

Structured Query Language) is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). It is particularly useful in handling structured data, i.e., data incorporating relations among entities and variables.

The SQL language is subdivided into several language elements, including:

- Clauses, which are constituent components of statements and queries. (In some cases, these are optional.)
- Expressions, which can produce either scalar values, or tables consisting of columns and rows of data
- Predicates, which specify conditions that can be evaluated to SQL three-valued logic (3VL) (true/false/unknown) or Boolean truth values and are used to limit the effects of statements and queries, or to change program flow.
- Queries, which retrieve the data based on specific criteria. This is an important element of *SQL*.
- Statements, which may have a persistent effect on schemata and data, or may control transactions, program flow, connections, sessions, or diagnostics.
    - SQL statements also include the semicolon (";") statement terminator. Though not required on every platform, it is defined as a standard part of the SQL grammar.
- Insignificant whitespace is generally ignored in SQL statements and queries, making it easier to format SQL code for readability.

SQL is Structured Query Language, which is a computer language for storing, manipulating, and retrieving data stored in a relational database.

SQL is the standard language for Relational Database System. All the Relational Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, Postgres, and SQL Server use SQL as their standard database language.

- Allows users to access data in the relational database management systems.
- Allows users to describe the data.
- Allows users to define the data in a database and manipulate that data.
- Allows to embed within other languages using SQL modules, libraries & pre-compilers.
- Allows users to create and drop databases and tables.
- Allows users to create view, stored procedure, functions in a database.
- Allows users to set permissions on tables, procedures, and views.

Technology learnt

Throughout this course, I have learnt an ample number of technologies related to big data or data analytics. These skills or knowledge about these technologies are well used in the large industries who manage data. And are very useful to ace in the analytics field.

The technologies I have learnt are the following:

Hive:

Hive is a data warehouse infrastructure tool to process structured data in Hadoop. It resides on top of Hadoop to summarize Big Data and makes querying and analyzing easy.

Initially Hive was developed by Facebook, later the Apache Software Foundation took it up and developed it further as an open source under the name Apache Hive. It is used by different companies. For example, Amazon uses it in Amazon Elastic MapReduce.

Features of Hive:

- It stores schema in a database and processed data into HDFS.
- It is designed for OLAP.
- It provides SQL type language for querying called HiveQL or HQL.
- It is familiar, fast, scalable, and extensible.

Impala:

Impala is a MPP (Massive Parallel Processing) SQL query engine for processing huge volumes of data that is stored in Hadoop cluster. It is an open-source software which is written in C++ and Java. It provides high performance and low latency compared to other SQL engines for Hadoop.

In other words, Impala is the highest performing SQL engine (giving RDBMS-like experience) which provides the fastest way to access data that is stored in Hadoop Distributed File System.

Why Impala?

Impala combines the SQL support and multi-user performance of a traditional analytic database with the scalability and flexibility of Apache Hadoop, by utilizing standard components such as HDFS, HBase, Meta store, YARN, and Sentry.

With Impala, users can communicate with HDFS or HBase using SQL queries in a faster way compared to other SQL engines like Hive.

Impala can read almost all the file formats such as Parquet, Avro, RCFile used by Hadoop. Impala uses the same metadata, SQL syntax (Hive SQL), ODBC driver, and user interface (Hue Beeswax) as Apache Hive, providing a familiar and unified platform for batch-oriented or real-time queries.

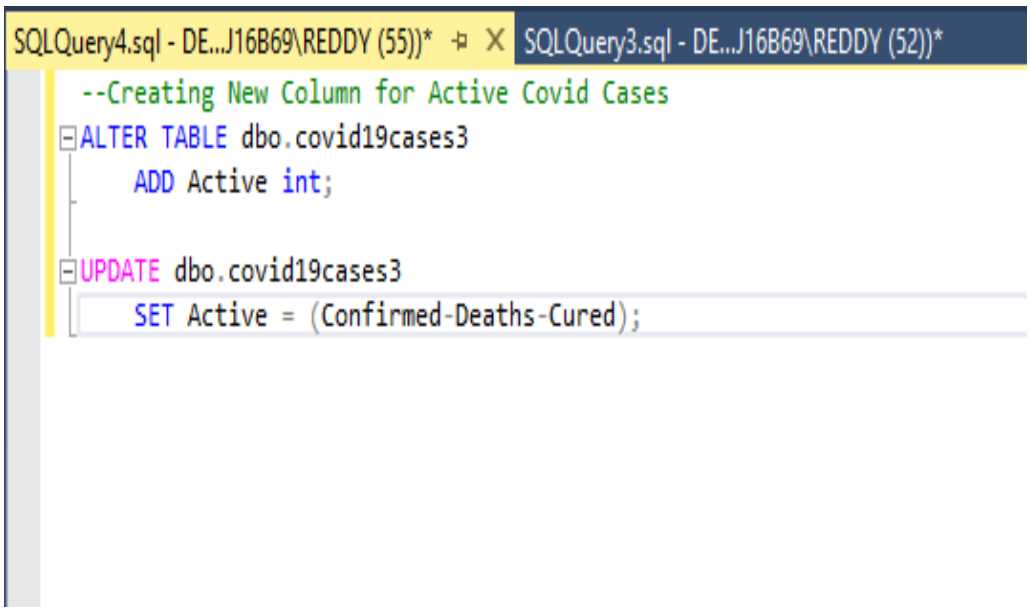
Unlike Apache Hive, Impala is not based on MapReduce algorithms. It implements a distributed architecture based on daemon processes that are responsible for all the aspects of query execution that run on the same machines.

Thus, it reduces the latency of utilizing MapReduce and this makes Impala faster than Apache Hive.

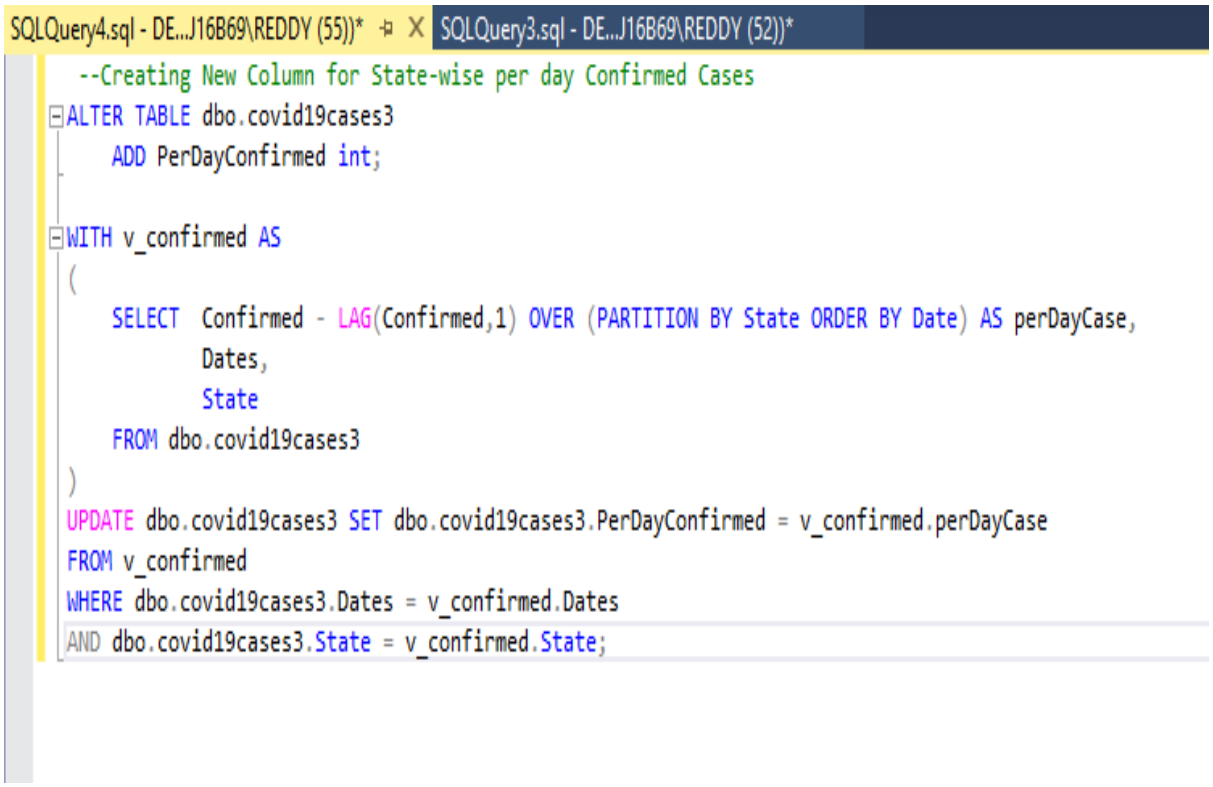
IMPLEMENTATION

I was taken this Data set from internet of covid19 in the course. And I was free to do anything with the data. I uploaded data set into Microsoft SQL server, and I run my queries and analyzed my data of covid 19.

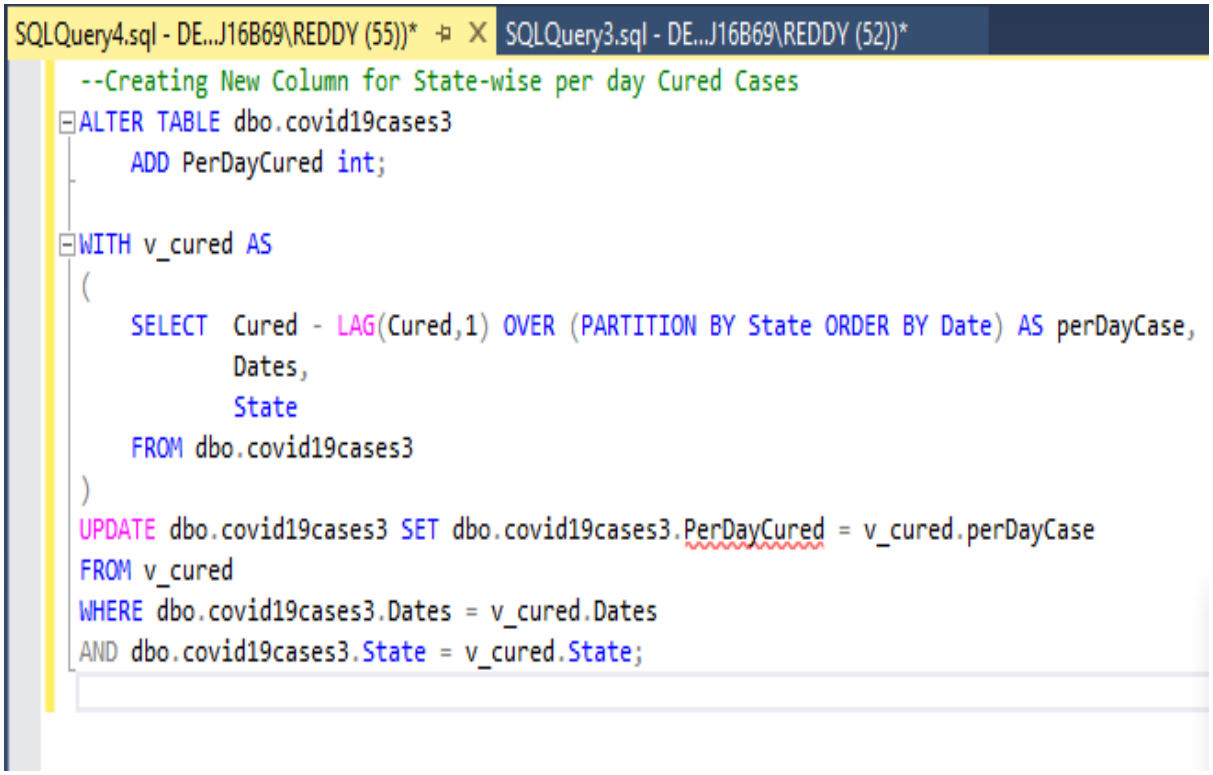
Creating new column for Active covid cases



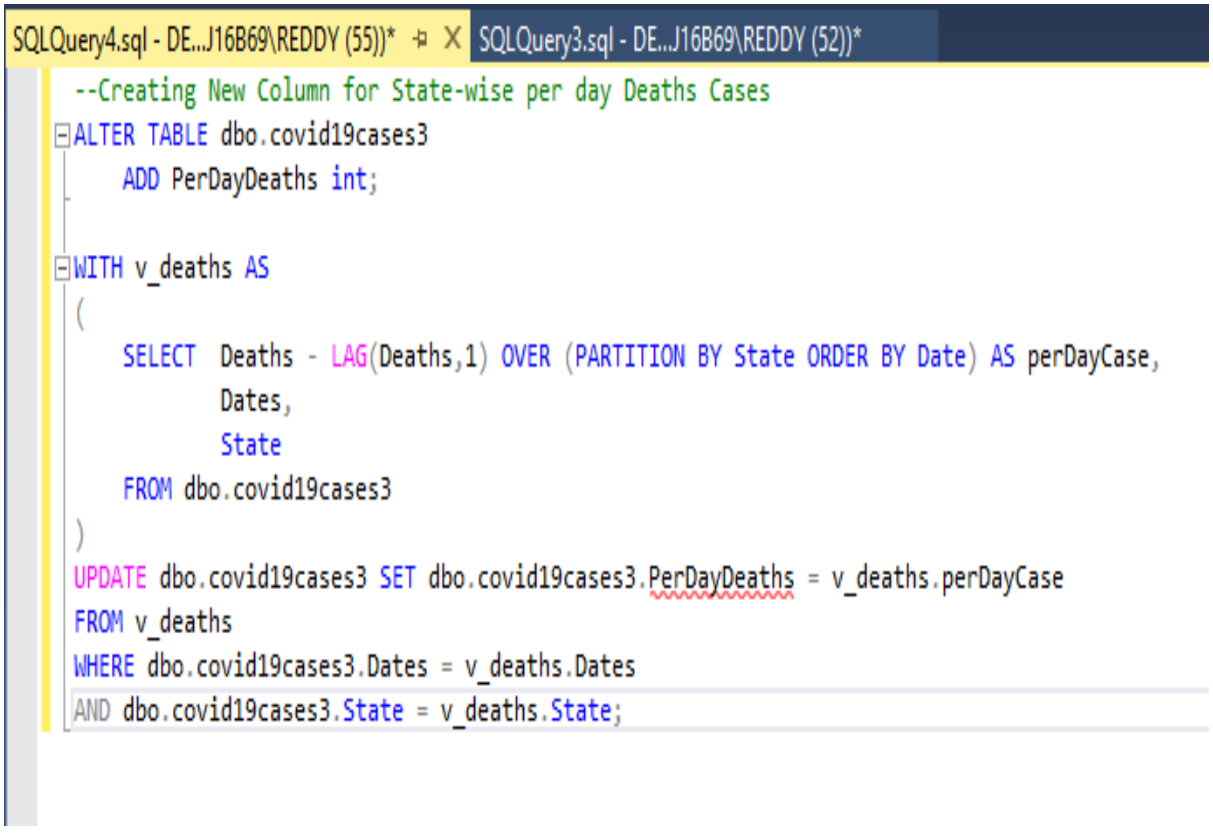
Creating new column for state wise per day confirmed cases



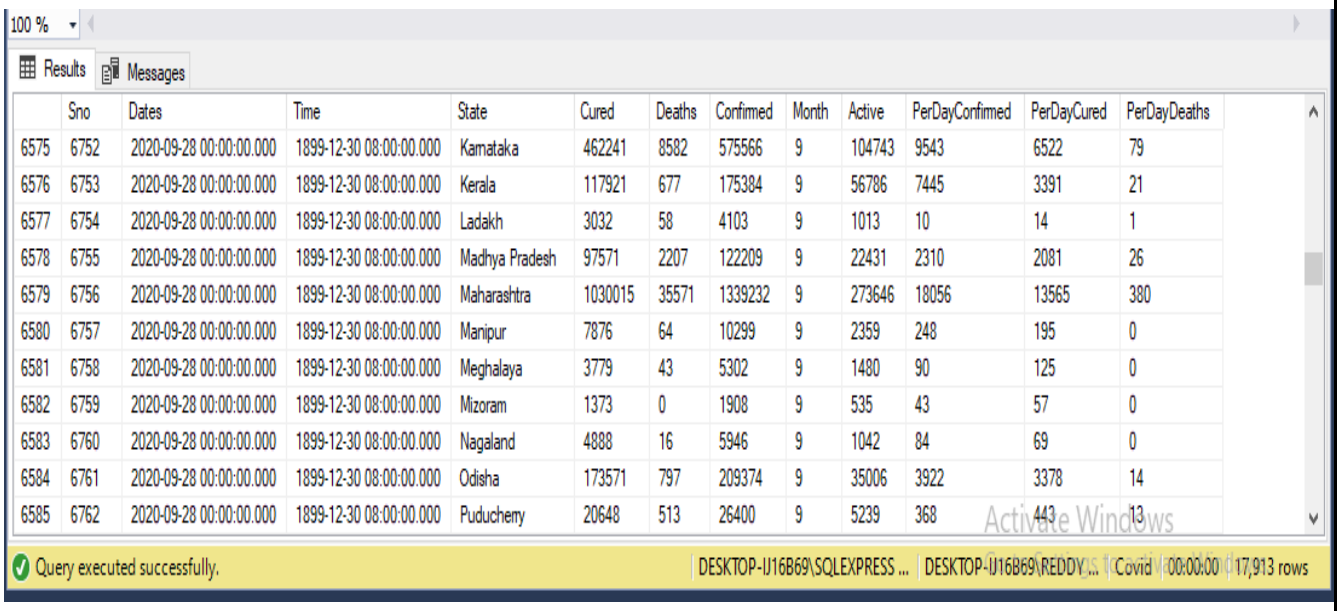
Creating new column for state wise per day cured cases



Creating new column for state wise per day deaths cases



After altering all the columns, the table is like this



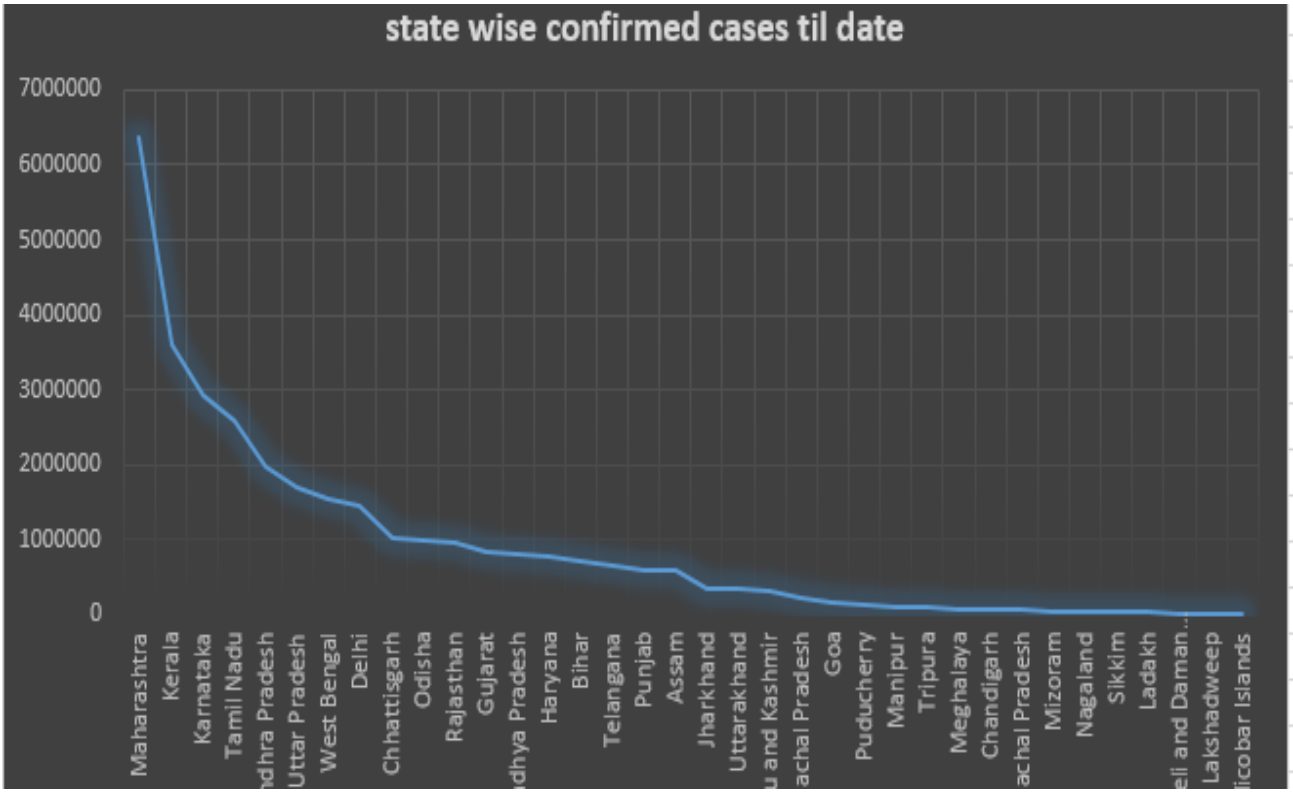
	Sno	Dates	Time	State	Cured	Deaths	Confirmed	Month	Active	PerDayConfirmed	PerDayCured	PerDayDeaths
6575	6752	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Kamataka	462241	8582	575566	9	104743	9543	6522	79
6576	6753	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Kerala	117921	677	175384	9	56786	7445	3391	21
6577	6754	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Ladakh	3032	58	4103	9	1013	10	14	1
6578	6755	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Madhya Pradesh	97571	2207	122209	9	22431	2310	2081	26
6579	6756	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Maharashtra	1030015	35571	1339232	9	273646	18056	13565	380
6580	6757	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Manipur	7876	64	10299	9	2359	248	195	0
6581	6758	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Meghalaya	3779	43	5302	9	1480	90	125	0
6582	6759	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Mizoram	1373	0	1908	9	535	43	57	0
6583	6760	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Nagaland	4888	16	5946	9	1042	84	69	0
6584	6761	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Odisha	173571	797	209374	9	35006	3922	3378	14
6585	6762	2020-09-28 00:00:00.000	1899-12-30 08:00:00.000	Puducherry	20648	513	26400	9	5239	368	443	13

Query executed successfully. DESKTOP-IJ16B69\SQLEXPRESS ... DESKTOP-IJ16B69\REDDY ... Covid 00:00:00 17,913 rows

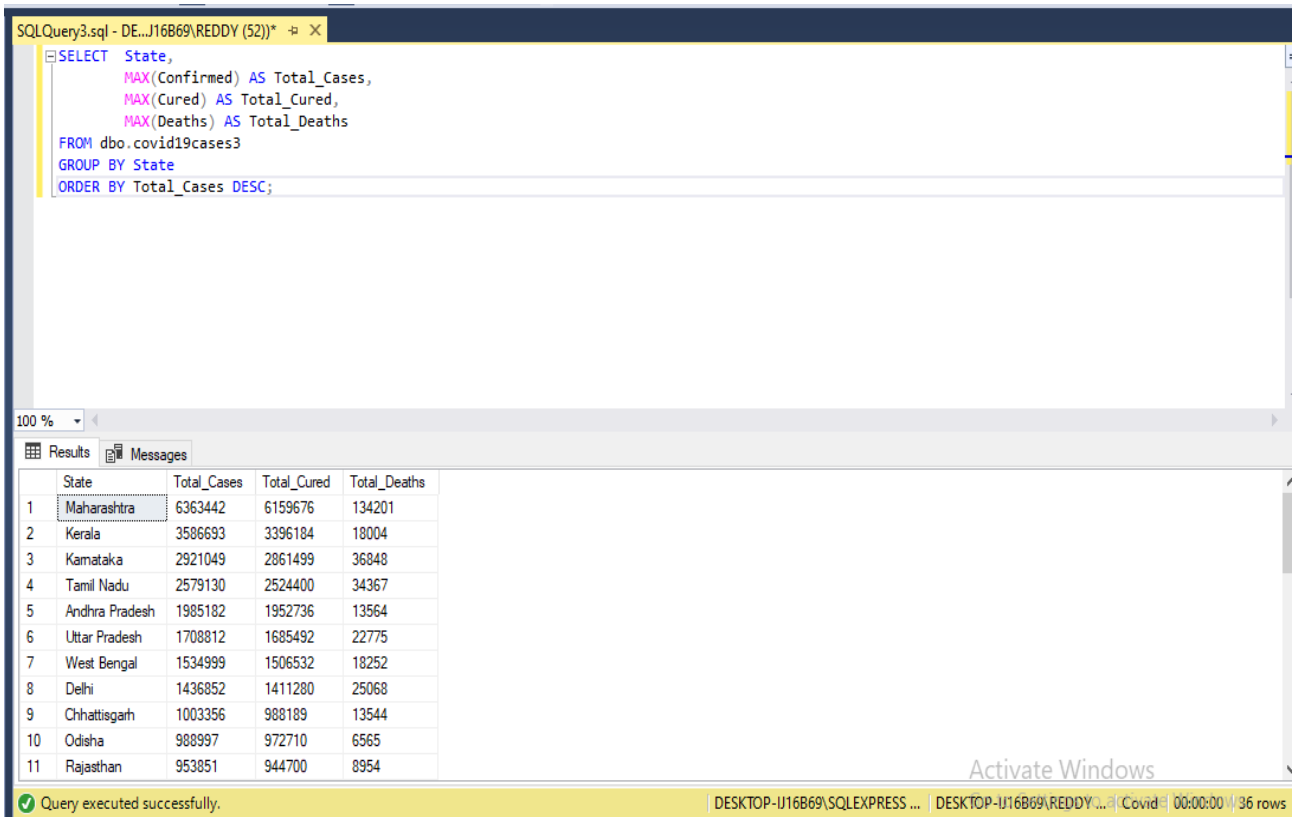
1. **Query for total state wise confirmed case till date.**

```
SQLQuery4.sql - DE...J16B69\REDDY (55))*   SQLQuery3.sql - DE...J16B69\REDDY (52))*
  --Total State-wise Confirmed cases till date
SELECT State,
       MAX(Confirmed) AS TotalCases
FROM dbo.covid19cases3
GROUP BY State
ORDER BY TotalCases DESC;
```

	State	TotalCases
1	Maharashtra	6363442
2	Kerala	3586693
3	Kamataka	2921049
4	Tamil Nadu	2579130
5	Andhra Pradesh	1985182
6	Uttar Pradesh	1708812
7	West Bengal	1534999
8	Delhi	1436852
9	Chhattisgarh	1003356
10	Odisha	988997
11	Rajasthan	953851



2.total confirmed cases, cured cases and death cases of all states



SQLQuery3.sql - DE...J16B69\REDDY (52))*

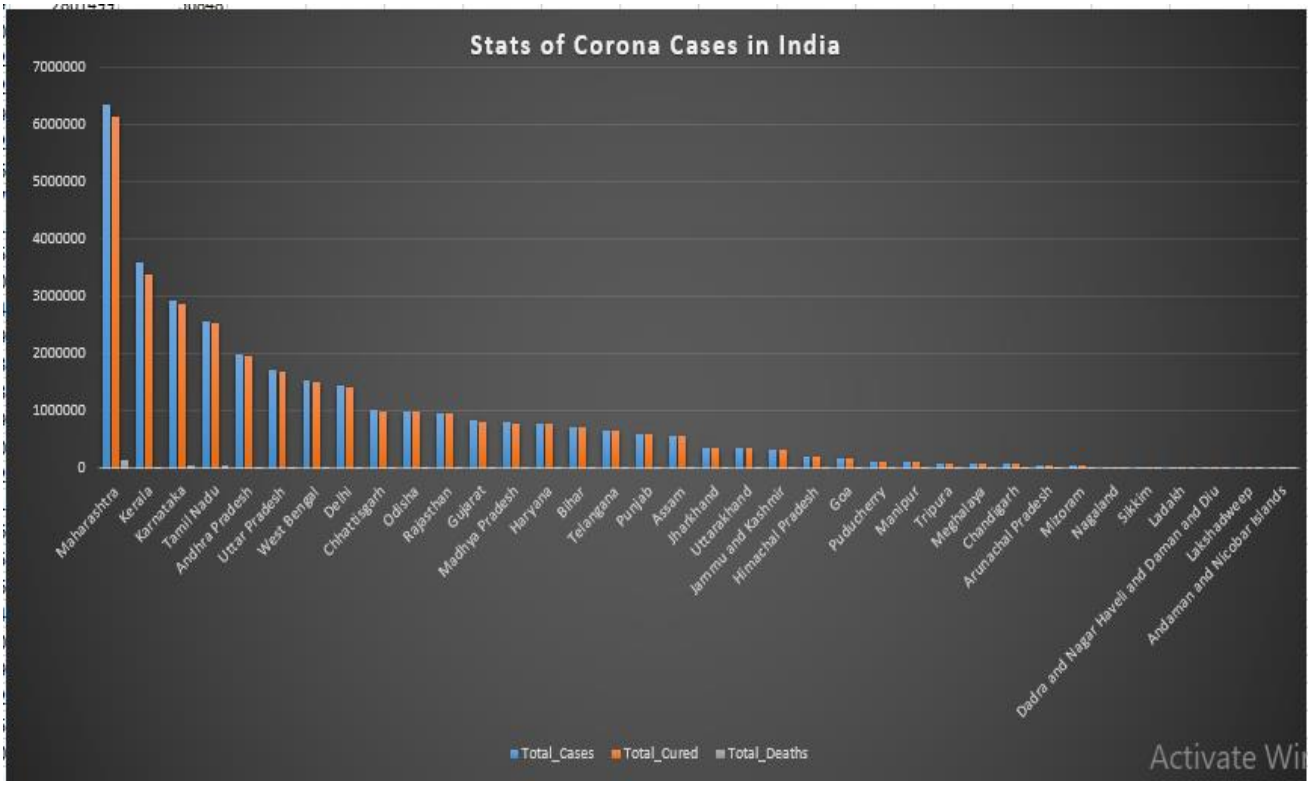
```
SELECT  State,
        MAX(Confirmed) AS Total_Cases,
        MAX(Cured) AS Total_Cured,
        MAX(Deaths) AS Total_Deaths
FROM dbo.covid19cases3
GROUP BY State
ORDER BY Total_Cases DESC;
```

100 %

Results | Messages

	State	Total_Cases	Total_Cured	Total_Deaths
1	Maharashtra	6363442	6159676	134201
2	Kerala	3586693	3396184	18004
3	Karnataka	2921049	2861499	36848
4	Tamil Nadu	2579130	2524400	34367
5	Andhra Pradesh	1985182	1952736	13564
6	Uttar Pradesh	1708812	1685492	22775
7	West Bengal	1534999	1506532	18252
8	Delhi	1436852	1411280	25068
9	Chhattisgarh	1003356	988189	13544
10	Odisha	988997	972710	6565
11	Rajasthan	953851	944700	8954

Query executed successfully. DESKTOP-IJ16B69\SQLEXPRESS ... DESKTOP-IJ16B69\REDDY ... Covid 00:00:00 36 rows



3.query for maximum per day confirmed case per state.

SQLQuery5.sql - DE...J16B69\REDDY (54))* | SQLQuery3.sql - DE...J16B69\REDDY (52))*

```
--Maximum Per-Day Confirmed case per State
SELECT  State,
        MaxPerDayConfirmed,
        Dates,
        Month
FROM
(SELECT  Dates,
         State,
         Month,
         MAX(PerDayConfirmed) OVER (PARTITION BY State) AS MaxPerDayConfirmed,
         DENSE_RANK() OVER (PARTITION BY State ORDER BY PerDayConfirmed desc) AS HighestConfirmed
FROM dbo.covid19cases3) cte
WHERE HighestConfirmed = 1
ORDER BY MaxPerDayConfirmed DESC;
```

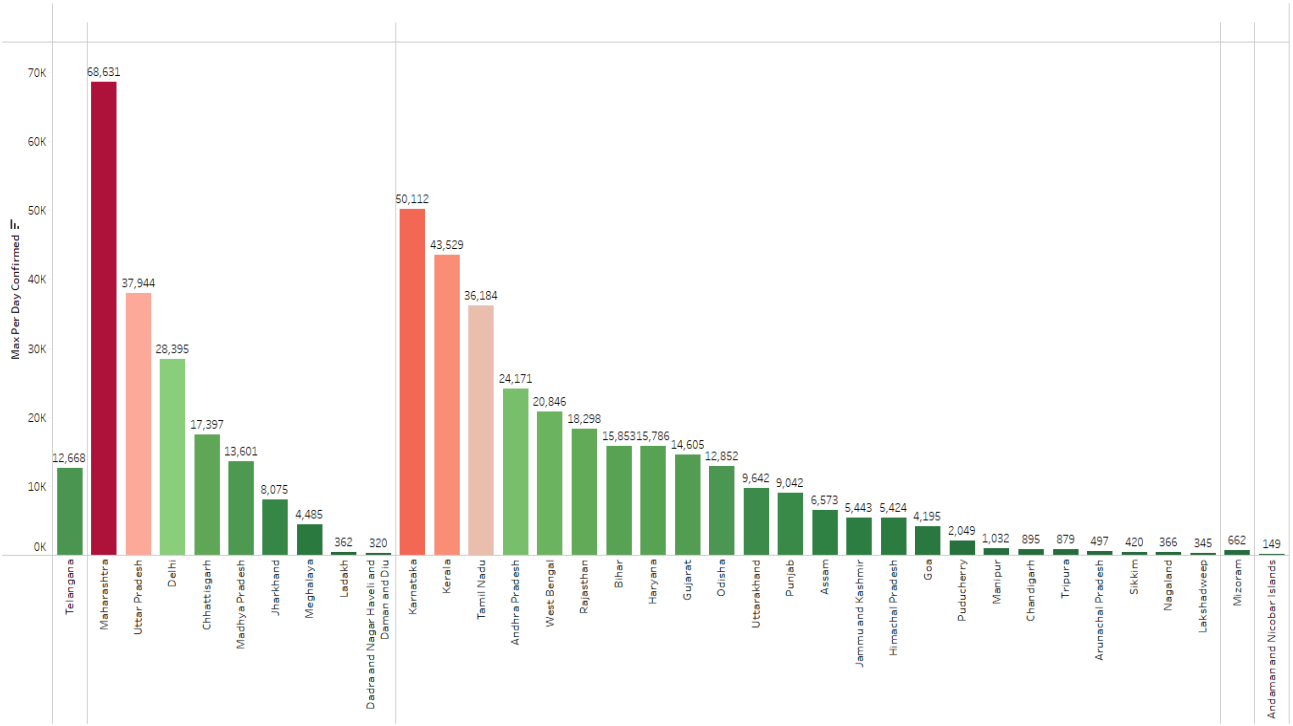
100 %

Results | Messages

	State	MaxPerDayConfirmed	Dates	Month
1	Maharashtra	68631	2021-04-19 00:00:00.000	4
2	Karnataka	50112	2021-05-06 00:00:00.000	5
3	Kerala	43529	2021-05-13 00:00:00.000	5
4	Uttar Pradesh	37944	2021-04-25 00:00:00.000	4
5	Tamil Nadu	36184	2021-05-22 00:00:00.000	5
6	Delhi	28395	2021-04-21 00:00:00.000	4
7	Andhra Pradesh	24171	2021-05-17 00:00:00.000	5
8	West Bengal	20846	2021-05-15 00:00:00.000	5
9	Rajasthan	18298	2021-05-03 00:00:00.000	5
10	Chhattisgarh	17397	2021-04-24 00:00:00.000	4
11	Bihar	15853	2021-05-01 00:00:00.000	4

Query executed successfully. | DESKTOP-IJ16B69\SQLEXPRES

Maximum Confirmed Cases/Day State-wise and Month it occurred



4. query for maximum per day deaths case per state.

SQLQuery5.sql - DE...J16B69\REDDY (54))* | SQLQuery3.sql - DE...J16B69\REDDY (52))*

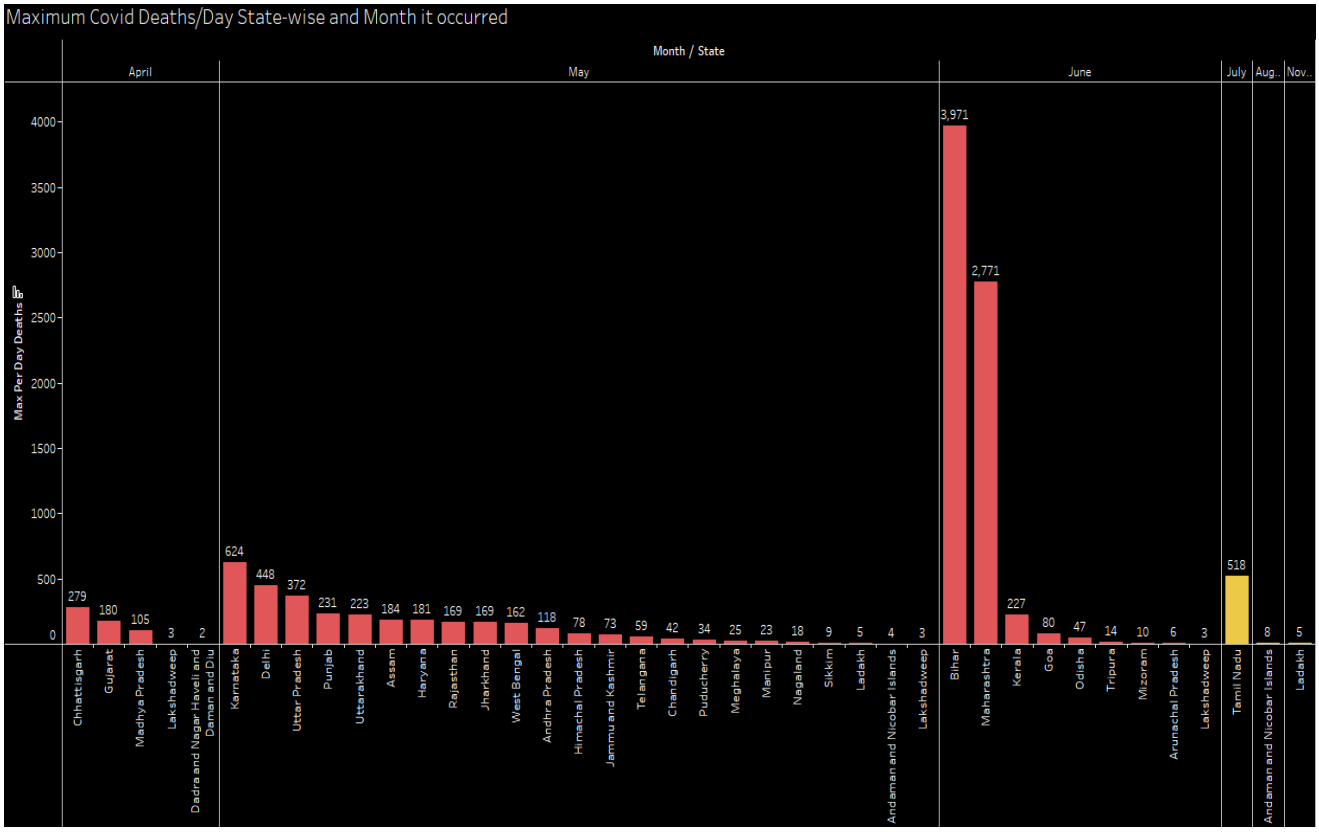
```sql
--Maximum Per-Day Deaths case per State
SELECT  State,
        MaxPerDayDeaths,
        Dates,
        Month
FROM
(SELECT  Dates,
         State,
         Month,
         MAX(PerDayDeaths) OVER (PARTITION BY State) AS MaxPerDayDeaths,
         DENSE_RANK() OVER (PARTITION BY State ORDER BY PerDayDeaths desc) AS HighestDeaths
FROM dbo.covid19cases3) cte
WHERE HighestDeaths = 1
ORDER BY MaxPerDayDeaths DESC;
```

100 %

Results | Messages

	State	MaxPerDayDeaths	Dates	Month
1	Bihar	3971	2021-06-10 00:00:00.000	6
2	Maharashtra	3821	2021-07-22 00:00:00.000	7
3	Madhya Pradesh	1483	2021-07-14 00:00:00.000	7
4	Karnataka	624	2021-05-24 00:00:00.000	5
5	Tamil Nadu	518	2020-07-23 00:00:00.000	7
6	Delhi	448	2021-05-04 00:00:00.000	5
7	Uttar Pradesh	372	2021-05-08 00:00:00.000	5
8	Chhattisgarh	279	2021-04-29 00:00:00.000	4
9	Punjab	231	2021-05-19 00:00:00.000	5
10	Kerala	227	2021-06-07 00:00:00.000	6

Query executed successfully. DESKTOP-IJ16B69\SQLEX



5. query for last 5-day active cases from each date for state ‘Delhi’

SQLQuery5.sql - DE...J16B69\REDDY (54))* | SQLQuery3.sql - DE...J16B69\REDDY (52))

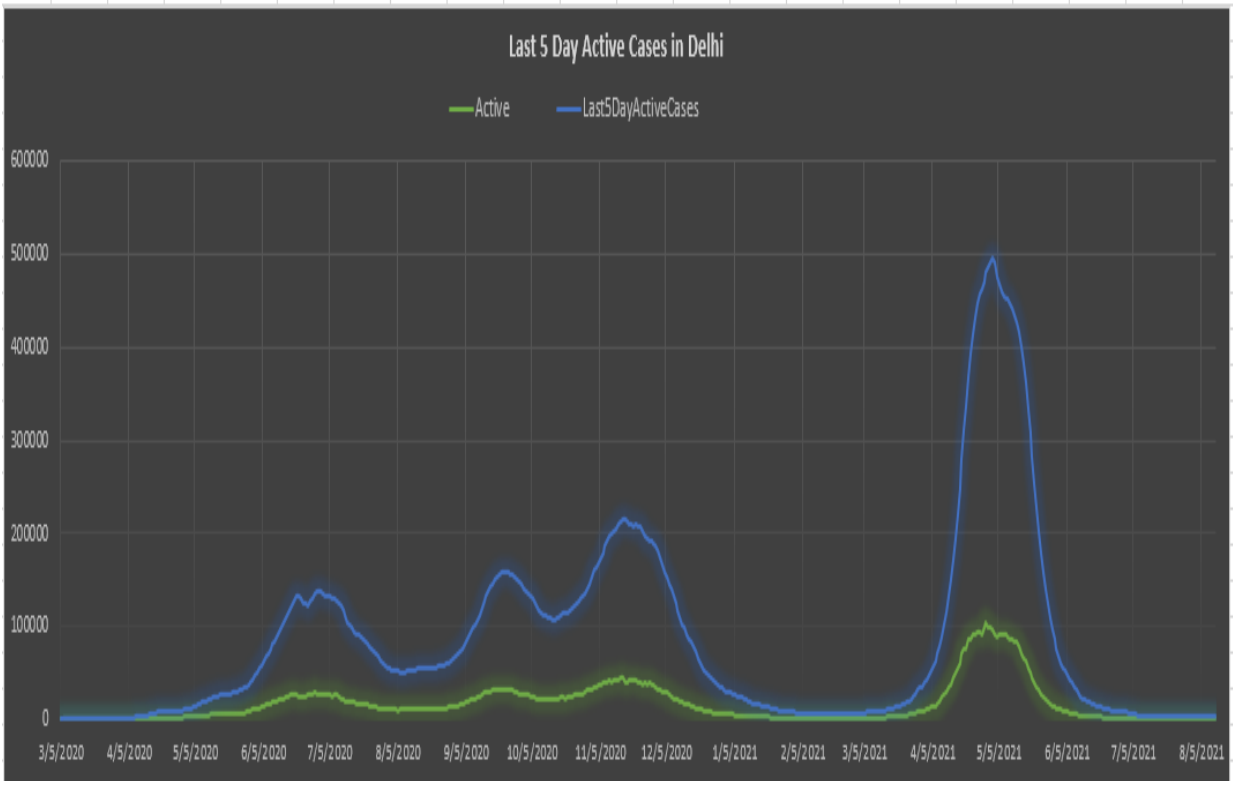
```
--Last 5 day Active cases from each date for state 'DELHI'
SELECT  Dates,
        Active,
        sum(Active) over (order by Dates Rows 4 preceding) as Last5dayActiveCases
FROM dbo.covid19cases3
where State='Delhi';
```

100 %

Results | Messages

	Dates	Active	Last5dayActiveCases
1	2020-03-05 00:00:00.000	2	2
2	2020-03-06 00:00:00.000	3	5
3	2020-03-07 00:00:00.000	3	8
4	2020-03-08 00:00:00.000	3	11
5	2020-03-09 00:00:00.000	4	15
6	2020-03-10 00:00:00.000	4	17
7	2020-03-11 00:00:00.000	5	19
8	2020-03-12 00:00:00.000	6	22
9	2020-03-13 00:00:00.000	6	25
10	2020-03-14 00:00:00.000	5	26
11	2020-03-15 00:00:00.000	4	26

Query executed successfully. DESKTOP-IJ16B69\SQLEXI



6.mortality rate of each state in India

SQLQuery5.sql - DE...J16B69\REDDY (54))* | SQLQuery3.sql - DE...J16B69\REDDY (52))

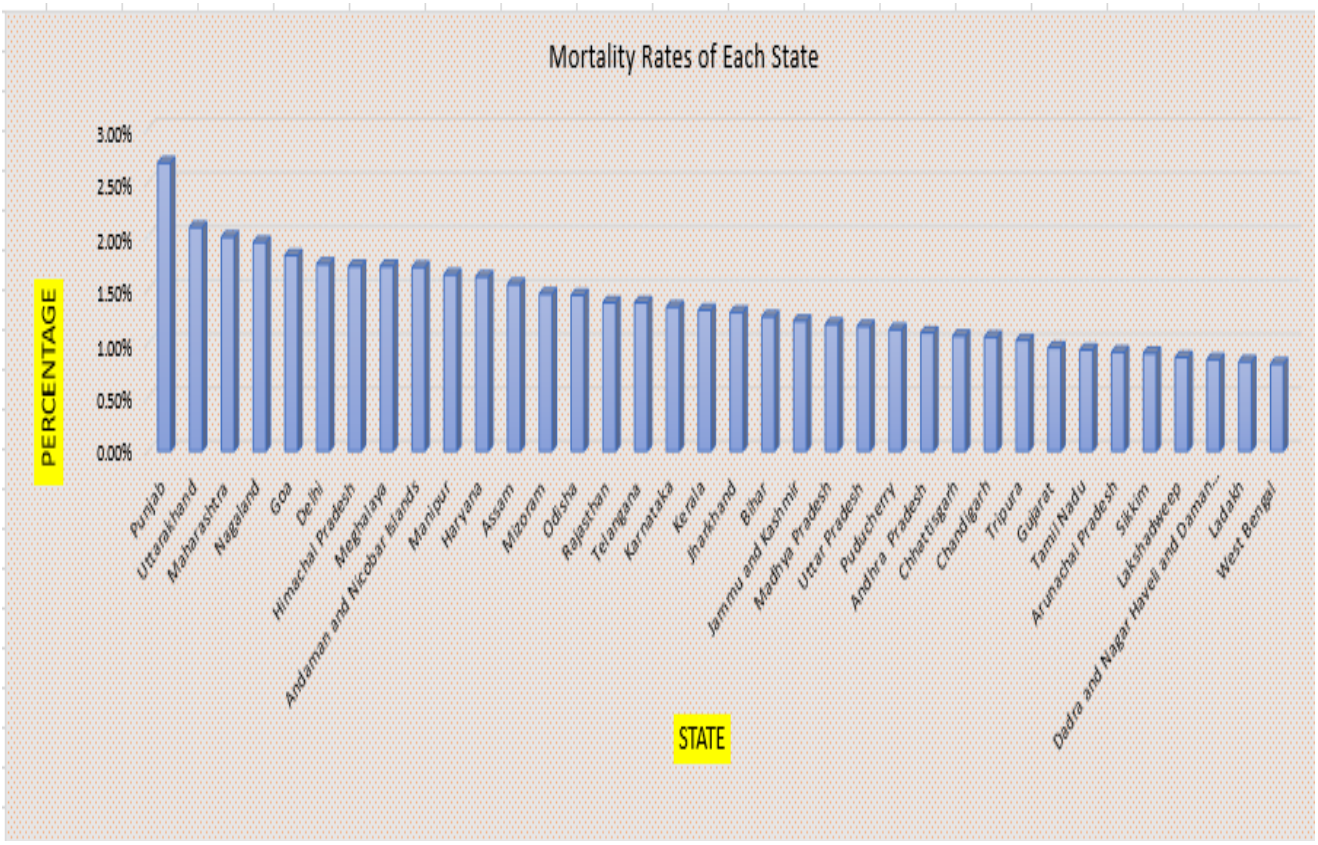
```
--Calculating the State-wise Mortality Rate
WITH cte AS
(SELECT  State,
        Max(Deaths) as Deaths,
        Max(Confirmed) as CovidCases
FROM dbo.covid19cases3
GROUP BY STATE)
SELECT  State,
        CAST(Round((Deaths/CovidCases)*100, 2) as float ) AS DeathRate
FROM cte
ORDER BY DeathRate DESC;
```

100 %

Results | Messages

	State	DeathRate
1	Punjab	2.72
2	Uttarakhand	2.15
3	Maharashtra	2.11
4	Nagaland	2.03
5	Goa	1.84
6	Delhi	1.74
7	Andaman and Nicobar Islands	1.71
8	Meghalaya	1.7
9	Himachal Pradesh	1.7
10	Manipur	1.58
11	Jharkhand	1.48

Query executed successfully.



7. finding time when covid waves attacked Delhi

SQLQuery5.sql - DE...J16B69\REDDY (54))* | SQLQuery3.sql - DE...J16B69\REDDY (52))

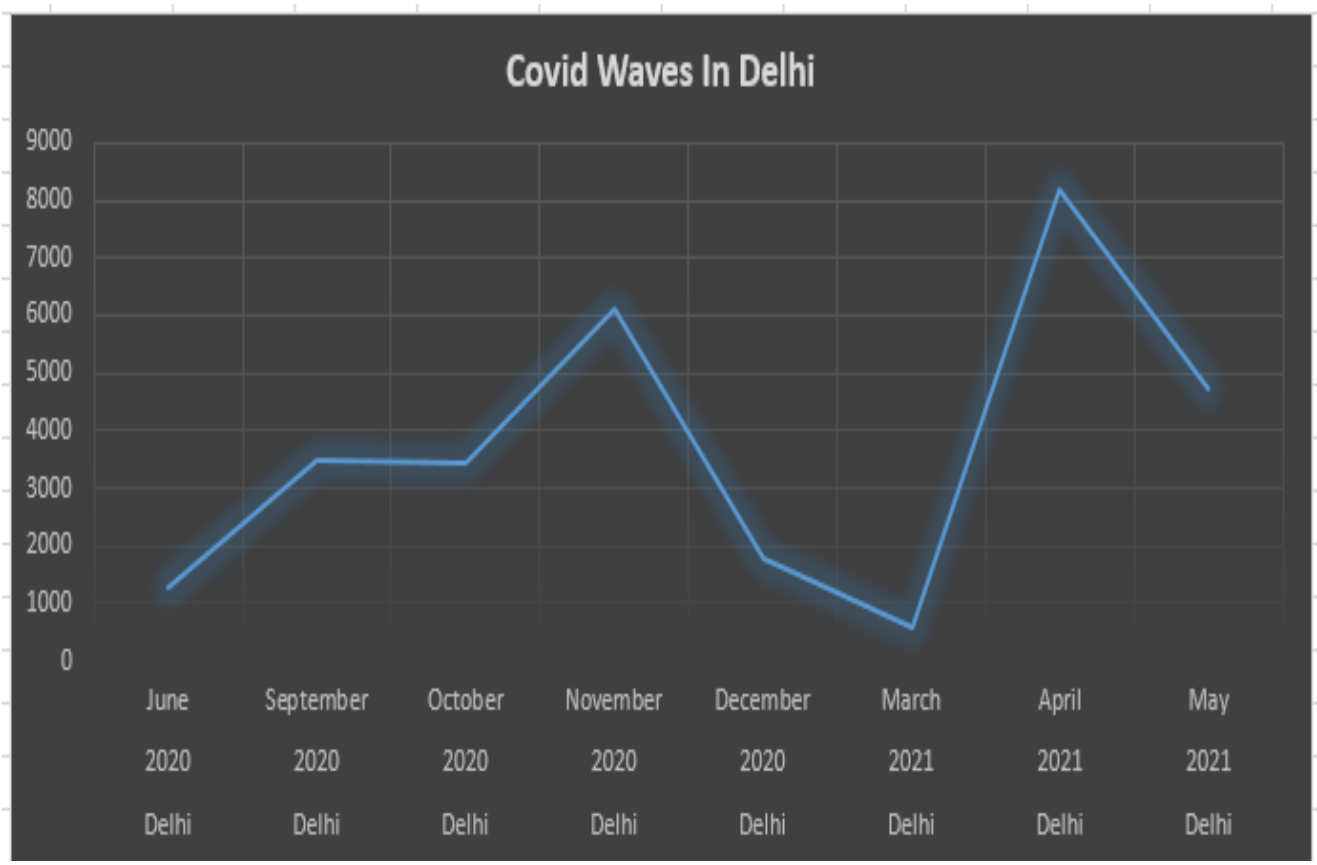
```
--Finding the 'Time' when Covid Waves attacked the capital city 'Delhi'
WITH monthlySpike AS
(SELECT  State,
         Dates,
         Years,
         Month,
         PerDayConfirmed,
         AVG(PerDayConfirmed) OVER (PARTITION BY Month) as Spike,
         DENSE_RANK() OVER (PARTITION BY Month ORDER BY PerDayConfirmed DESC) AS rnk
FROM dbo.covid19cases3
WHERE State = 'Delhi')
SELECT  State,
        Years,
        Month,
        ROUND(Spike,2) as AvgSpikeInCases
FROM monthlySpike
WHERE rnk = 1 AND PerDayConfirmed >3000
ORDER BY Dates;
```

100 %

Results | Messages

	State	Years	Month	AvgSpikeInCases
1	Delhi	2020	6	1257
2	Delhi	2020	9	3498
3	Delhi	2020	10	3451
4	Delhi	2020	11	6122
5	Delhi	2020	12	1774
6	Delhi	2021	3	585
7	Delhi	2021	4	8171
8	Delhi	2021	5	4729

Query executed successfully. DESKTOP-IJ16B6


Covid Waves In Delhi
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
June 2020 Delhi
September 2020 Delhi
October 2020 Delhi
November 2020 Delhi
December 2020 Delhi
March 2021 Delhi
April 2021 Delhi
May 2021 Delhi

Learning Outcomes

Throughout this course, I have learnt many skills and concepts. The most important thing I have learnt from this summer training is Exploratory Analysis. I was given a dataset of covid 19 from internet in the course. And I was free to do anything with the data.

- I have learned to frame questions that would be helpful to us in one way or other.
- I have learned how to extract the data I wanted from the dataset to answer the questions I framed.
- I have learned how to export the data and cross platform into excel to make appealing graphs.
  Using graphs helps to showcase the information very easy and clear.
- I have improved my familiarity with the query language 'SQL'
- I have learned the basic concepts of big data about what it is, how it is managed, how many types it has, what is the use of big data and many more.

- Talking about the analysis we did on this project. We can notice many points by relating multiple answers. For example,

  I learned how big data managed to see the future market trends.
- I learned where big data is using most and how much data they are handling using big data.
- How organisations develop their products based on consumer value.
- How big data is used in organisations for risk free management.
- I also learned how these increasing/ decreasing in graph trends works. Ex. covid-19 cases.

SWOT analysis of Big Data

Strengths of Big Data

The fundamental strength of big data can be easily associated with the three Vs it represents: volume, velocity, variety. Regarding volume and velocity, over years.

• Volume.

• Variety.

• Velocity.

• Value to organizations.

• Better decision making.

• More efficiency.

• Continuous growth.

Weakness of Big Data

To accomplish the benefits that big data can deliver it's necessary to define effective policies and procedures for handling and maintaining big data. To deal with big data is required new infrastructures able to address volume, variety, and velocity regarding data. It's also necessary scalability, capacity of operating and support immediate response**.**

• Lack of adequate infrastructure

• Data acquisition

• Data aggregation and storage

• Data access and sharing

• Interoperability

• Risk of poor-quality insights

• Lack of trained professionals

Opportunities of Big Data

Without any doubt big data has the potential to cause impact in technology, economic and society, boosting innovation and improving business models. According to (BDV, 2016), big data will open new opportunities and enable breakthroughs in very different perspectives:

1. Descriptive to answer what happened

2. Diagnostic to answer the reason why happened

3. Predictive to understand what will happen

4. Prescriptive to detect how we can make it happen.

• Improve lifestyle support

• Improve public health

• Reduce costs

• New business models

• Fraud detection

• Patient-adjusted treatments

Threats of Big Data

• Data privacy

• Data security

• Lack of incentive

• Ethical/moral questions

• Misuse of information.

CONCLUSION

Big data provides an opportunity for "big analysis" leading to "big opportunities" to advance the quality of life, or to solve the mysteries of the world. We are in the development area of big data. In this paper details about Big kas at Data have been discussed taking the Oracle Framework as a base. We have characteristics of Big Data and provided deep information on the various components of big data from a Apache Hive perspective. There are various challenges and issues of big data. There must support and encourage fundamental research towards these technical issues if we want to achieve the benefits of big data. In today if we see the information overloads almost everywhere, by centralizing data acquisition and consolidation in the cloud, Bigdata methods offer new insight into existing data sets. Apache Hive is a fast-growing data framework, Apache Hive offers a free, cohesive platform that encapsulates data integration, data processing. monitoring and workflow scheduling etc. Future work would involve a detailed study on challenges and issues with big data various industries.

References

https://www.coursera.org/learn/cloudera-big-data-analysis-sql-queries

https://www.coursera.org/learn/foundations-big-data-analysis-sql

https://www.coursera.org

https://www.youtube.com/watch?v=HXV3zeQKqGY

Thank you