# Project Title - Covid_19 Data Analysis

we download the dataset of covid_19 cases in India from Kaggle . I have done EDA on covid_19 dataset.For cleaning, sorting, merging we use Numpy and pandas Module of Python. For Visualizing i have use Matplotlib and seborn and try analysis the covid 19 cases in Inddia.

## Downloading the Dataset

from kaggle i have downloaded on my hardisk then upload the dataset of my repository.

```
project_name = "covid_19 Data Analysis"
```

```
!pip install jovian --upgrade -q
```

```
import jovian
```

```
jovian.commit(project=project_name)
```

```
[jovian] Attempting to save notebook..
[jovian] Creating a new project "mddanishqamar179/covid_19 Data Analysis"
[jovian] Uploading notebook..
[jovian] Capturing environment..
[jovian] Committed successfully! https://jovian.ml/mddanishqamar179/covid-19-data-analysis
```

```
'https://jovian.ml/mddanishqamar179/covid-19-data-analysis'
```

## Data Preparation and Cleaning

** I am doing data analyis on covid_19 in India.I have downloaded dataset on kaglle it contains 9 columns and more than 7000 row which has data upto 10 october 2020.

```
import numpy as np
import pandas as pd
```

```
covid_df=pd.read_csv("covid_19_india.csv")
```

```
covid_df
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths |
|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 30/01/20 | 6:00 PM | Kerala | 1 | 0 | 0 | 0 |
| **1** | 2 | 31/01/20 | 6:00 PM | Kerala | 1 | 0 | 0 | 0 |

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths |
|---|---|---|---|---|---|---|---|---|
| **2** | 3 | 01/02/20 | 6:00 PM | Kerala | 2 | 0 | 0 | 0 |
| **3** | 4 | 02/02/20 | 6:00 PM | Kerala | 3 | 0 | 0 | 0 |
| **4** | 5 | 03/02/20 | 6:00 PM | Kerala | 3 | 0 | 0 | 0 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **7011** | 7012 | 05/10/20 | 8:00 AM | Telengana | - | - | 172388 | 1171 |
| **7012** | 7013 | 05/10/20 | 8:00 AM | Tripura | - | - | 21876 | 299 |
| **7013** | 7014 | 05/10/20 | 8:00 AM | Uttarakhand | - | - | 41740 | 652 |
| **7014** | 7015 | 05/10/20 | 8:00 AM | Uttar Pradesh | - | - | 362052 | 6029 |
| **7015** | 7016 | 05/10/20 | 8:00 AM | West Bengal | - | - | 237698 | 5194 |

7016 rows × 9 columns

```
type(covid_df)
```

```
pandas.core.frame.DataFrame
```

```
covid_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7016 entries, 0 to 7015
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype 
---  ------                    --------------  ----- 
 0   Sno                       7016 non-null   int64 
 1   Date                      7016 non-null   object
 2   Time                      7016 non-null   object
 3   State/UnionTerritory      7016 non-null   object
 4   ConfirmedIndianNational   7016 non-null   object
 5   ConfirmedForeignNational  7016 non-null   object
 6   Cured                     7016 non-null   int64 
 7   Deaths                    7016 non-null   int64 
 8   Confirmed                 7016 non-null   int64 
dtypes: int64(4), object(5)
memory usage: 493.4+ KB
```

```
covid_df.columns
```

```
Index(['Sno', 'Date', 'Time', 'State/UnionTerritory',
       'ConfirmedIndianNational', 'ConfirmedForeignNational', 'Cured',
```

```
       'Deaths', 'Confirmed'],
      dtype='object')
```

```
covid_df.head(5)
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths | Conf |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 30/01/20 | 6:00 PM | Kerala | 1 | 0 | 0 | 0 | |
| **1** | 2 | 31/01/20 | 6:00 PM | Kerala | 1 | 0 | 0 | 0 | |
| **2** | 3 | 01/02/20 | 6:00 PM | Kerala | 2 | 0 | 0 | 0 | |
| **3** | 4 | 02/02/20 | 6:00 PM | Kerala | 3 | 0 | 0 | 0 | |
| **4** | 5 | 03/02/20 | 6:00 PM | Kerala | 3 | 0 | 0 | 0 | |

```
covid_df.shape
```

```
(7016, 9)
```

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Attempting to save notebook..
[jovian] Updating notebook "mddanishqamar179/covid-19-data-analysis" on
https://jovian.ml/
[jovian] Uploading notebook..
[jovian] Capturing environment..
[jovian] Committed successfully! https://jovian.ml/mddanishqamar179/covid-19-data-
analysis
```

```
'https://jovian.ml/mddanishqamar179/covid-19-data-analysis'
```

# Exploratory Analysis and Visualization

i am going to Done EDA and Visualizaion

```
covid_df.isnull()
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths | Co |
|---|---|---|---|---|---|---|---|---|---|
| **0** | False | False | False | False | False | False | False | False | |
| **1** | False | False | False | False | False | False | False | False | |
| **2** | False | False | False | False | False | False | False | False | |
| **3** | False | False | False | False | False | False | False | False | |
| **4** | False | False | False | False | False | False | False | False | |

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths | Co |
|---|---|---|---|---|---|---|---|---|---|
| ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **7011** | False | False | False | False | False | False | False | False | |
| **7012** | False | False | False | False | False | False | False | False | |
| **7013** | False | False | False | False | False | False | False | False | |
| **7014** | False | False | False | False | False | False | False | False | |
| **7015** | False | False | False | False | False | False | False | False | |

7016 rows × 9 columns

```
covid_df.min()
```

```
Sno                                          1
Date                                    01/02/20
Time                                    10:00 AM
State/UnionTerritory        Andaman and Nicobar Islands
ConfirmedIndianNational                      -
ConfirmedForeignNational                     -
Cured                                        0
Deaths                                       0
Confirmed                                    0
dtype: object
```

```
covid_df.describe()
```

| | Sno | Cured | Deaths | Confirmed |
|---|---|---|---|---|
| **count** | 7016.000000 | 7.016000e+03 | 7016.000000 | 7.016000e+03 |
| **mean** | 3508.500000 | 3.295071e+04 | 814.867303 | 4.388143e+04 |
| **std** | 2025.489077 | 9.584158e+04 | 2930.476907 | 1.228508e+05 |
| **min** | 1.000000 | 0.000000e+00 | 0.000000 | 0.000000e+00 |
| **25%** | 1754.750000 | 3.700000e+01 | 1.000000 | 1.600000e+02 |
| **50%** | 3508.500000 | 1.428000e+03 | 21.000000 | 2.680500e+03 |
| **75%** | 5262.250000 | 1.585000e+04 | 378.250000 | 2.465375e+04 |
| **max** | 7016.000000 | 1.149603e+06 | 38084.000000 | 1.443409e+06 |

```
covid_df.sort_values("Confirmed", ascending=False)
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Death |
|---|---|---|---|---|---|---|---|---|
| **7000** | 7001 | 05/10/20 | 8:00 AM | Maharashtra | - | - | 1149603 | 3808 |
| **6965** | 6966 | 04/10/20 | 8:00 AM | Maharashtra | - | - | 1134555 | 3775 |
| **6930** | 6931 | 03/10/20 | 8:00 AM | Maharashtra | - | - | 1117720 | 3748 |
| **6895** | 6896 | 02/10/20 | 8:00 AM | Maharashtra | - | - | 1104426 | 3705 |

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Death |
|---|---|---|---|---|---|---|---|---|
| **6860** | 6861 | 01/10/20 | 8:00 AM | Maharashtra | - | - | 1088322 | 3666 |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **981** | 982 | 15/04/20 | 5:00 PM | Nagaland | - | - | 0 | |
| **1146** | 1147 | 20/04/20 | 5:00 PM | Nagaland | - | - | 0 | |
| **1047** | 1048 | 17/04/20 | 5:00 PM | Nagaland | - | - | 0 | |
| **1113** | 1114 | 19/04/20 | 5:00 PM | Nagaland | - | - | 0 | |
| **1179** | 1180 | 21/04/20 | 5:00 PM | Nagaland | - | - | 0 | |

7016 rows × 9 columns

```
covid_df.sort_values("Cured", ascending= False)
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Death |
|---|---|---|---|---|---|---|---|---|
| **7000** | 7001 | 05/10/20 | 8:00 AM | Maharashtra | - | - | 1149603 | 3808 |
| **6965** | 6966 | 04/10/20 | 8:00 AM | Maharashtra | - | - | 1134555 | 3775 |
| **6930** | 6931 | 03/10/20 | 8:00 AM | Maharashtra | - | - | 1117720 | 3748 |
| **6895** | 6896 | 02/10/20 | 8:00 AM | Maharashtra | - | - | 1104426 | 3705 |
| **6860** | 6861 | 01/10/20 | 8:00 AM | Maharashtra | - | - | 1088322 | 3666 |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **3566** | 3567 | 29/06/20 | 8:00 AM | Cases being reassigned to states | - | - | 0 | |
| **2550** | 2551 | 01/06/20 | 8:00 AM | Sikkim | - | - | 0 | |
| **420** | 421 | 28/03/20 | 6:00 PM | Andaman and Nicobar Islands | 6 | 0 | 0 | |
| **2298** | 2299 | 25/05/20 | 8:00 AM | Sikkim | - | - | 0 | |
| **0** | 1 | 30/01/20 | 6:00 PM | Kerala | 1 | 0 | 0 | |

7016 rows × 9 columns

```
covid_df.sort_values("Deaths", ascending=False)
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Death |
|---|---|---|---|---|---|---|---|---|
| **7000** | 7001 | 05/10/20 | 8:00 AM | Maharashtra | - | - | 1149603 | 3808 |

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Death |
|---|---|---|---|---|---|---|---|---|
| **6965** | 6966 | 04/10/20 | 8:00 AM | Maharashtra | - | - | 1134555 | 3775 |
| **6930** | 6931 | 03/10/20 | 8:00 AM | Maharashtra | - | - | 1117720 | 3748 |
| **6895** | 6896 | 02/10/20 | 8:00 AM | Maharashtra | - | - | 1104426 | 3705 |
| **6860** | 6861 | 01/10/20 | 8:00 AM | Maharashtra | - | - | 1088322 | 3666 |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **2991** | 2992 | 14/06/20 | 8:00 AM | Andaman and Nicobar Islands | - | - | 33 | |
| **2993** | 2994 | 14/06/20 | 8:00 AM | Arunachal Pradesh | - | - | 4 | |
| **1346** | 1347 | 26/04/20 | 5:00 PM | Tripura | - | - | 2 | |
| **1347** | 1348 | 26/04/20 | 5:00 PM | Uttarakhand | - | - | 26 | |
| **0** | 1 | 30/01/20 | 6:00 PM | Kerala | 1 | 0 | 0 | |

7016 rows × 9 columns

```
covid_df.sort_values(["Confirmed","Cured","Deaths"], ascending=False)
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Death |
|---|---|---|---|---|---|---|---|---|
| **7000** | 7001 | 05/10/20 | 8:00 AM | Maharashtra | - | - | 1149603 | 3808 |
| **6965** | 6966 | 04/10/20 | 8:00 AM | Maharashtra | - | - | 1134555 | 3775 |
| **6930** | 6931 | 03/10/20 | 8:00 AM | Maharashtra | - | - | 1117720 | 3748 |
| **6895** | 6896 | 02/10/20 | 8:00 AM | Maharashtra | - | - | 1104426 | 3705 |
| **6860** | 6861 | 01/10/20 | 8:00 AM | Maharashtra | - | - | 1088322 | 3666 |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **1047** | 1048 | 17/04/20 | 5:00 PM | Nagaland | - | - | 0 | |
| **1080** | 1081 | 18/04/20 | 5:00 PM | Nagaland | - | - | 0 | |
| **1113** | 1114 | 19/04/20 | 5:00 PM | Nagaland | - | - | 0 | |
| **1146** | 1147 | 20/04/20 | 5:00 PM | Nagaland | - | - | 0 | |
| **1179** | 1180 | 21/04/20 | 5:00 PM | Nagaland | - | - | 0 | |

7016 rows × 9 columns

```
covid_df[covid_df["Confirmed"]>100000].head(10)
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths |
|---|---|---|---|---|---|---|---|---|
| **2974** | 2975 | 13/06/20 | 8:00 AM | Maharashtra | - | - | 47796 | 3717 |
| **3010** | 3011 | 14/06/20 | 8:00 AM | Maharashtra | - | - | 49346 | 3830 |
| **3046** | 3047 | 15/06/20 | 8:00 AM | Maharashtra | - | - | 50978 | 3950 |
| **3082** | 3083 | 16/06/20 | 8:00 AM | Maharashtra | - | - | 56049 | 4128 |
| **3118** | 3119 | 17/06/20 | 8:00 AM | Maharashtra | - | - | 57851 | 5537 |
| **3154** | 3155 | 18/06/20 | 8:00 AM | Maharashtra | - | - | 59166 | 5651 |
| **3190** | 3191 | 19/06/20 | 8:00 AM | Maharashtra | - | - | 60838 | 5751 |
| **3226** | 3227 | 20/06/20 | 8:00 AM | Maharashtra | - | - | 62773 | 5893 |
| **3262** | 3263 | 21/06/20 | 8:00 AM | Maharashtra | - | - | 64153 | 5984 |
| **3298** | 3299 | 22/06/20 | 8:00 AM | Maharashtra | - | - | 65744 | 6170 |

```
covid_df[covid_df["State/UnionTerritory"].isin(["Bihar","Delhi"])]
```

| | Sno | Date | Time | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths |
|---|---|---|---|---|---|---|---|---|
| **34** | 35 | 02/03/20 | 6:00 PM | Delhi | 1 | 0 | 0 | 0 |
| **38** | 39 | 03/03/20 | 6:00 PM | Delhi | 1 | 0 | 0 | 0 |
| **42** | 43 | 04/03/20 | 6:00 PM | Delhi | 1 | 0 | 0 | 0 |
| **45** | 46 | 05/03/20 | 6:00 PM | Delhi | 2 | 0 | 0 | 0 |
| **51** | 52 | 06/03/20 | 6:00 PM | Delhi | 3 | 0 | 0 | 0 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... |
| **6919** | 6920 | 03/10/20 | 8:00 AM | Delhi | - | - | 253784 | 5438 |
| **6950** | 6951 | 04/10/20 | 8:00 AM | Bihar | - | - | 173932 | 912 |
| **6954** | 6955 | 04/10/20 | 8:00 AM | Delhi | - | - | 257224 | 5472 |
| **6985** | 6986 | 05/10/20 | 8:00 AM | Bihar | - | - | 175458 | 915 |
| **6989** | 6990 | 05/10/20 | 8:00 AM | Delhi | - | - | 260350 | 5510 |

416 rows × 9 columns

```
covid_df.drop(["Time"],axis='columns' ,inplace=True)
```

```
covid_df
```

| | Sno | Date | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths | Conf |
|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 30/01/20 | Kerala | 1 | 0 | 0 | 0 | |
| **1** | 2 | 31/01/20 | Kerala | 1 | 0 | 0 | 0 | |
| **2** | 3 | 01/02/20 | Kerala | 2 | 0 | 0 | 0 | |
| **3** | 4 | 02/02/20 | Kerala | 3 | 0 | 0 | 0 | |
| **4** | 5 | 03/02/20 | Kerala | 3 | 0 | 0 | 0 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **7011** | 7012 | 05/10/20 | Telengana | - | - | 172388 | 1171 | 20 |
| **7012** | 7013 | 05/10/20 | Tripura | - | - | 21876 | 299 | 2 |
| **7013** | 7014 | 05/10/20 | Uttarakhand | - | - | 41740 | 652 | 5 |
| **7014** | 7015 | 05/10/20 | Uttar Pradesh | - | - | 362052 | 6029 | 41 |
| **7015** | 7016 | 05/10/20 | West Bengal | - | - | 237698 | 5194 | 27 |

7016 rows × 8 columns

```
covid_df['RecRate'] = covid_df['Cured']/covid_df['Confirmed']
covid_df.sort_values("RecRate", ascending=False)
```

| | Sno | Date | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths | Confir |
|---|---|---|---|---|---|---|---|---|
| **1465** | 1466 | 30/04/20 | Manipur | - | - | 2 | 0 | |
| **1369** | 1370 | 27/04/20 | Manipur | - | - | 2 | 0 | |
| **2293** | 2294 | 25/05/20 | Mizoram | - | - | 1 | 0 | |
| **1410** | 1411 | 28/04/20 | Tripura | - | - | 2 | 0 | |
| **1825** | 1826 | 11/05/20 | Mizoram | - | - | 1 | 0 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **1047** | 1048 | 17/04/20 | Nagaland | - | - | 0 | 0 | |
| **1080** | 1081 | 18/04/20 | Nagaland | - | - | 0 | 0 | |
| **1113** | 1114 | 19/04/20 | Nagaland | - | - | 0 | 0 | |
| **1146** | 1147 | 20/04/20 | Nagaland | - | - | 0 | 0 | |
| **1179** | 1180 | 21/04/20 | Nagaland | - | - | 0 | 0 | |

7016 rows × 9 columns

```
new_df1=covid_df.groupby("Date")[["Cured","Deaths","Confirmed"]].sum()
new_df1
```

| | Cured | Deaths | Confirmed |
|---|---|---|---|
| **Date** | | | |

| Date | Cured | Deaths | Confirmed |
|---|---|---|---|
| **01/02/20** | 0 | 0 | 2 |
| **01/03/20** | 0 | 0 | 3 |
| **01/04/20** | 144 | 41 | 1834 |
| **01/05/20** | 9065 | 1152 | 34972 |
| **01/06/20** | 91819 | 5394 | 190535 |
| **...** | ... | ... | ... |
| **31/01/20** | 0 | 0 | 1 |
| **31/03/20** | 124 | 35 | 1397 |
| **31/05/20** | 86984 | 5164 | 182143 |
| **31/07/20** | 1057805 | 35747 | 1638870 |
| **31/08/20** | 2774801 | 64469 | 3621245 |

250 rows × 3 columns

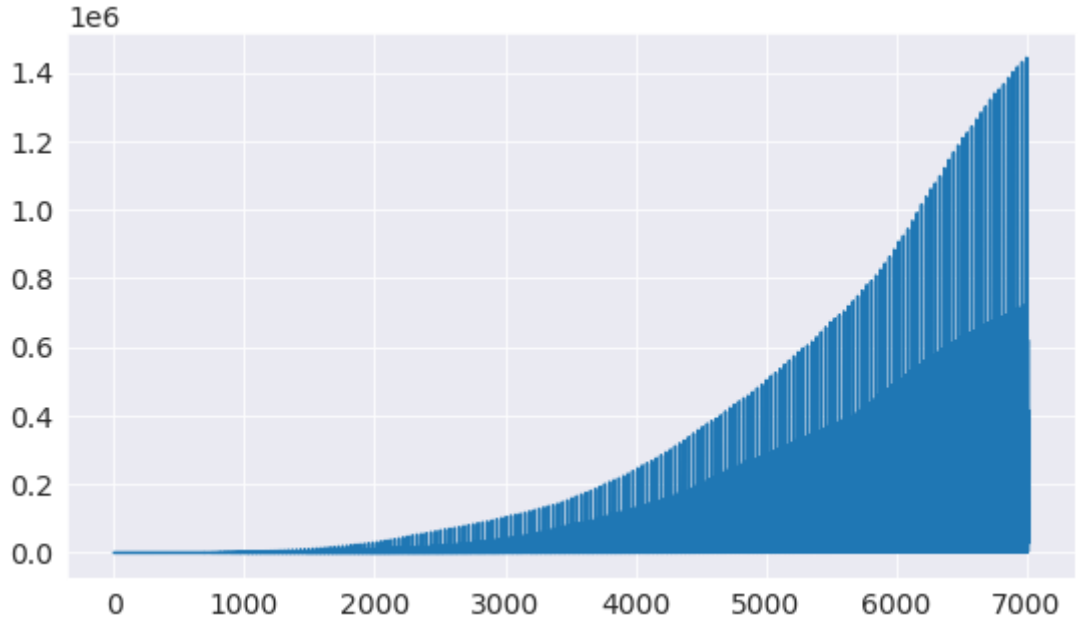
```
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

sns.set_style('darkgrid')
matplotlib.rcParams['font.size'] = 14
matplotlib.rcParams['figure.figsize'] = (9, 5)
matplotlib.rcParams['figure.facecolor'] = '#00000000'
```

```
new_df2=covid_df.groupby("Date")["Confirmed"].sum()
new_df2
```

```
Date
01/02/20          2
01/03/20          3
01/04/20       1834
01/05/20      34972
01/06/20     190535
             ...
31/01/20          1
31/03/20       1397
31/05/20     182143
31/07/20    1638870
31/08/20    3621245
Name: Confirmed, Length: 250, dtype: int64
```

```
covid_df.Confirmed.plot()
```

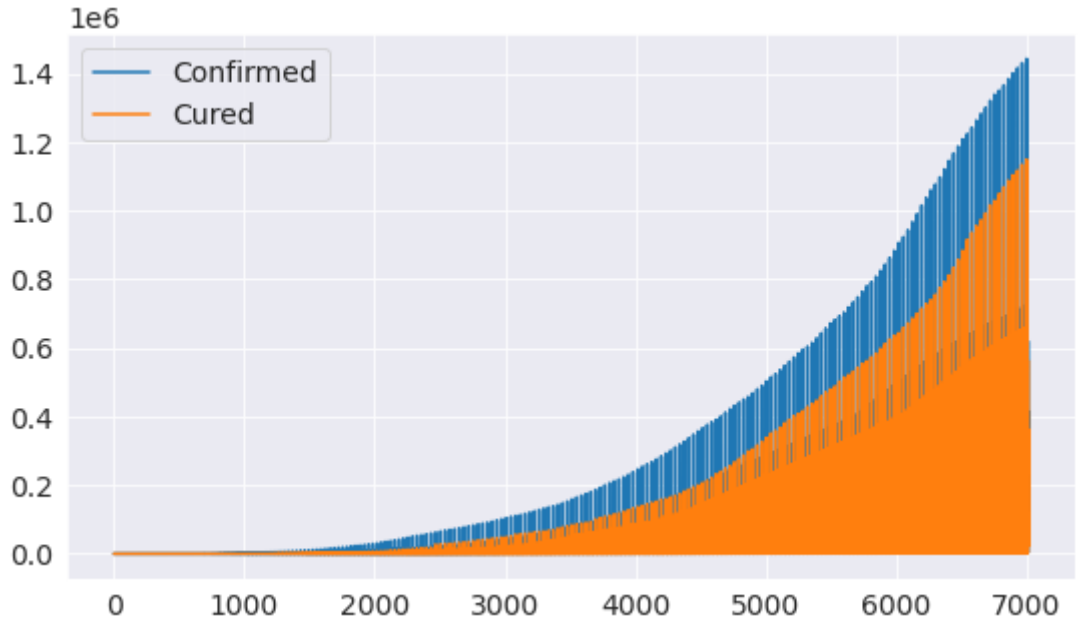
```
<AxesSubplot:>
```




## - we see that continously covid_19 cases increses

```
covid_df.Confirmed.plot()
covid_df.Cured.plot();
plt.legend(["Confirmed","Cured"])
```

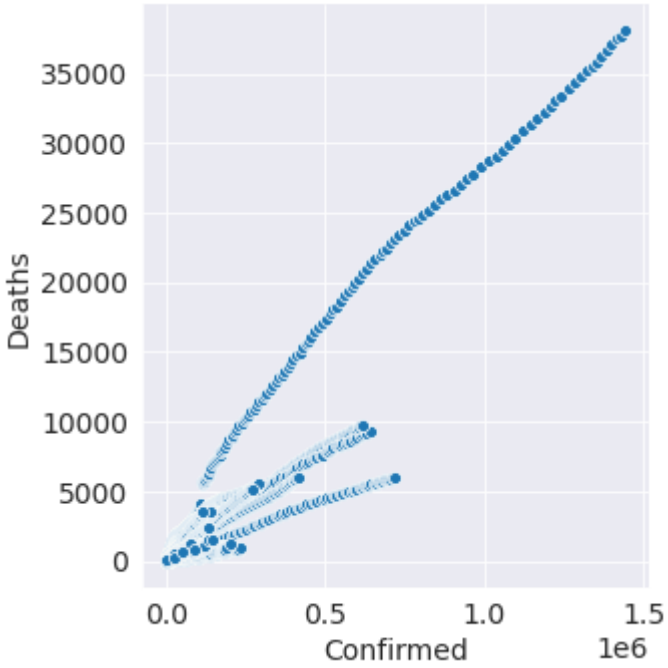
```
<matplotlib.legend.Legend at 0x7fdffb3c94c0>
```




```
## we we see that No of cured patients from covid_19 is slightly equal to confirmed cas
##hence most of the chance is for cure from corona
```

```
sns.relplot(x="Confirmed",y="Deaths", data=covid_df)
```

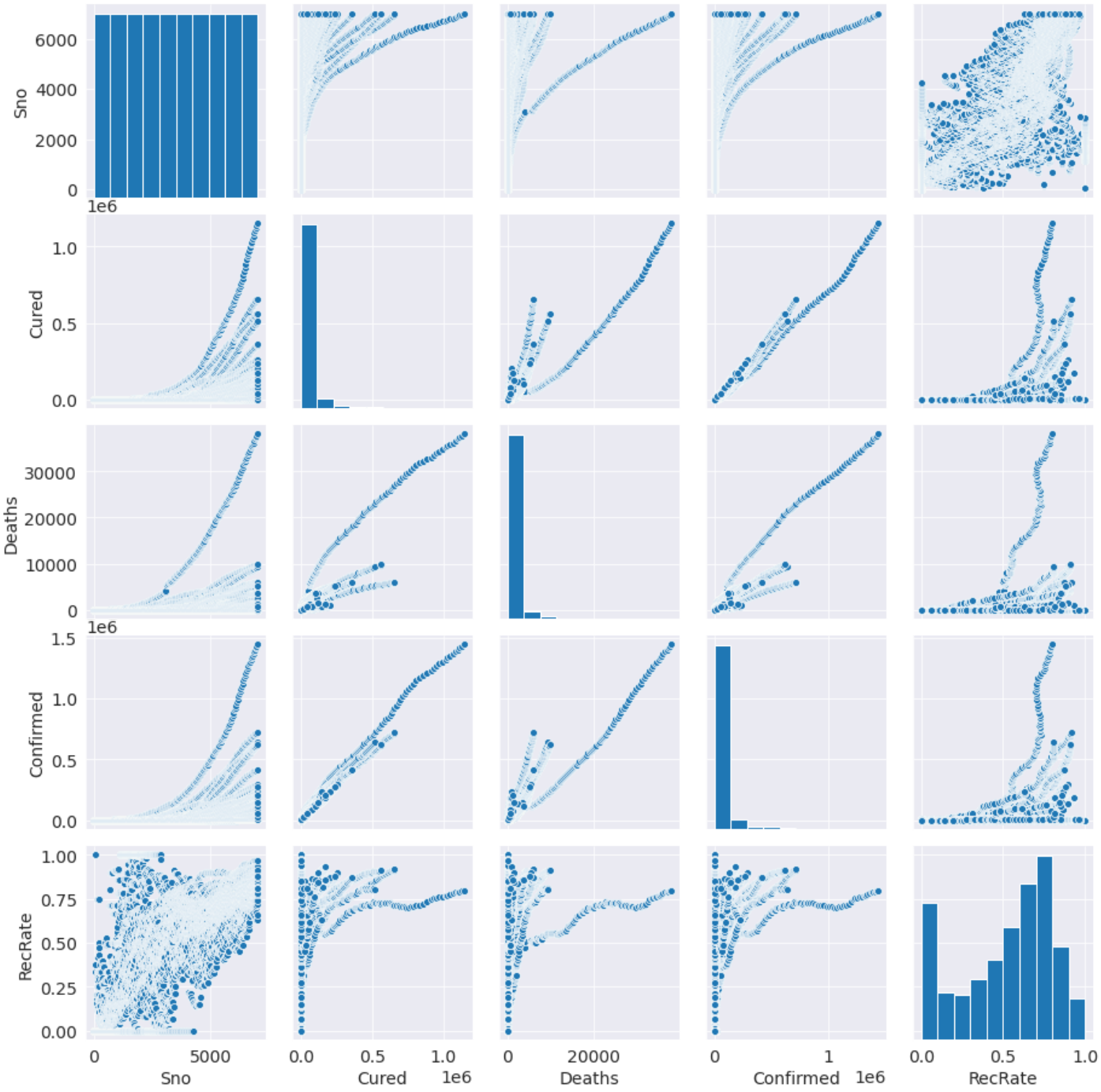
```
<seaborn.axisgrid.FacetGrid at 0x7fdffb43bac0>
```




## Death rate is very less

```
sns.pairplot(covid_df)
```

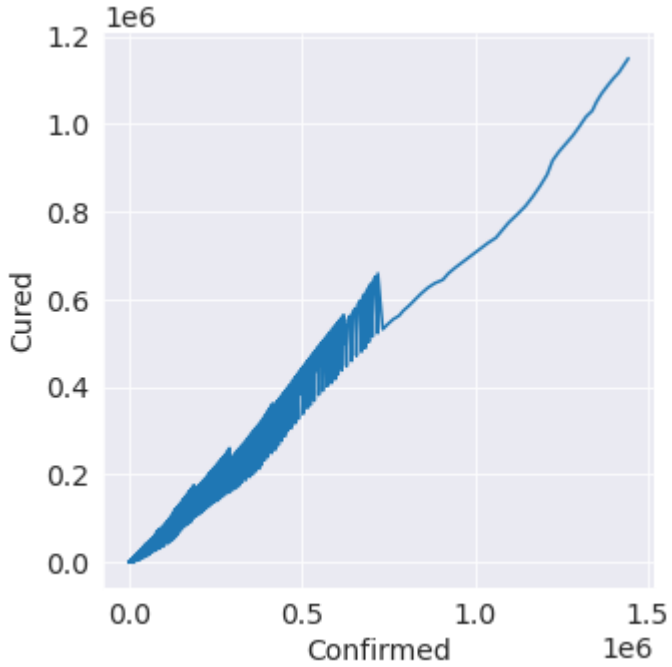
```
<seaborn.axisgrid.PairGrid at 0x7fdffb3f0b80>
```




from the above pair plot we visualize many graphs and find that Condirmed cases is increasing day by day and cured too. and many things

```
sns.relplot(x="Confirmed",y="Cured",kind='line', data=covid_df)
```

```
<seaborn.axisgrid.FacetGrid at 0x7fe0041d4d00>
```




Let us save and upload our work to Jovian before continuing

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Attempting to save notebook..
[jovian] Updating notebook "mddanishqamar179/covid-19-data-analysis" on https://jovian.ml/
[jovian] Uploading notebook..
[jovian] Capturing environment..
[jovian] Committed successfully! https://jovian.ml/mddanishqamar179/covid-19-data-analysis
```

```
'https://jovian.ml/mddanishqamar179/covid-19-data-analysis'
```

## Asking and Answering Questions

# After done of EDA and Visualization i have asked 5 Question from data set and find the answers either by using numpy/pandas or visualizing.

### Ques-1 Which state has highest no. of covid_19 cases in India?

```
covid_df.sort_values(["Confirmed"], ascending=False)
```

| | Sno | Date | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths | Cor |
|---|---|---|---|---|---|---|---|---|
| **7000** | 7001 | 05/10/20 | Maharashtra | - | - | 1149603 | 38084 | 14 |
| **6965** | 6966 | 04/10/20 | Maharashtra | - | - | 1134555 | 37758 | 14 |
| **6930** | 6931 | 03/10/20 | Maharashtra | - | - | 1117720 | 37480 | 14 |
| **6895** | 6896 | 02/10/20 | Maharashtra | - | - | 1104426 | 37056 | 14 |
| **6860** | 6861 | 01/10/20 | Maharashtra | - | - | 1088322 | 36662 | 13 |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **981** | 982 | 15/04/20 | Nagaland | - | - | 0 | 0 | |
| **1146** | 1147 | 20/04/20 | Nagaland | - | - | 0 | 0 | |
| **1047** | 1048 | 17/04/20 | Nagaland | - | - | 0 | 0 | |
| **1113** | 1114 | 19/04/20 | Nagaland | - | - | 0 | 0 | |
| **1179** | 1180 | 21/04/20 | Nagaland | - | - | 0 | 0 | |

7016 rows × 9 columns

# from the sorting the dataset we see that Maharastra has highest no of cases.

## Q2: Which state has least no. or has no any of Covid_19 cases in India?

```
covid_df.sort_values(["Confirmed"] , ascending=True)
```

| | Sno | Date | State/UnionTerritory | ConfirmedIndianNational | ConfirmedForeignNational | Cured | Deaths | Cor |
|---|---|---|---|---|---|---|---|---|
| **1047** | 1048 | 17/04/20 | Nagaland | - | - | 0 | 0 | |
| **1014** | 1015 | 16/04/20 | Nagaland | - | - | 0 | 0 | |
| **981** | 982 | 15/04/20 | Nagaland | - | - | 0 | 0 | |
| **1080** | 1081 | 18/04/20 | Nagaland | - | - | 0 | 0 | |
| **1179** | 1180 | 21/04/20 | Nagaland | - | - | 0 | 0 | |
| **...** | ... | ... | ... | ... | ... | ... | ... | |
| **6860** | 6861 | 01/10/20 | Maharashtra | - | - | 1088322 | 36662 | 13 |
| **6895** | 6896 | 02/10/20 | Maharashtra | - | - | 1104426 | 37056 | 14 |
| **6930** | 6931 | 03/10/20 | Maharashtra | - | - | 1117720 | 37480 | 14 |
| **6965** | 6966 | 04/10/20 | Maharashtra | - | - | 1134555 | 37758 | 14 |
| **7000** | 7001 | 05/10/20 | Maharashtra | - | - | 1149603 | 38084 | 14 |

7016 rows × 9 columns

# Nagaland has no any single cases of Covid_19 cases in India

```

```

## Q3: Top 5 states with highest no of covid cases?

```
high_cases=covid_df.groupby("State/UnionTerritory")[["Confirmed","Cured"]].sum()
```

```
high_cases.sort_values(["Confirmed"], ascending=False).head(5)
```

| State/UnionTerritory | Confirmed | Cured |
|---|---|---|
| **Maharashtra** | 71118458 | 48966523 |
| **Tamil Nadu** | 35346088 | 29079365 |
| **Andhra Pradesh** | 31237161 | 24462927 |
| **Karnataka** | 26069064 | 18503223 |
| **Uttar Pradesh** | 18247465 | 13645658 |

we see that Maharastra, Tamil Nadu , AP, Karnataka, and Up are Top 5 state with highest no. of cases.

## Q4: Top 10 state with highest Recovery

```
high_cured=covid_df.groupby("State/UnionTerritory")[["Confirmed","Cured"]].sum()
```

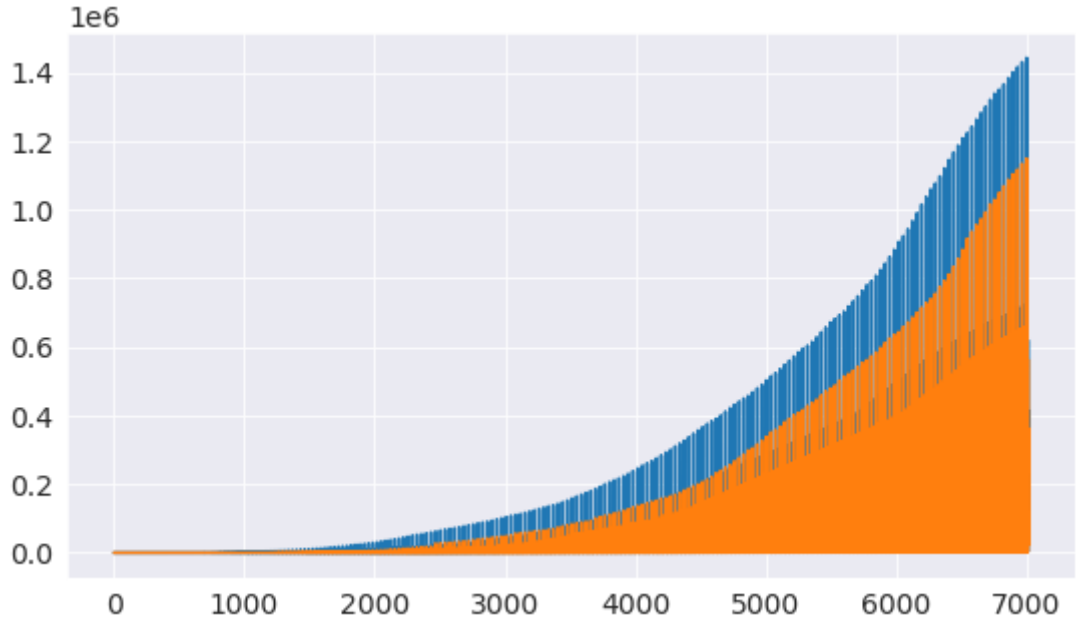
```
high_cured.sort_values(["Cured"],ascending=False).head(10)
```

| State/UnionTerritory | Confirmed | Cured |
|---|---|---|
| **Maharashtra** | 71118458 | 48966523 |
| **Tamil Nadu** | 35346088 | 29079365 |
| **Andhra Pradesh** | 31237161 | 24462927 |
| **Karnataka** | 26069064 | 18503223 |
| **Delhi** | 18176487 | 15017212 |
| **Uttar Pradesh** | 18247465 | 13645658 |
| **West Bengal** | 12696423 | 10160569 |
| **Bihar** | 9703526 | 8071758 |
| **Telengana** | 8956822 | 7018155 |
| **Gujarat** | 9063456 | 7001600 |

```

```

## Q5: visualize Confirmed cases Vs Cured

```
covid_df.Confirmed.plot()
covid_df.Cured.plot();
```




Let us save and upload our work to Jovian before continuing.

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Attempting to save notebook..
[jovian] Updating notebook "mddanishqamar179/covid-19-data-analysis" on https://jovian.ml/
[jovian] Uploading notebook..
[jovian] Capturing environment..
[jovian] Committed successfully! https://jovian.ml/mddanishqamar179/covid-19-data-analysis
```

```
'https://jovian.ml/mddanishqamar179/covid-19-data-analysis'
```

# Inferences and Conclusion

we analysis the covid_19 dataset of India, we use numpy , pandas, for sorting , cleaning , merging, subsetting the data . For visulaziaion we use Matplotlib and seaborn and plotted many graphs to understand. we see that Maharastra has higest no. of cases, as well as highest no. of Recovery patients. Using this EDA of covid_19 dataset we can analysis the covid cases in for better prevenation.

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Attempting to save notebook..
[jovian] Updating notebook "mddanishqamar179/covid-19-data-analysis" on https://jovian.ml/
[jovian] Uploading notebook..
[jovian] Capturing environment..
[jovian] Committed successfully! https://jovian.ml/mddanishqamar179/covid-19-data-analysis
```

```
'https://jovian.ml/mddanishqamar179/covid-19-data-analysis'
```

## References and Future Work

on analysis of given dataset we can use for better information regarding covid_19 cases and can upgrade our health system of that state which has least recovery rate learning highest no of recovery rate states

```
import jovian
```

```
jovian.commit()
```

```
[jovian] Attempting to save notebook..
[jovian] Updating notebook "mddanishqamar179/covid-19-india" on https://jovian.ml/
[jovian] Uploading notebook..
[jovian] Capturing environment..
[jovian] Committed successfully! https://jovian.ml/mddanishqamar179/covid-19-india
```

```
'https://jovian.ml/mddanishqamar179/covid-19-india'
```

```

```

```

```

```

```