

```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

```
In [2]: data = pd.read_csv('covid_19_india.csv')
data
```

Out[2]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed
0	1	2020-01-30	6:00 PM	Kerala	1	0	0	0	1
1	2	2020-01-31	6:00 PM	Kerala	1	0	0	0	1
2	3	2020-02-01	6:00 PM	Kerala	2	0	0	0	2
3	4	2020-02-02	6:00 PM	Kerala	3	0	0	0	3
4	5	2020-02-03	6:00 PM	Kerala	3	0	0	0	3
...
16845	16846	2021-07-07	8:00 AM	Telangana	-	-	613124	3703	628282
16846	16847	2021-07-07	8:00 AM	Tripura	-	-	63964	701	68612
16847	16848	2021-07-07	8:00 AM	Uttarakhand	-	-	332006	7338	340882
16848	16849	2021-07-07	8:00 AM	Uttar Pradesh	-	-	1682130	22656	1706818
16849	16850	2021-07-07	8:00 AM	West Bengal	-	-	1472132	17834	1507241

16850 rows × 9 columns

In []:

In [3]: data.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16850 entries, 0 to 16849
Data columns (total 9 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Sno                                    16850 non-null  int64
1   Date                                  16850 non-null  object
2   Time                                  16850 non-null  object
3   State/UnionTerritory                 16850 non-null  object
4   ConfirmedIndianNational              16850 non-null  object
5   ConfirmedForeignNational             16850 non-null  object
6   Cured                                16850 non-null  int64
7   Deaths                              16850 non-null  int64
8   Confirmed                            16850 non-null  int64
dtypes: int64(4), object(5)
memory usage: 1.2+ MB
```

In [4]: data.isnull().sum()

```
Out[4]: Sno                0
Date                0
Time                0
State/UnionTerritory  0
ConfirmedIndianNational  0
ConfirmedForeignNational  0
Cured                0
Deaths              0
Confirmed            0
dtype: int64
```

In [5]: data.describe()

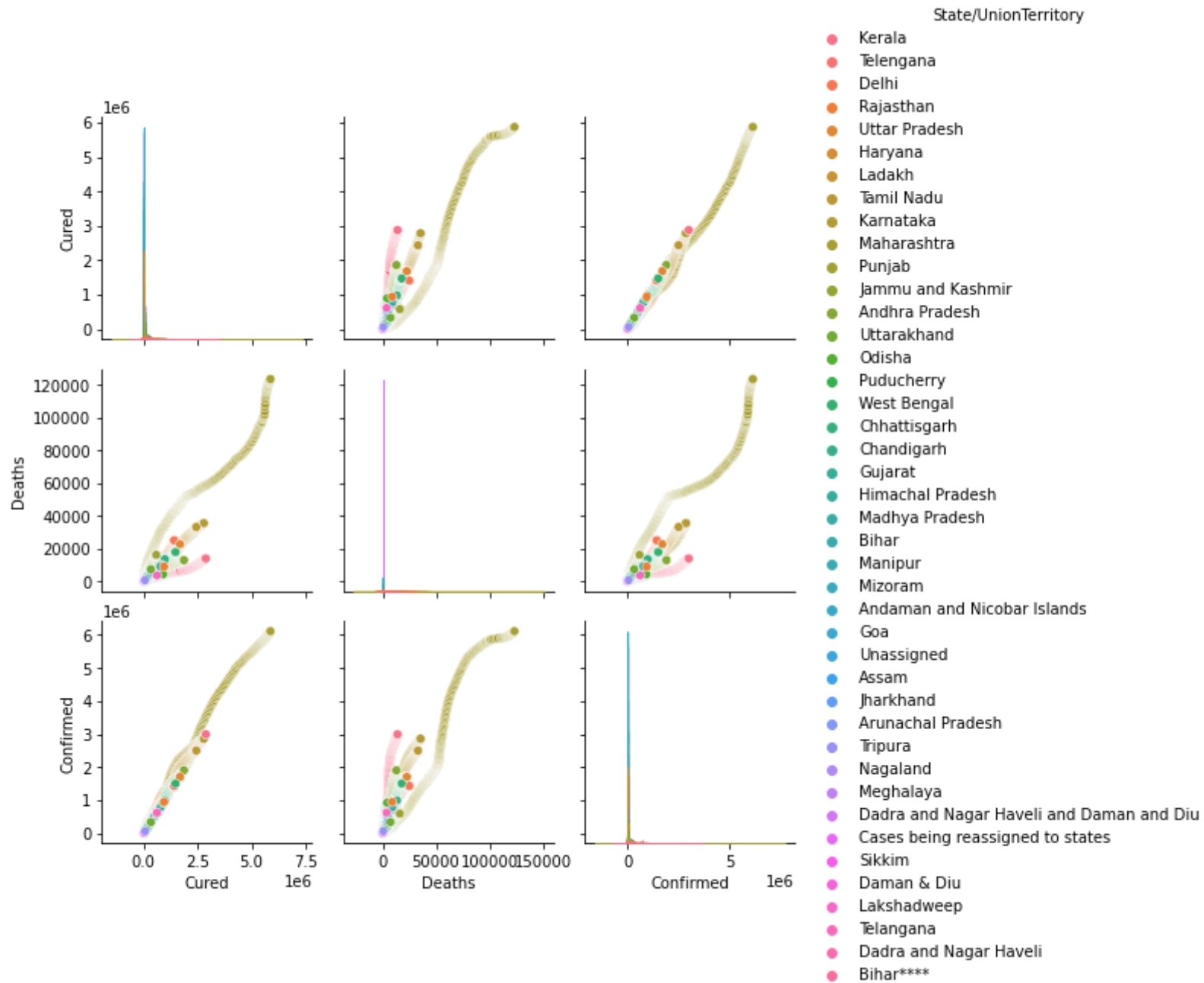
Out[5]:

	Sno	Cured	Deaths	Confirmed
count	16850.000000	1.685000e+04	16850.000000	1.685000e+04
mean	8425.500000	2.360353e+05	3485.222552	2.583667e+05
std	4864.320353	5.225438e+05	9330.541749	5.672808e+05
min	1.000000	0.000000e+00	0.000000	0.000000e+00
25%	4213.250000	2.658500e+03	22.000000	3.644750e+03
50%	8425.500000	2.889500e+04	453.000000	3.336150e+04
75%	12637.750000	2.537510e+05	3071.250000	2.666530e+05
max	16850.000000	5.872268e+06	123531.000000	6.113335e+06

In []:

```
In [6]: fig=plt.figure(figsize=(10,6))
sns.pairplot(data.drop(['Sno'],axis=1),hue="State/UnionTerritory")
plt.show
```

```
Out[6]: <function matplotlib.pyplot.show(close=None, block=None)>
<Figure size 720x432 with 0 Axes>
```

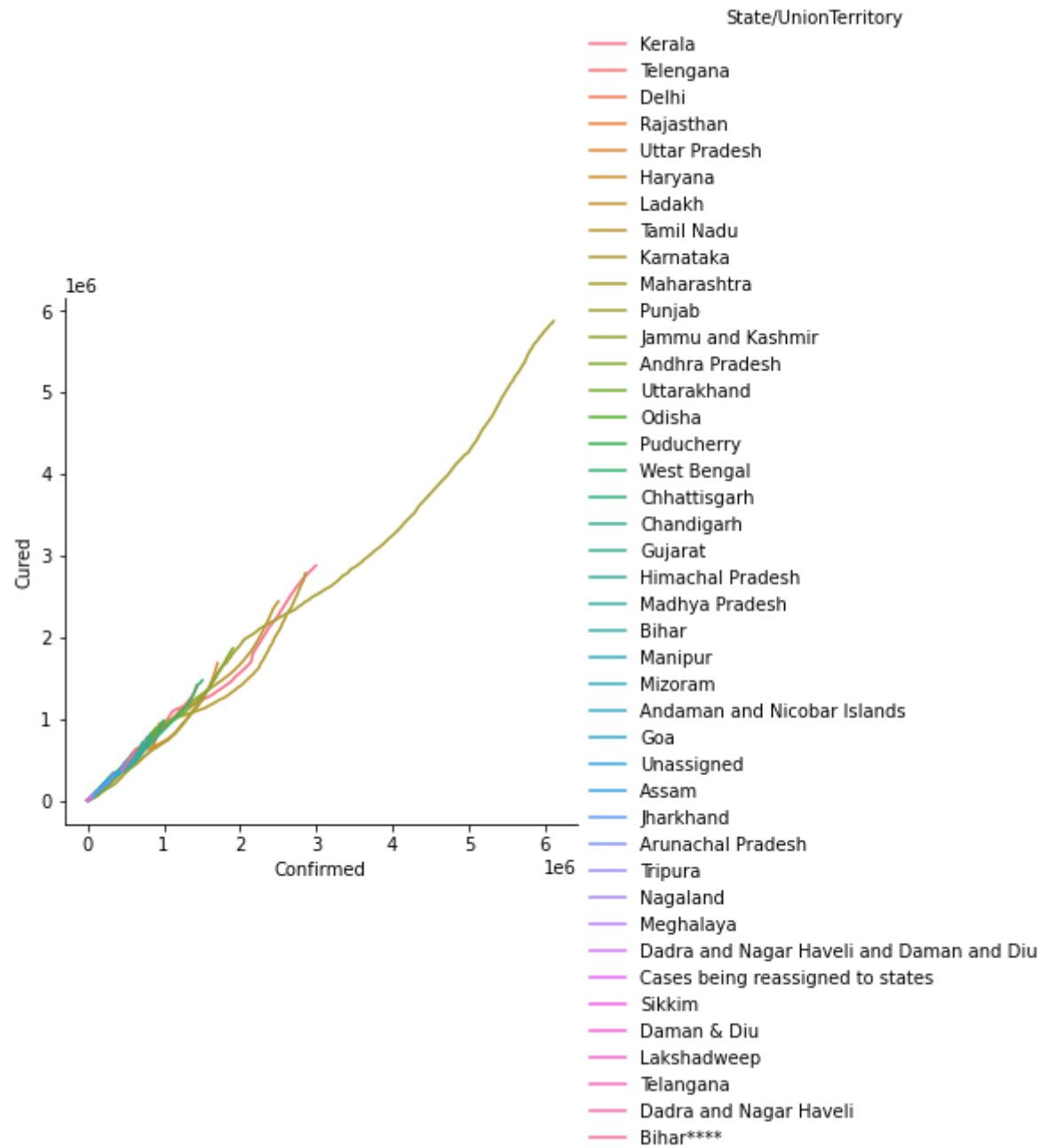


```
In [7]: data.columns
```

```
Out[7]: Index(['Sno', 'Date', 'Time', 'State/UnionTerritory',  
              'ConfirmedIndianNational', 'ConfirmedForeignNational', 'Cured',  
              'Deaths', 'Confirmed'],  
             dtype='object')
```

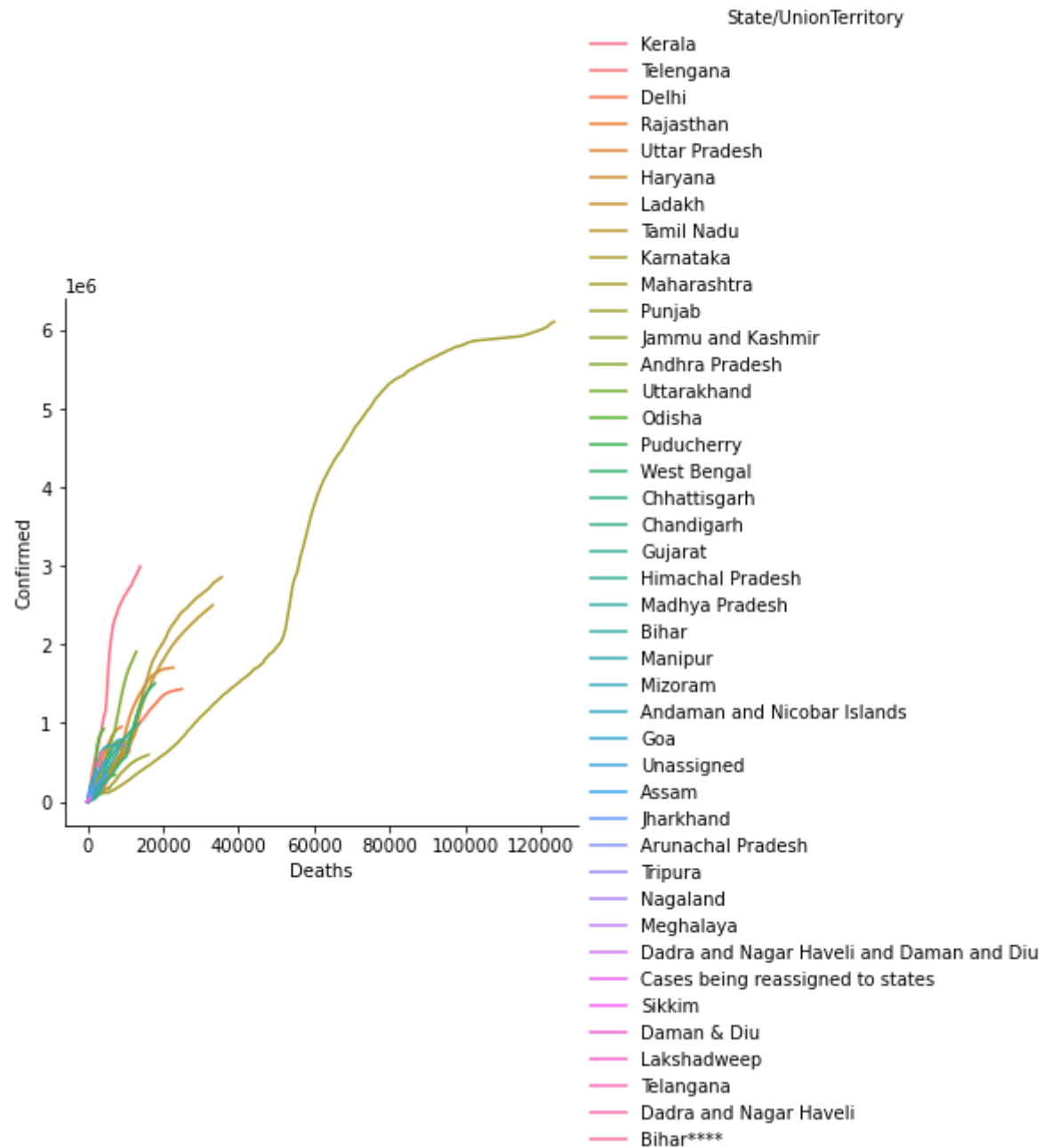
```
In [8]: fig=plt.figure(figsize=(10,10))
sns.relplot(x='Confirmed',y='Cured',hue=data['State/UnionTerritory'],data=data,kind='line')
plt.show()
```

<Figure size 720x720 with 0 Axes>




```
In [9]: sns.relplot(y='Confirmed',x='Deaths',hue=data['State/UnionTerritory'],data=data,kind='line')
```

```
Out[9]: <seaborn.axisgrid.FacetGrid at 0x18046194df0>
```



In []:

```
In [10]: covid19=data.copy()  
covid19
```

Out[10]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed
0	1	2020-01-30	6:00 PM	Kerala	1	0	0	0	1
1	2	2020-01-31	6:00 PM	Kerala	1	0	0	0	1
2	3	2020-02-01	6:00 PM	Kerala	2	0	0	0	2
3	4	2020-02-02	6:00 PM	Kerala	3	0	0	0	3
4	5	2020-02-03	6:00 PM	Kerala	3	0	0	0	3
...
16845	16846	2021-07-07	8:00 AM	Telangana	-	-	613124	3703	628282
16846	16847	2021-07-07	8:00 AM	Tripura	-	-	63964	701	68612
16847	16848	2021-07-07	8:00 AM	Uttarakhand	-	-	332006	7338	340882
16848	16849	2021-07-07	8:00 AM	Uttar Pradesh	-	-	1682130	22656	1706818
16849	16850	2021-07-07	8:00 AM	West Bengal	-	-	1472132	17834	1507241

16850 rows × 9 columns

```
In [11]: covid19['Active'] = covid19['Confirmed'] - (covid19['Cured'] + covid19['Deaths'])
covid19
```

Out[11]:

	Sno	Date	Time	State/UnionTerritory	ConfirmedIndianNational	ConfirmedForeignNational	Cured	Deaths	Confirmed	Active
0	1	2020-01-30	6:00 PM	Kerala	1	0	0	0	1	1
1	2	2020-01-31	6:00 PM	Kerala	1	0	0	0	1	1
2	3	2020-02-01	6:00 PM	Kerala	2	0	0	0	2	2
3	4	2020-02-02	6:00 PM	Kerala	3	0	0	0	3	3
4	5	2020-02-03	6:00 PM	Kerala	3	0	0	0	3	3
...
16845	16846	2021-07-07	8:00 AM	Telangana	-	-	613124	3703	628282	11455
16846	16847	2021-07-07	8:00 AM	Tripura	-	-	63964	701	68612	3947
16847	16848	2021-07-07	8:00 AM	Uttarakhand	-	-	332006	7338	340882	1538
16848	16849	2021-07-07	8:00 AM	Uttar Pradesh	-	-	1682130	22656	1706818	2032
16849	16850	2021-07-07	8:00 AM	West Bengal	-	-	1472132	17834	1507241	17275

16850 rows × 10 columns

```
In [12]: covid19.drop(columns = ['Sno', 'ConfirmedIndianNational', 'ConfirmedForeignNational'], inplace=True)
```

```
In [13]: covid19['State/UnionTerritory'].unique()
```

```
Out[13]: array(['Kerala', 'Telengana', 'Delhi', 'Rajasthan', 'Uttar Pradesh',  
                'Haryana', 'Ladakh', 'Tamil Nadu', 'Karnataka', 'Maharashtra',  
                'Punjab', 'Jammu and Kashmir', 'Andhra Pradesh', 'Uttarakhand',  
                'Odisha', 'Puducherry', 'West Bengal', 'Chhattisgarh',  
                'Chandigarh', 'Gujarat', 'Himachal Pradesh', 'Madhya Pradesh',  
                'Bihar', 'Manipur', 'Mizoram', 'Andaman and Nicobar Islands',  
                'Goa', 'Unassigned', 'Assam', 'Jharkhand', 'Arunachal Pradesh',  
                'Tripura', 'Nagaland', 'Meghalaya',  
                'Dadra and Nagar Haveli and Daman and Diu',  
                'Cases being reassigned to states', 'Sikkim', 'Daman & Diu',  
                'Lakshadweep', 'Telangana', 'Dadra and Nagar Haveli', 'Bihar****'],  
              dtype=object)
```

Data Cleansing

```
In [14]: covid19.loc[covid19['State/UnionTerritory']=='Bihar****', 'State/UnionTerritory']='Bihar'
```

```
In [15]: covid19.loc[covid19['State/UnionTerritory']=='Daman & Diu', 'State/UnionTerritory']='Dadra and Nagar Haveli and Daman and
```

```
In [16]: covid19.loc[covid19['State/UnionTerritory']=='Dadra and Nagar Haveli', 'State/UnionTerritory']='Dadra and Nagar Haveli and
```



```
In [17]: covid19.loc[covid19['State/UnionTerritory']=='Telengana', 'State/UnionTerritory']='Telangana'
```

```
In [18]: covid19.groupby('State/UnionTerritory').count()
```

Out[18]:

	Date	Time	Cured	Deaths	Confirmed	Active
State/UnionTerritory						
Andaman and Nicobar Islands	469	469	469	469	469	469
Andhra Pradesh	483	483	483	483	483	483
Arunachal Pradesh	461	461	461	461	461	461
Assam	463	463	463	463	463	463
Bihar	473	473	473	473	473	473
Cases being reassigned to states	60	60	60	60	60	60
Chandigarh	476	476	476	476	476	476
Chhattisgarh	476	476	476	476	476	476
Dadra and Nagar Haveli and Daman and Diu	429	429	429	429	429	429
Delhi	493	493	493	493	493	493
Goa	469	469	469	469	469	469
Gujarat	475	475	475	475	475	475
Haryana	491	491	491	491	491	491
Himachal Pradesh	474	474	474	474	474	474
Jammu and Kashmir	486	486	486	486	486	486
Jharkhand	463	463	463	463	463	463
Karnataka	486	486	486	486	486	486
Kerala	525	525	525	525	525	525
Ladakh	488	488	488	488	488	488
Lakshadweep	209	209	209	209	209	209
Madhya Pradesh	474	474	474	474	474	474
Maharashtra	486	486	486	486	486	486
Manipur	471	471	471	471	471	471

	Date	Time	Cured	Deaths	Confirmed	Active
State/UnionTerritory						
Meghalaya	450	450	450	450	450	450
Mizoram	470	470	470	470	470	470
Nagaland	417	417	417	417	417	417
Odisha	479	479	479	479	479	479
Puducherry	477	477	477	477	477	477
Punjab	486	486	486	486	486	486
Rajasthan	492	492	492	492	492	492
Sikkim	410	410	410	410	410	410
Tamil Nadu	488	488	488	488	488	488
Telangana	493	493	493	493	493	493
Tripura	457	457	457	457	457	457
Unassigned	3	3	3	3	3	3
Uttar Pradesh	491	491	491	491	491	491
Uttarakhand	480	480	480	480	480	480
West Bengal	477	477	477	477	477	477

India Covid-19 Statewise Data with Visualization

```
In [19]: allstates=covid19["State/UnionTerritory"].value_counts()  
allstates
```

```
Out[19]: Kerala                    525  
Delhi                            493  
Telangana                       493  
Rajasthan                      492  
Uttar Pradesh                  491  
Haryana                       491  
Ladakh                         488  
Tamil Nadu                    488  
Punjab                        486  
Jammu and Kashmir             486  
Maharashtra                   486  
Karnataka                    486  
Andhra Pradesh               483  
Uttarakhand                  480  
Odisha                      479  
Puducherry                   477  
West Bengal                  477  
Chhattisgarh                 476  
Chandigarh                   476  
Gujarat                      475  
Himachal Pradesh             474  
Madhya Pradesh               474  
Bihar                       473  
Manipur                     471  
Mizoram                     470  
Andaman and Nicobar Islands  469  
Goa                         469  
Assam                      463  
Jharkhand                   463  
Arunachal Pradesh           461  
Tripura                    457  
Meghalaya                   450  
Dadra and Nagar Haveli and Daman and Diu 429  
Nagaland                    417  
Sikkim                      410  
Lakshadweep                 209  
Cases being reassigned to states    60
```


Unassigned
Name: State/UnionTerritory, dtype: int64

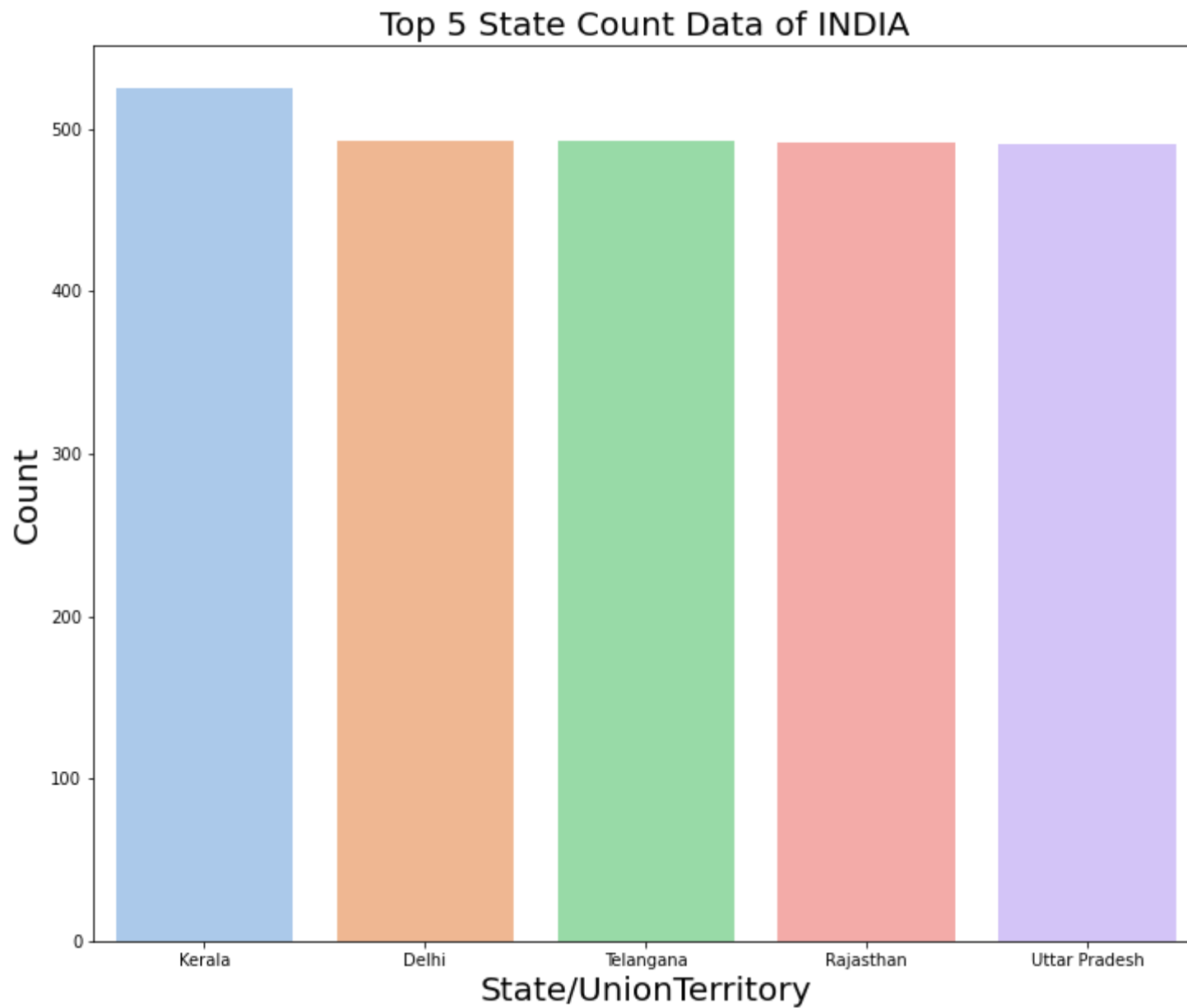
3

```
In [20]: top5_states_count=allstates.head()  
top5_states_count
```

```
Out[20]: Kerala          525  
Delhi              493  
Telangana          493  
Rajasthan          492  
Uttar Pradesh      491  
Name: State/UnionTerritory, dtype: int64
```

```
In [21]: fig = plt.figure()
fig.set_figheight(10)
fig.set_figwidth(12)

sns.barplot(x=top5_states_count.index,y=top5_states_count.values,palette='pastel')
plt.xlabel('State/UnionTerritory',size=20)
plt.ylabel('Count',size=20)
plt.title("Top 5 State Count Data of INDIA",size=20)
plt.show()
```



In []:

```
In [22]: total_cured_sum=covid19['Cured'].sum()  
total_cured_sum
```

```
Out[22]: 3977194136
```

```
In [23]: total_deaths_sum=covid19['Deaths'].sum()  
total_deaths_sum
```

```
Out[23]: 58726000
```

```
In [24]: total_confirmed_sum= covid19['Confirmed'].sum()  
total_confirmed_sum
```

```
Out[24]: 4353478074
```

```
In [25]: total_active_cases_sum=covid19['Active'].sum()  
total_active_cases_sum
```

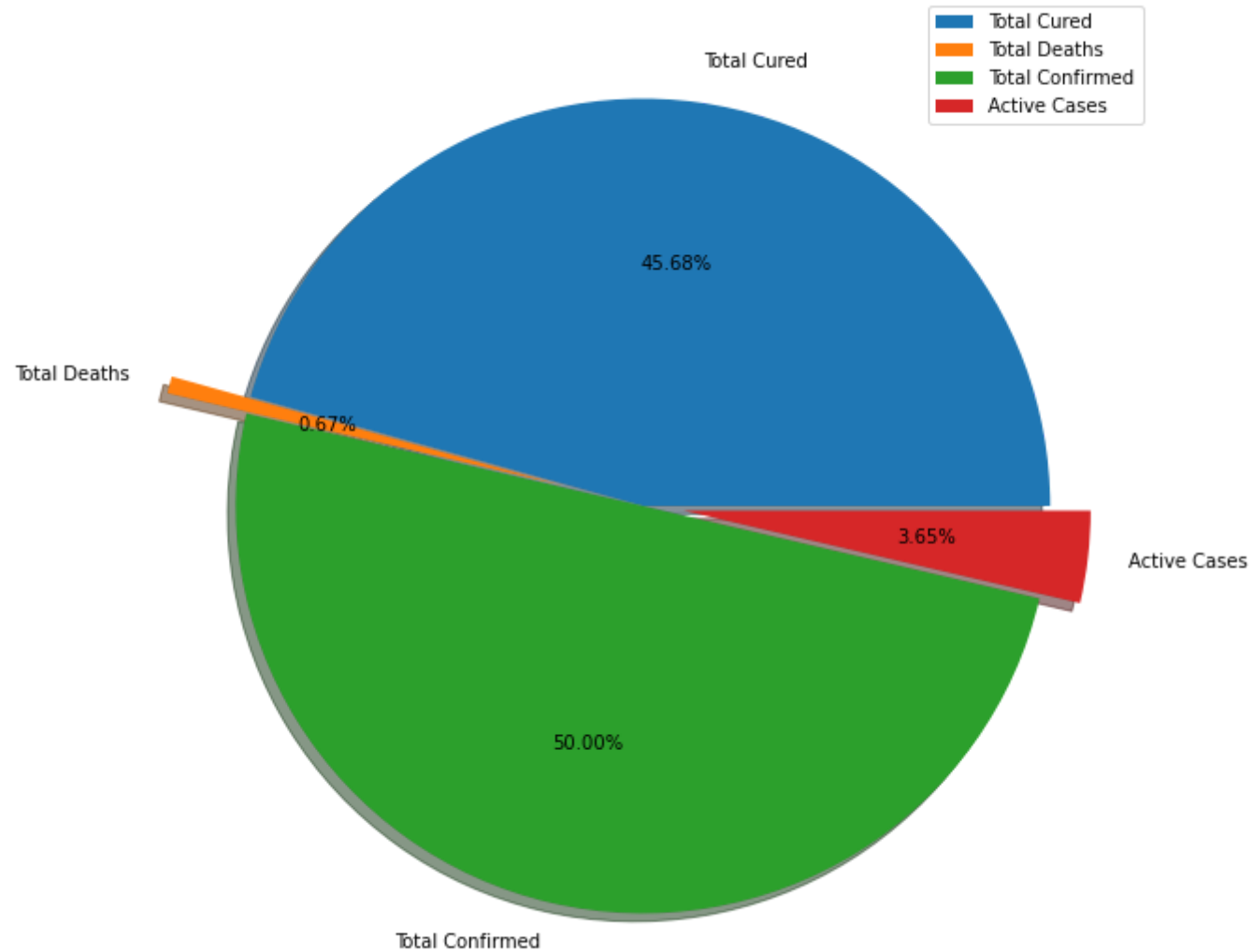
```
Out[25]: 317557938
```

```
In [26]: result=np.array([total_cured_sum,total_deaths_sum,total_confirmed_sum,total_active_cases_sum])  
lable = ["Total Cured","Total Deaths","Total Confirmed","Active Cases"]  
e = [0.0,0.2,0.0,0.1]
```

```
In [27]: fig = plt.figure()
fig.set_figheight(10)
fig.set_figwidth(12)

plt.pie(result, labels=lable, autopct='%1.2f%%', explode=e, shadow=True)
plt.title("Total COVID-19 INDIA Data", size=20, color='k')
plt.legend(lable)
plt.show()
```

Total COVID-19 INDIA Data



In []:

In []:

In [28]: covid19

Out[28]:

	Date	Time	State/UnionTerritory	Cured	Deaths	Confirmed	Active
0	2020-01-30	6:00 PM	Kerala	0	0	1	1
1	2020-01-31	6:00 PM	Kerala	0	0	1	1
2	2020-02-01	6:00 PM	Kerala	0	0	2	2
3	2020-02-02	6:00 PM	Kerala	0	0	3	3
4	2020-02-03	6:00 PM	Kerala	0	0	3	3
...
16845	2021-07-07	8:00 AM	Telangana	613124	3703	628282	11455
16846	2021-07-07	8:00 AM	Tripura	63964	701	68612	3947
16847	2021-07-07	8:00 AM	Uttarakhand	332006	7338	340882	1538
16848	2021-07-07	8:00 AM	Uttar Pradesh	1682130	22656	1706818	2032
16849	2021-07-07	8:00 AM	West Bengal	1472132	17834	1507241	17275

16850 rows × 7 columns

In []:

```
In [29]: allstates_max=covid19.groupby('State/UnionTerritory')[['Active', 'Cured', 'Deaths', 'Confirmed']].max().sort_values(by='Confirmed', ascending=False)
allstates_max=allstates_max.reset_index()
allstates_max
```

Out[29]:

	State/UnionTerritory	Active	Cured	Deaths	Confirmed
0	Maharashtra	701614	5872268	123531	6113335
1	Kerala	445692	2877557	13960	2996094
2	Karnataka	605515	2784030	35526	2859595
3	Tamil Nadu	313048	2435872	33132	2503481
4	Andhra Pradesh	211554	1861937	12898	1908065
5	Uttar Pradesh	310783	1682130	22656	1706818
6	West Bengal	132181	1472132	17834	1507241
7	Delhi	103424	1408853	25001	1434687
8	Chhattisgarh	131245	977893	13462	996359
9	Rajasthan	212753	942882	8942	952836
10	Odisha	106493	897362	4299	927186
11	Gujarat	148297	811699	10072	823964
12	Madhya Pradesh	111366	780578	9017	790042
13	Haryana	116867	758442	9506	769030
14	Bihar	115152	711913	9612	722746
15	Telangana	80695	613124	3703	628282
16	Punjab	79963	578590	16131	596736
17	Assam	56295	493306	4717	522267
18	Jharkhand	61195	340365	5118	346038
19	Uttarakhand	80000	332006	7338	340882
20	Jammu and Kashmir	52848	309554	4345	317481
21	Himachal Pradesh	40008	198134	3485	202945

	State/UnionTerritory	Active	Cured	Deaths	Confirmed
22	Goa	32953	162787	3079	167823
23	Puducherry	18277	114673	1763	118227
24	Manipur	9613	66132	1218	73581
25	Tripura	8302	63964	701	68612
26	Chandigarh	8653	60837	809	61752
27	Meghalaya	8255	47173	880	52358
28	Arunachal Pradesh	3918	34525	181	37879
29	Nagaland	5049	23982	503	25619
30	Mizoram	4471	18383	98	22155
31	Sikkim	4306	19200	309	21403
32	Ladakh	2041	19733	204	20137
33	Dadra and Nagar Haveli and Daman and Diu	2081	10532	4	10575
34	Lakshadweep	2320	9643	49	9947
35	Cases being reassigned to states	9265	0	0	9265
36	Andaman and Nicobar Islands	1154	7343	128	7487
37	Unassigned	77	0	0	77

In []:

```
In [30]: top_10=allstates_max.head(10)
top_10
```

Out[30]:

	State/UnionTerritory	Active	Cured	Deaths	Confirmed
0	Maharashtra	701614	5872268	123531	6113335
1	Kerala	445692	2877557	13960	2996094
2	Karnataka	605515	2784030	35526	2859595
3	Tamil Nadu	313048	2435872	33132	2503481
4	Andhra Pradesh	211554	1861937	12898	1908065
5	Uttar Pradesh	310783	1682130	22656	1706818
6	West Bengal	132181	1472132	17834	1507241
7	Delhi	103424	1408853	25001	1434687
8	Chhattisgarh	131245	977893	13462	996359
9	Rajasthan	212753	942882	8942	952836

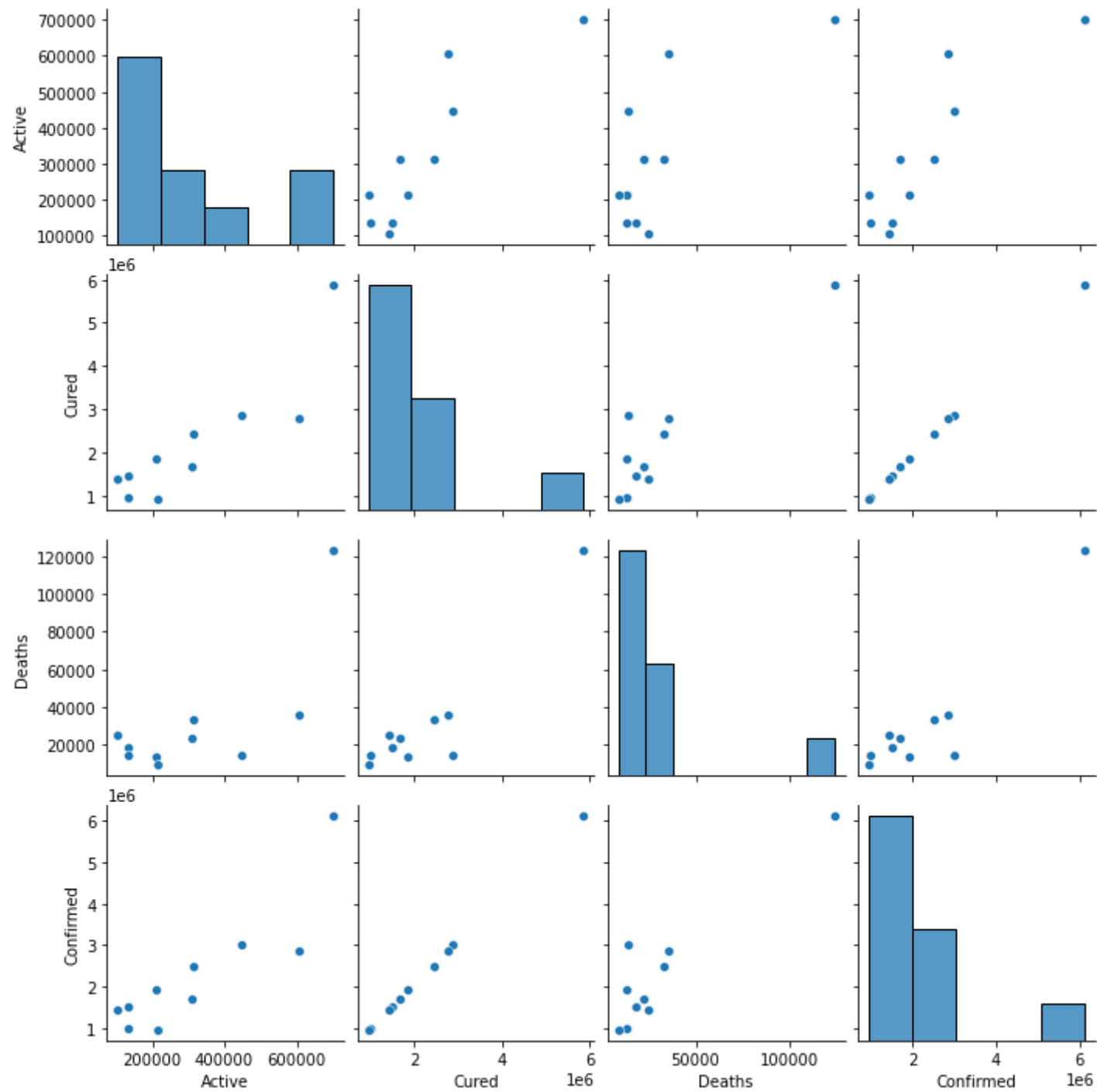
```
In [31]: top_10.columns
```

Out[31]: Index(['State/UnionTerritory', 'Active', 'Cured', 'Deaths', 'Confirmed'], dtype='object')

In [32]:

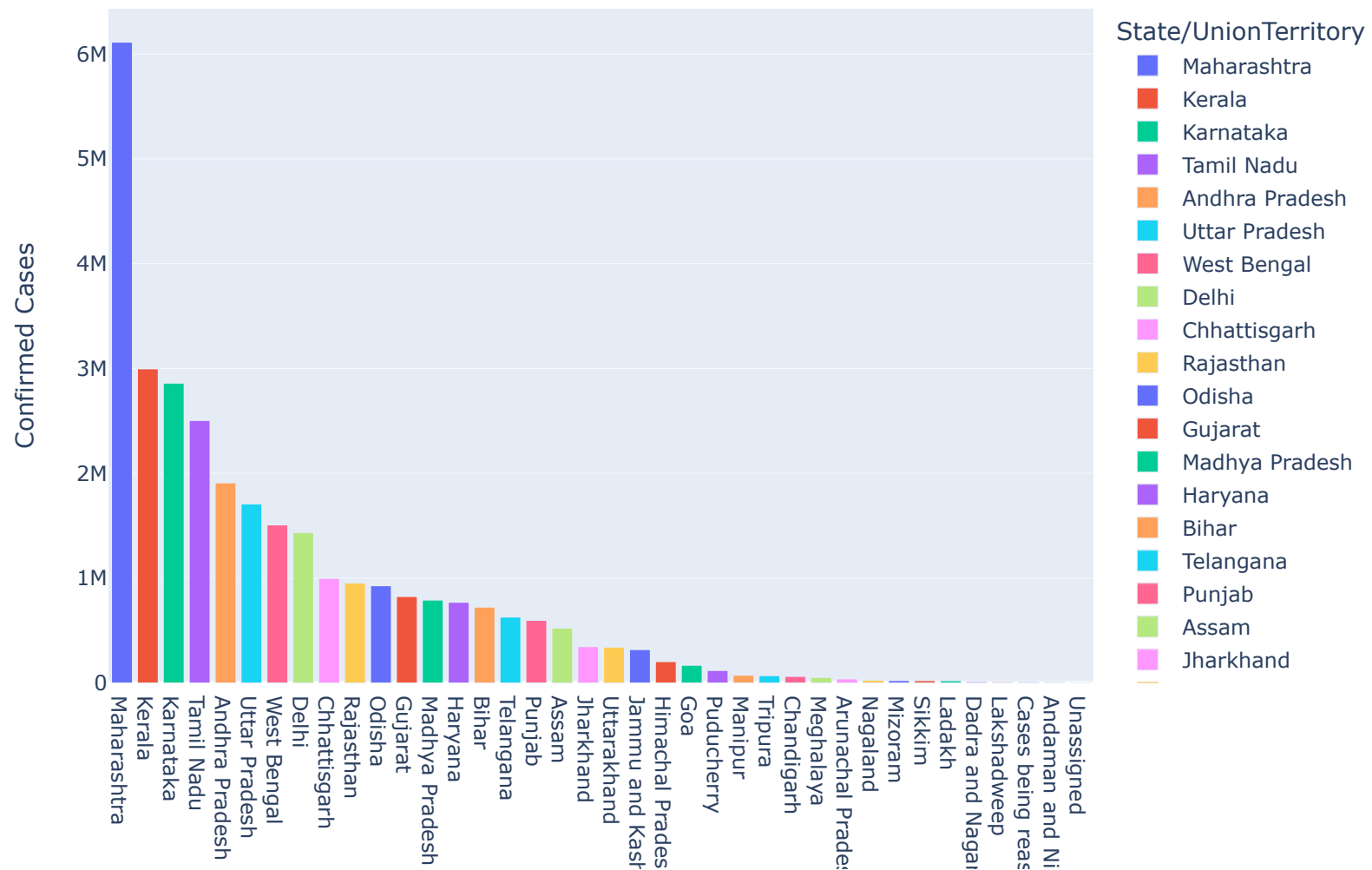
```
fig=plt.figure(figsize=(10,6))  
sns.pairplot(top_10)  
plt.show()
```

<Figure size 720x432 with 0 Axes>




```
In [33]: px.bar(allstates_max,x='State/UnionTerritory', y='Confirmed',
               title="India State Wise Confirmed Cases",
               labels={'Confirmed':'Confirmed Cases'},
               color='State/UnionTerritory',
               height=800)
```

India State Wise Confirmed Cases



cobar Islands
signed to states

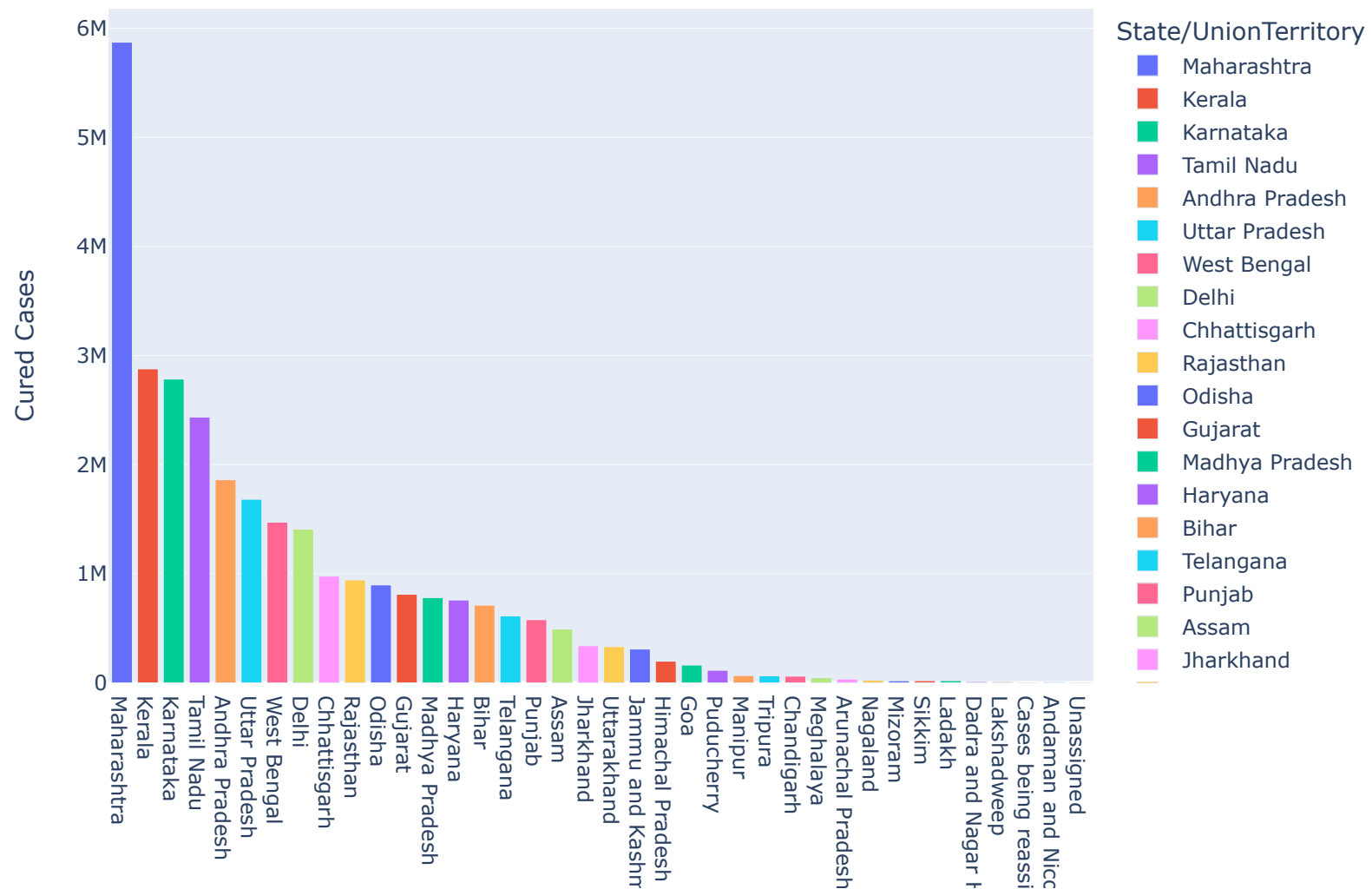
r Haveli and Daman and C

sh

h
ymir

```
In [34]: px.bar(allstates_max,x='State/UnionTerritory', y='Cured',
               title="India State Wise Cured Cases",
               labels={'Cured':'Cured Cases'},
               color='State/UnionTerritory',
               height=800)
```

India State Wise Cured Cases



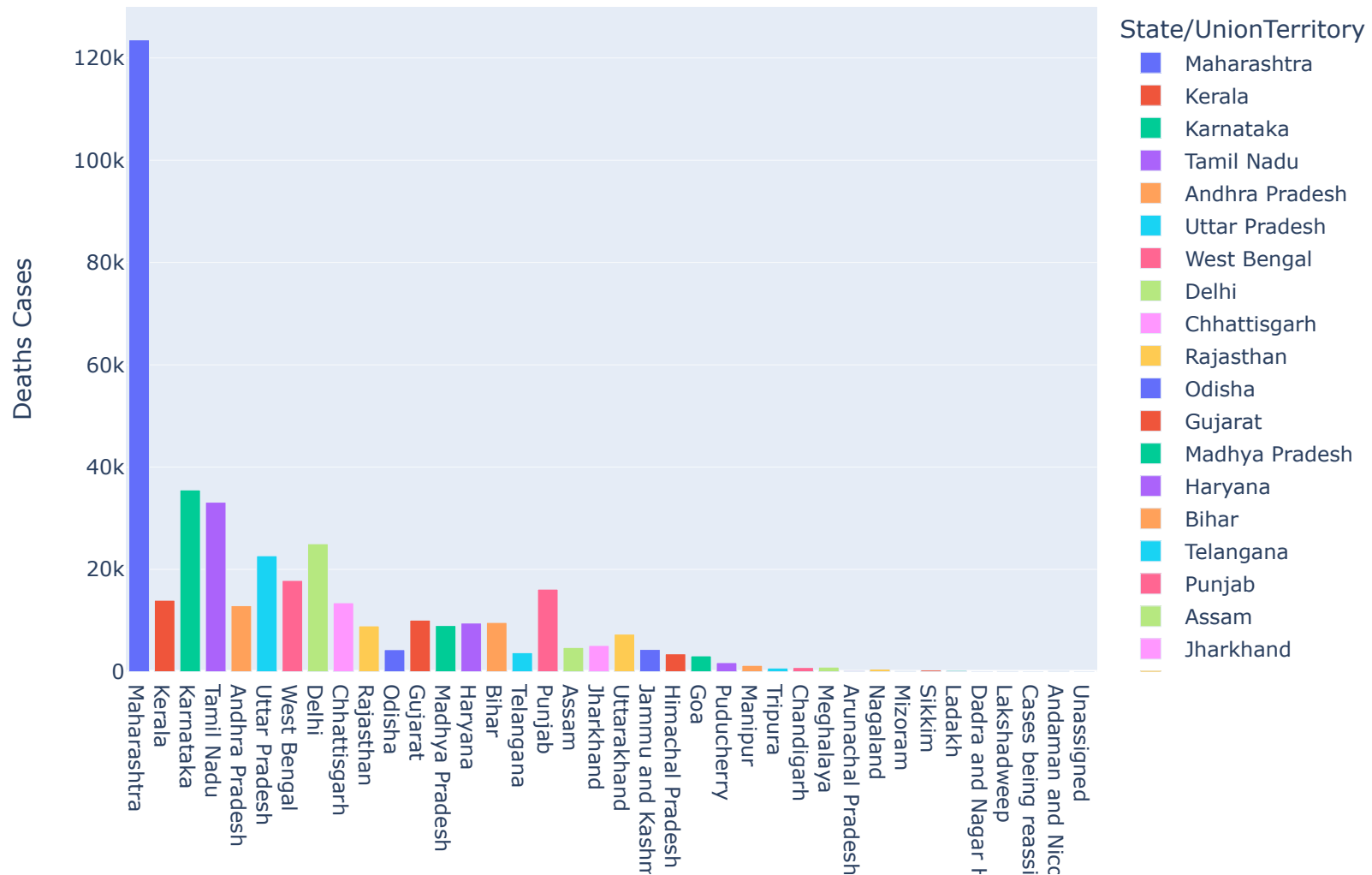
ubar Islands
igned to states

taveli and Daman and C

hir

```
In [35]: px.bar(allstates_max,x='State/UnionTerritory', y='Deaths',
               title="India State Wise Deaths Cases",
               labels={'Deaths':'Deaths Cases'},
               color='State/UnionTerritory',
               height=800)
```

India State Wise Deaths Cases



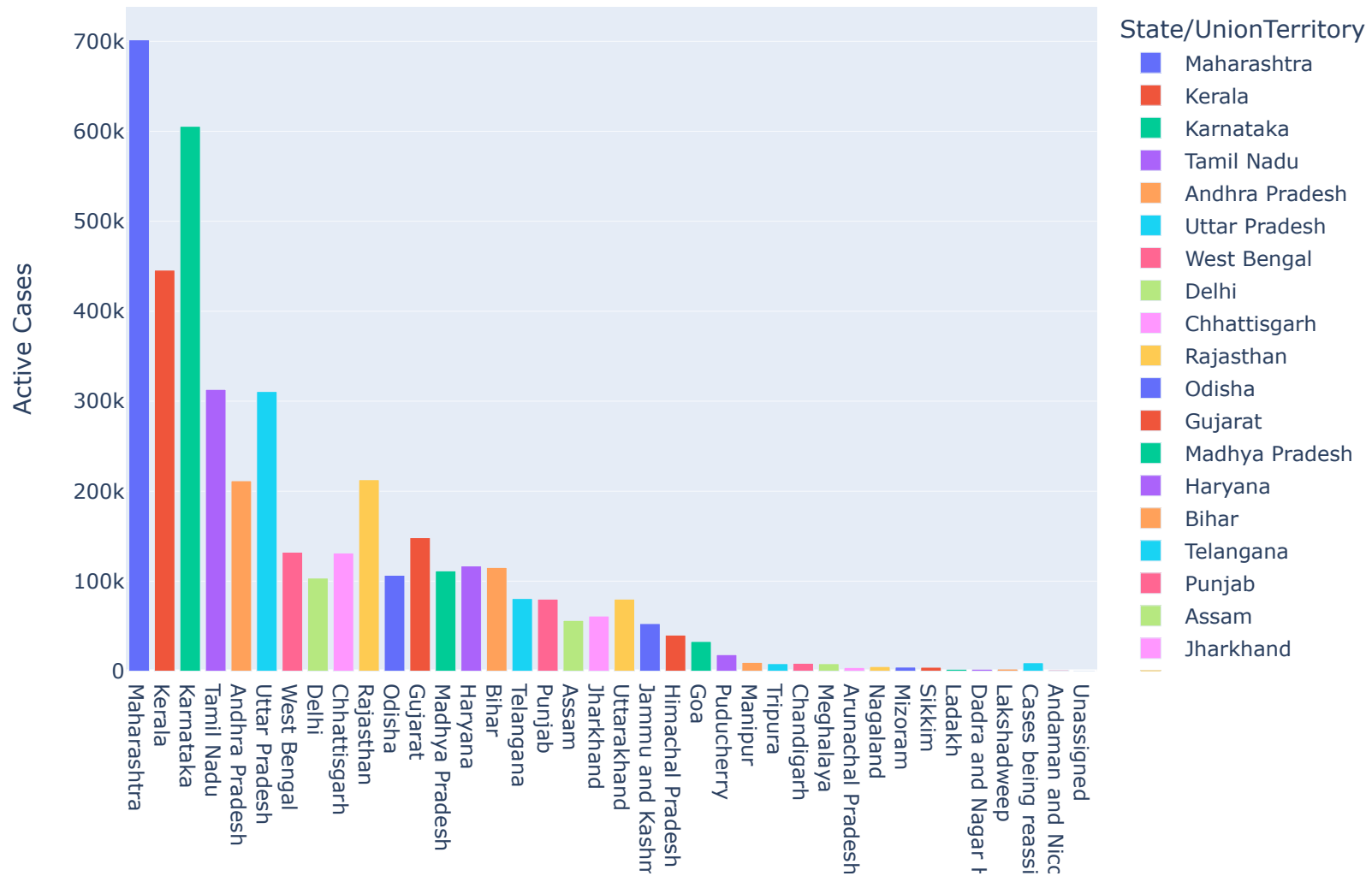
ubar Islands
igned to states

taveli and Daman and C

hir

```
In [36]: px.bar(allstates_max,x='State/UnionTerritory', y='Active',
               title="India State Wise Active Cases",
               labels={'Active':'Active Cases'},
               color='State/UnionTerritory',
               height=800)
```

India State Wise Active Cases



nir

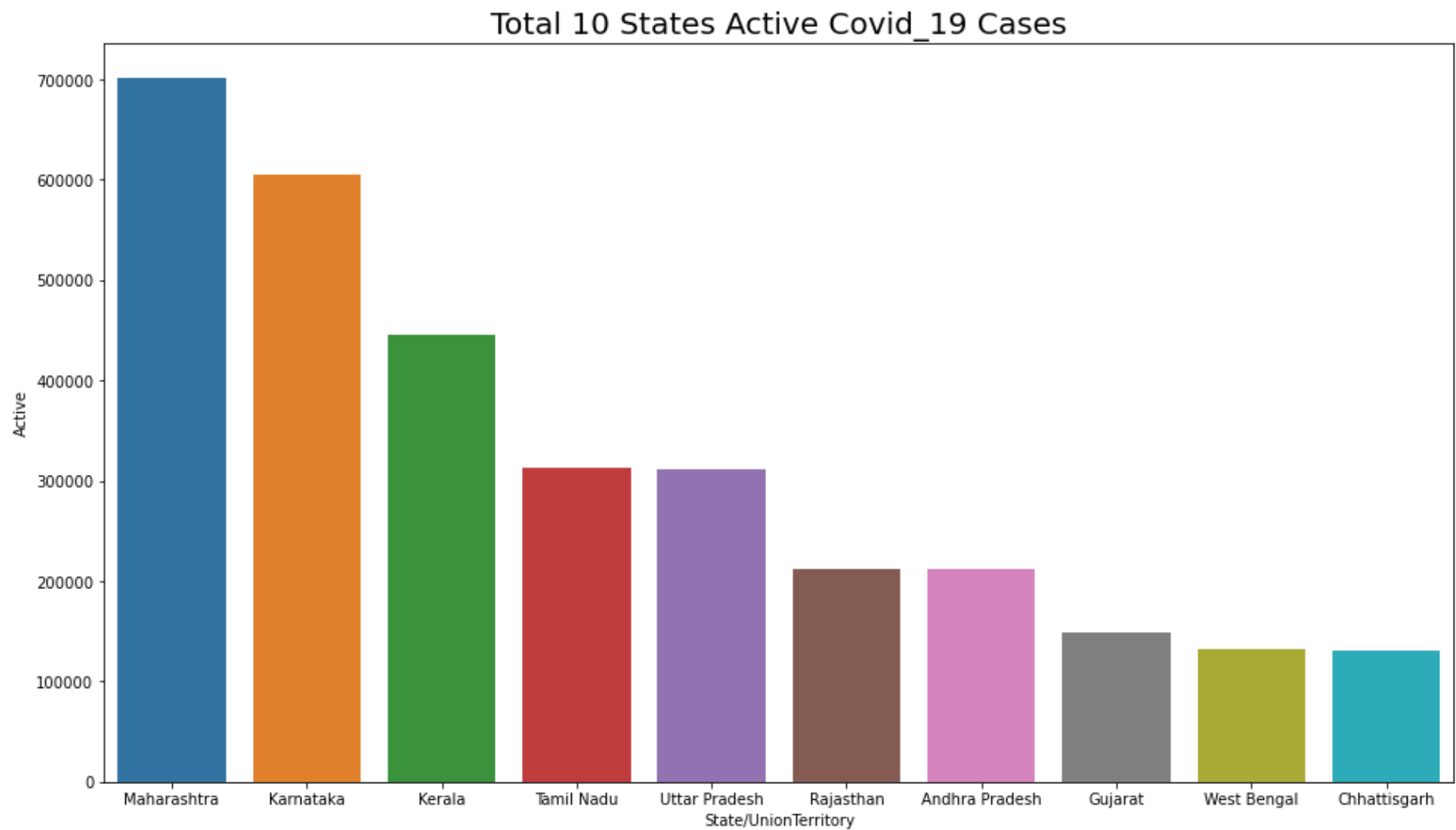
bar Islands
ned to states
aveli and Daman and C

```
In [37]: active_case = covid19.groupby('State/UnionTerritory').max()[['Active', 'Date']].sort_values(by=['Active'],ascending=False)  
active_case.head(10)
```

Out[37]:

	State/UnionTerritory	Active	Date
0	Maharashtra	701614	2021-07-07
1	Karnataka	605515	2021-07-07
2	Kerala	445692	2021-07-07
3	Tamil Nadu	313048	2021-07-07
4	Uttar Pradesh	310783	2021-07-07
5	Rajasthan	212753	2021-07-07
6	Andhra Pradesh	211554	2021-07-07
7	Gujarat	148297	2021-07-07
8	West Bengal	132181	2021-07-07
9	Chhattisgarh	131245	2021-07-07

```
In [38]: fig = plt.figure(figsize=(16,9))
sns.barplot(data=active_case.head(10),x='State/UnionTerritory',y='Active')
plt.title("Total 10 States Active Covid_19 Cases",size=20)
plt.show()
```



In []:

```
In [39]: cured_rate = (total_cured_sum/total_confirmed_sum)*100  
cured_rate
```

Out[39]: 91.35670533757235

```
In [40]: death_rate = (total_deaths_sum/total_confirmed_sum)*100  
death_rate
```

Out[40]: 1.3489444302183478

```
In [41]: active_rate = (total_active_cases_sum/total_confirmed_sum)*100  
active_rate
```

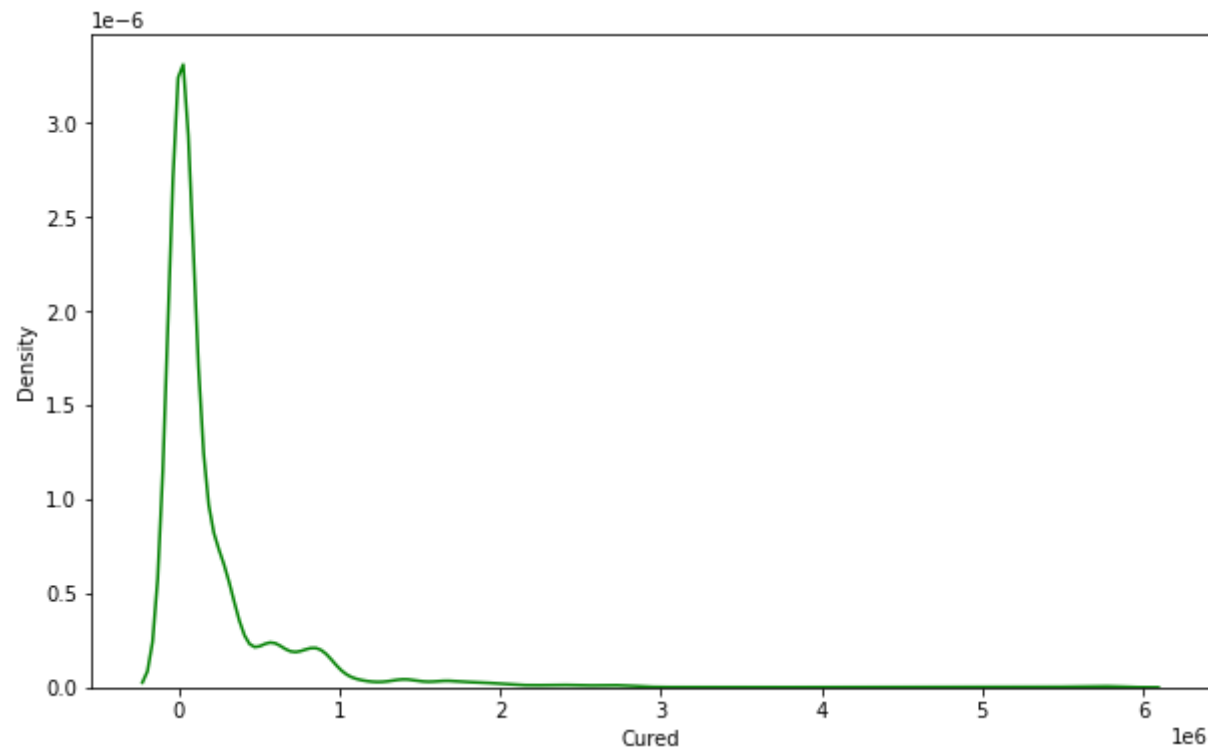
Out[41]: 7.294350232209301

In []:

```
In [42]: fig = plt.figure(figsize=(10,6))
sns.distplot(covid19['Cured'],kde=True,hist=False,color='green')
plt.show()
```

C:\Users\Yash\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning:

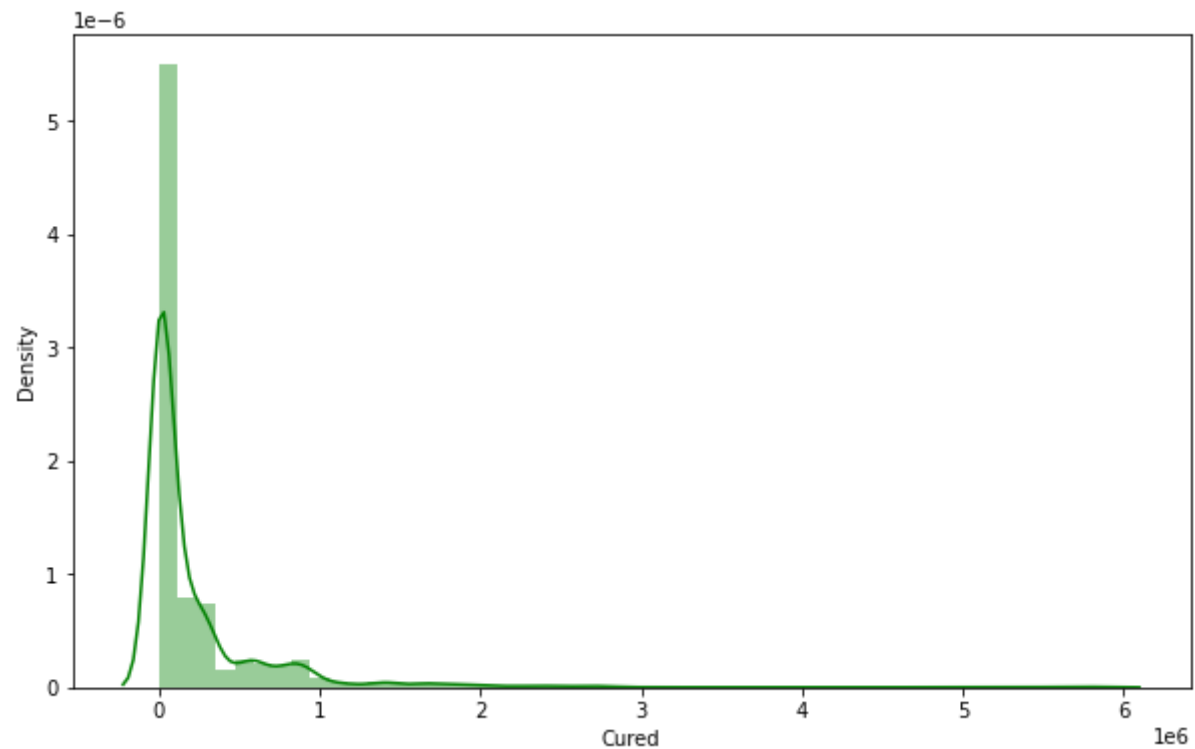
`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `kdeplot` (an axes-level function for kernel density plots).




```
In [43]: fig = plt.figure(figsize=(10,6))
sns.distplot(covid19['Cured'],kde=True,color='green')
plt.show()
```

C:\Users\Yash\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning:

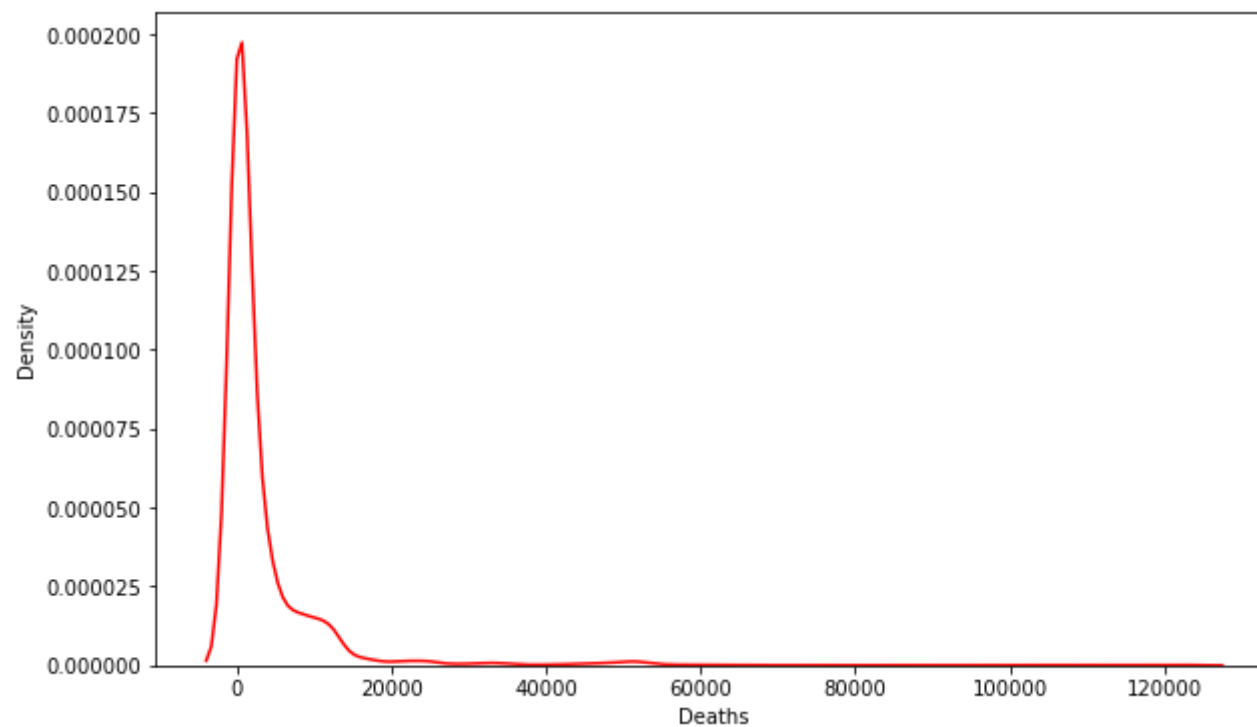
`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).



```
In [44]: fig = plt.figure(figsize=(10,6))
sns.distplot(covid19['Deaths'],kde=True,hist=False,color='red')
plt.show()
```

C:\Users\Yash\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning:

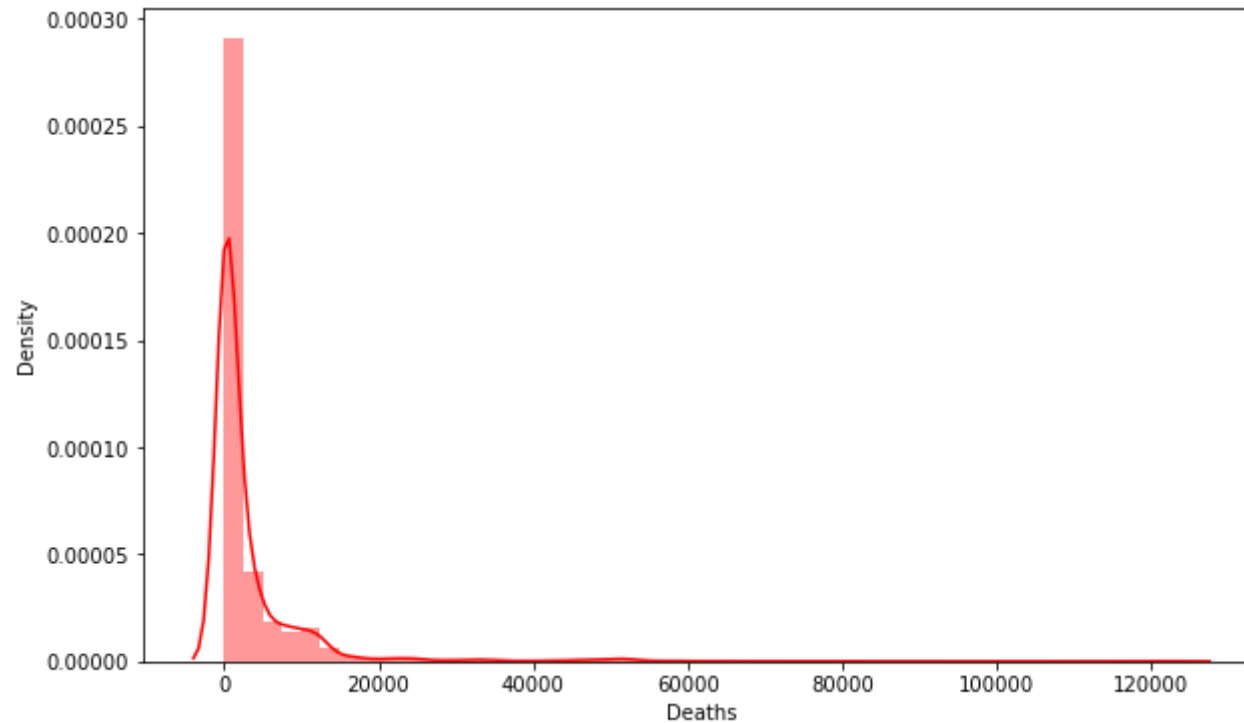
`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `kdeplot` (an axes-level function for kernel density plots).



```
In [45]: fig = plt.figure(figsize=(10,6))  
sns.distplot(covid19['Deaths'],kde=True,color='red')  
plt.show()
```

C:\Users\Yash\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning:

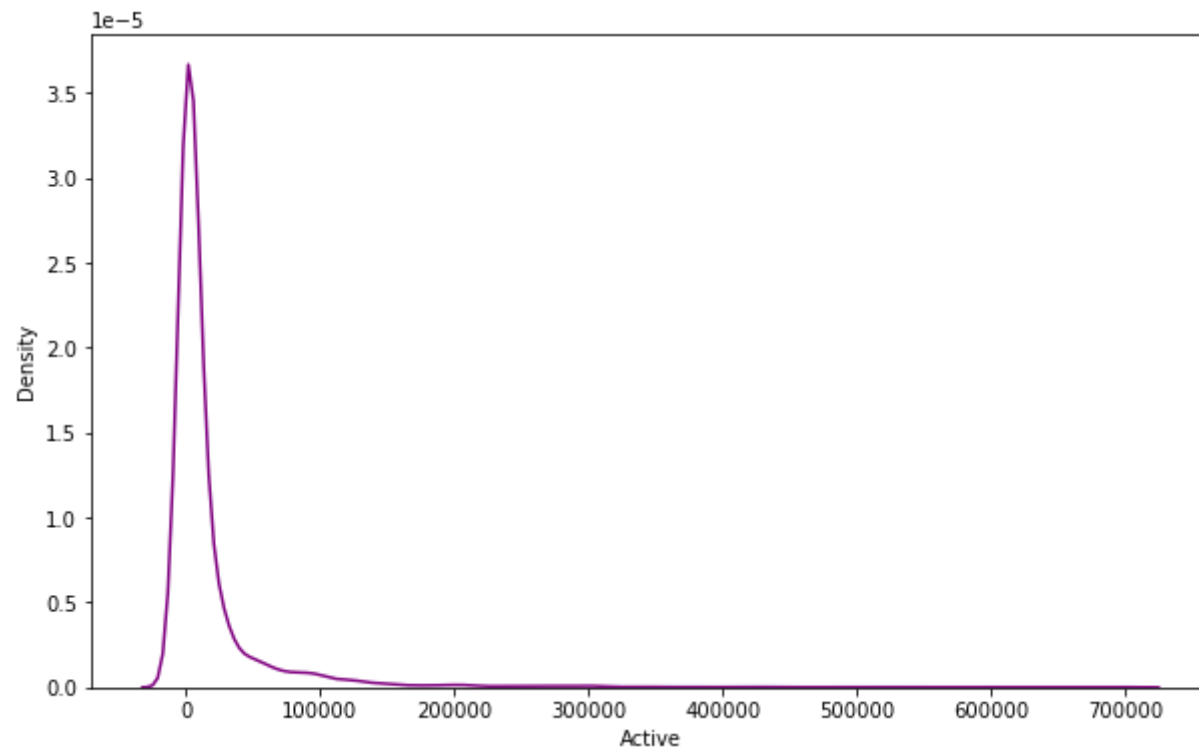
`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).



```
In [46]: fig = plt.figure(figsize=(10,6))
sns.distplot(covid19['Active'],kde=True,hist=False,color='purple')
plt.show()
```

C:\Users\Yash\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning:

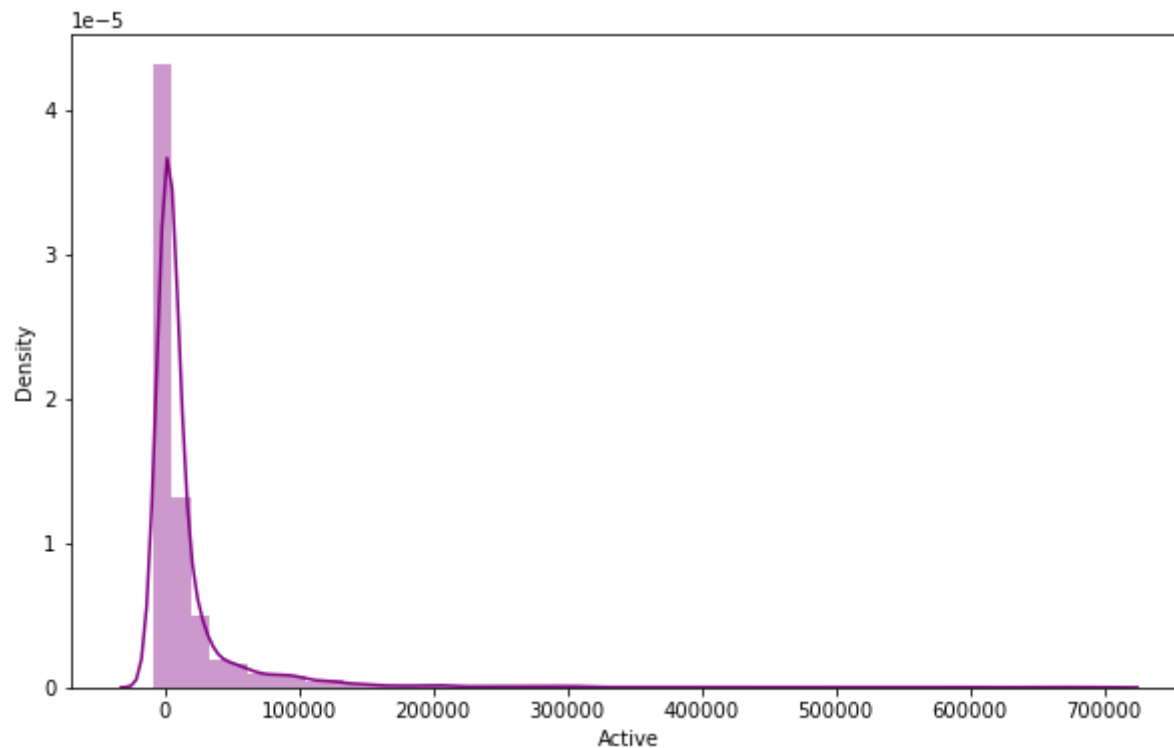
`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `kdeplot` (an axes-level function for kernel density plots).



```
In [47]: fig = plt.figure(figsize=(10,6))
sns.distplot(covid19['Active'],kde=True,color='purple')
plt.show()
```

C:\Users\Yash\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning:

`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).



In []:

Analyzing Date Wise India Covid-19 Data

```
In [48]: covid19["Date"] = pd.to_datetime(covid19["Date"], format = "%Y-%m-%d")
covid19['Year'] = pd.DatetimeIndex(covid19['Date']).year
covid19['Month'] = pd.DatetimeIndex(covid19['Date']).month
covid19['Day'] = pd.DatetimeIndex(covid19['Date']).day
```

```
In [49]: covid19['Year'].unique()
```

```
Out[49]: array([2020, 2021], dtype=int64)
```

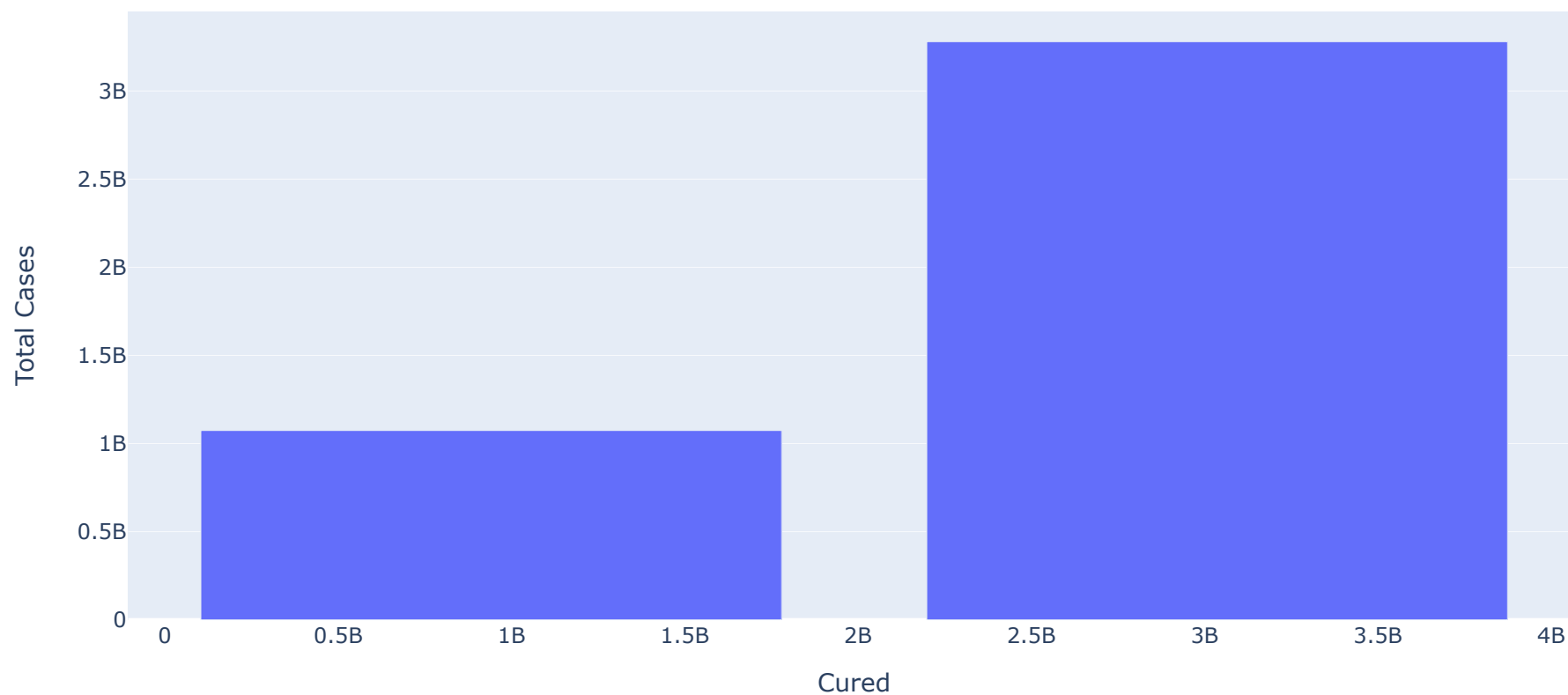
```
In [50]: yearly_data=covid19.groupby('Year')[['Cured','Deaths','Confirmed']].sum()
yearly_data
```

```
Out[50]:
```

	Cured	Deaths	Confirmed
Year			
2020	941314195	17022508	1074022781
2021	3035879941	41703492	3279455293

```
In [51]: px.bar(yearly_data, x='Cured', y='Confirmed',  
               hover_name='Deaths',  
               labels={'Confirmed': 'Total Cases'},  
               title="India State Wise Cases 2020 Vs 2021"  
               )
```

India State Wise Cases 2020 Vs 2021




```
In [52]: allstates_yearly_data=covid19.groupby(['State/UnionTerritory', 'Year'])[['Cured', 'Deaths', 'Confirmed']].sum()  
allstates_yearly_data
```

Out[52]:

		Cured	Deaths	Confirmed
State/UnionTerritory	Year			
Andaman and Nicobar Islands	2020	534731	7772	590838
	2021	1055204	14852	1084410
Andhra Pradesh	2020	95094768	871178	104134066
	2021	208333131	1604638	220012717
Arunachal Pradesh	2020	1442769	4488	1679508
...
Uttar Pradesh	2021	177050153	2411300	189954972
Uttarakhand	2020	6864570	121701	7960257
	2021	29819818	606811	33219139
West Bengal	2020	44508487	941478	50095209
	2021	150788352	2273362	159727639

74 rows × 5 columns

```
In [53]: allstates_monthly_data=covid19.groupby(['State/UnionTerritory', 'Month'])[['Cured', 'Deaths', 'Confirmed']].sum()  
allstates_monthly_data
```

Out[53]:

		Cured	Deaths	Confirmed
State/UnionTerritory	Month			
Andaman and Nicobar Islands	1	151473	1922	154187
	2	138309	1736	140209
	3	153888	1922	156090
	4	155615	1905	160840
	5	194448	2728	204112
...
West Bengal	8	2668216	73557	3537564
	9	5379943	121453	6232914
	10	8463405	181984	9643839
	11	11728981	228800	12862215
	12	15314155	282550	16211044

433 rows × 3 columns

In []:

In []:

In []:

In []:

India Covid-19 Data of Maharashtra State

```
In [54]: maha= covid19[covid19["State/UnionTerritory"] == 'Maharashtra']  
maha
```

Out[54]:

	Date	Time	State/UnionTerritory	Cured	Deaths	Confirmed	Active	Year	Month	Day
76	2020-03-09	6:00 PM	Maharashtra	0	0	2	2	2020	3	9
91	2020-03-10	6:00 PM	Maharashtra	0	0	5	5	2020	3	10
97	2020-03-11	6:00 PM	Maharashtra	0	0	2	2	2020	3	11
120	2020-03-12	6:00 PM	Maharashtra	0	0	11	11	2020	3	12
133	2020-03-13	6:00 PM	Maharashtra	0	0	14	14	2020	3	13
...
16690	2021-07-03	8:00 AM	Maharashtra	5836920	122353	6079352	120079	2021	7	3
16726	2021-07-04	8:00 AM	Maharashtra	5845315	122724	6088841	120802	2021	7	4
16762	2021-07-05	8:00 AM	Maharashtra	5848693	123030	6098177	126454	2021	7	5
16798	2021-07-06	8:00 AM	Maharashtra	5861720	123136	6104917	120061	2021	7	6
16834	2021-07-07	8:00 AM	Maharashtra	5872268	123531	6113335	117536	2021	7	7

486 rows × 10 columns

In [55]: maha.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 486 entries, 76 to 16834
Data columns (total 10 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Date                  486 non-null   datetime64[ns]
1   Time                  486 non-null   object
2   State/UnionTerritory  486 non-null   object
3   Cured                 486 non-null   int64
4   Deaths               486 non-null   int64
5   Confirmed             486 non-null   int64
6   Active               486 non-null   int64
7   Year                  486 non-null   int64
8   Month                 486 non-null   int64
9   Day                   486 non-null   int64
dtypes: datetime64[ns](1), int64(7), object(2)
memory usage: 41.8+ KB
```

In [56]: maha.describe()

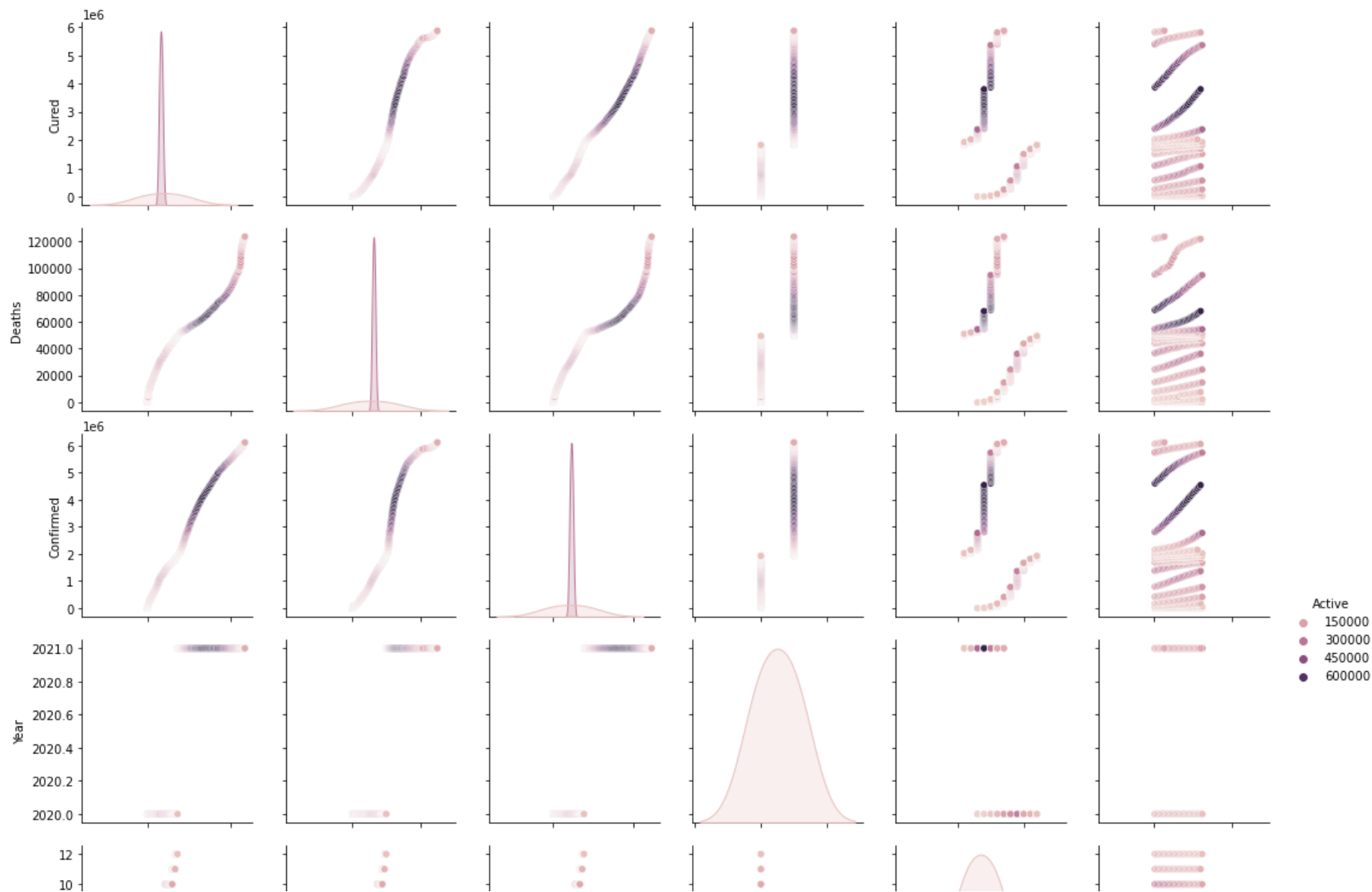
Out[56]:

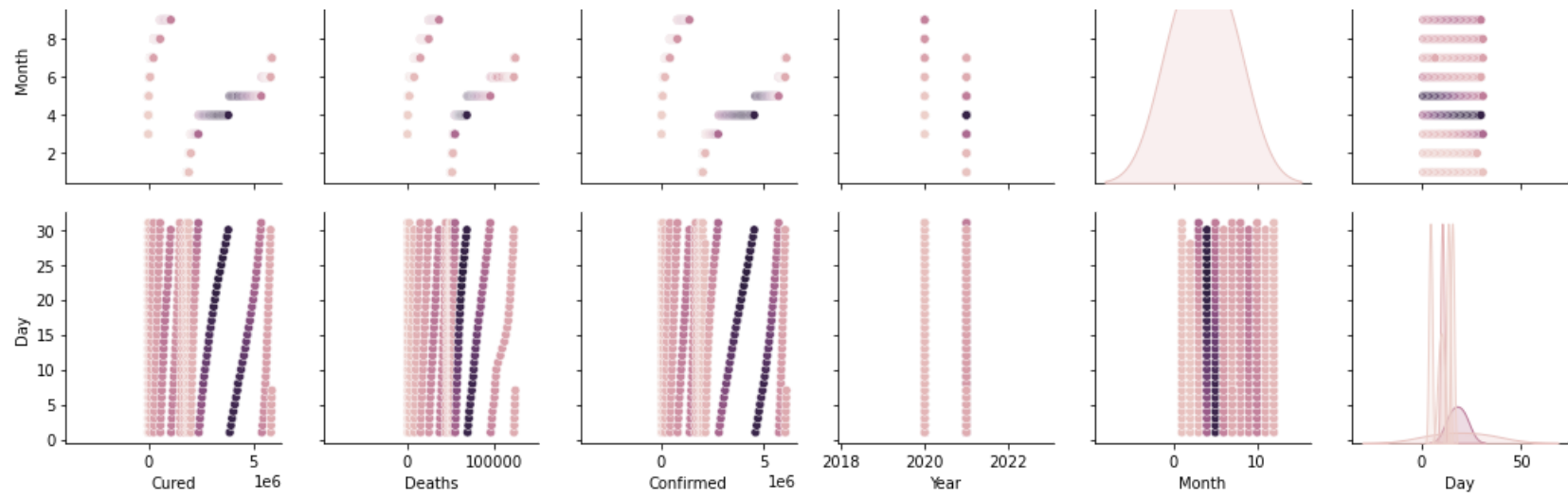
	Cured	Deaths	Confirmed	Active	Year	Month	Day
count	4.860000e+02	486.000000	4.860000e+02	486.000000	486.000000	486.000000	486.000000
mean	1.674463e+06	39741.835391	1.870149e+06	155944.508230	2020.386831	6.080247	15.744856
std	1.710989e+06	31861.231600	1.831266e+06	168833.300211	0.487526	3.146548	8.810065
min	0.000000e+00	0.000000	2.000000e+00	2.000000	2020.000000	1.000000	1.000000
25%	1.197165e+05	9299.500000	2.187718e+05	46101.000000	2020.000000	4.000000	8.000000
50%	1.556812e+06	44884.500000	1.706879e+06	96492.500000	2020.000000	6.000000	16.000000
75%	2.066541e+06	52468.500000	2.216942e+06	193650.750000	2021.000000	8.750000	23.000000
max	5.872268e+06	123531.000000	6.113335e+06	701614.000000	2021.000000	12.000000	31.000000

In []:

```
In [57]: fig = plt.figure(figsize=(10,6))
sns.pairplot(maha,hue='Active')
plt.show()
```

<Figure size 720x432 with 0 Axes>

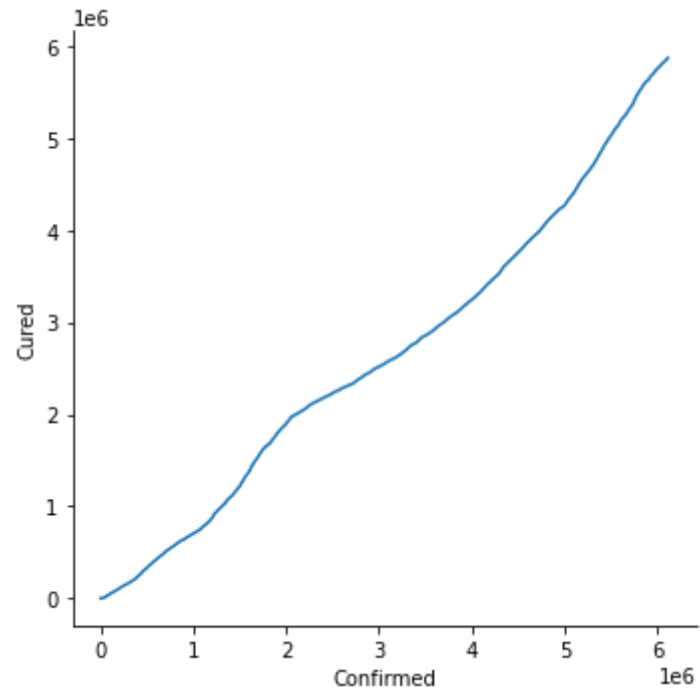




In []:

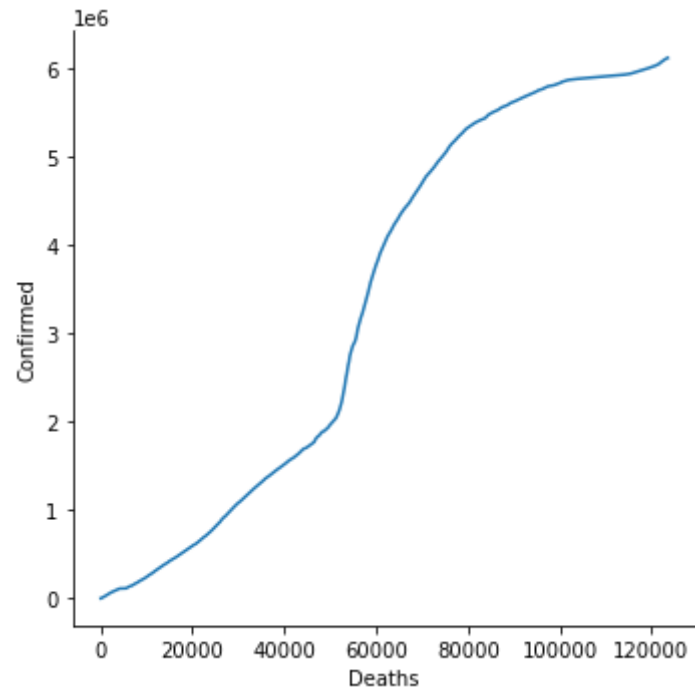
```
In [58]: fig = plt.figure(figsize=(10,6))  
sns.relplot(x='Confirmed',y='Cured',data=maha,kind='line')  
plt.show()
```

<Figure size 720x432 with 0 Axes>

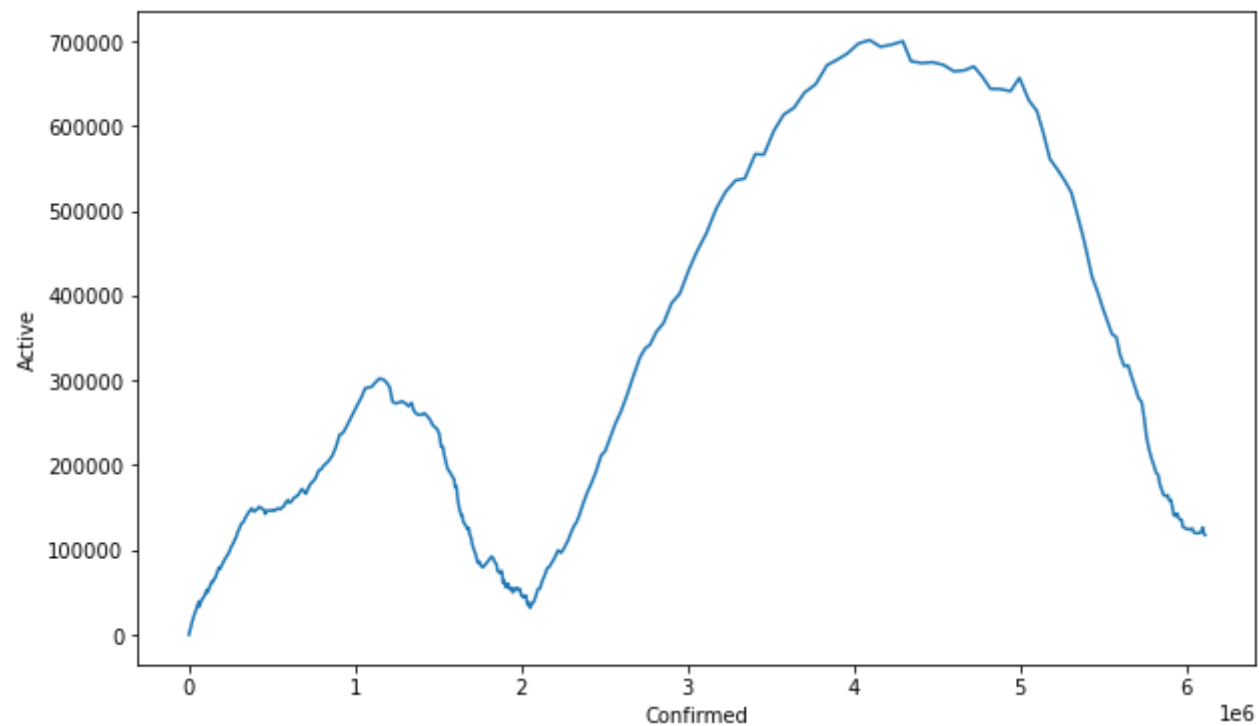


```
In [59]: fig = plt.figure(figsize=(10,6))  
sns.relplot(x='Deaths',y='Confirmed',data=maha,kind='line')  
plt.show()
```

<Figure size 720x432 with 0 Axes>



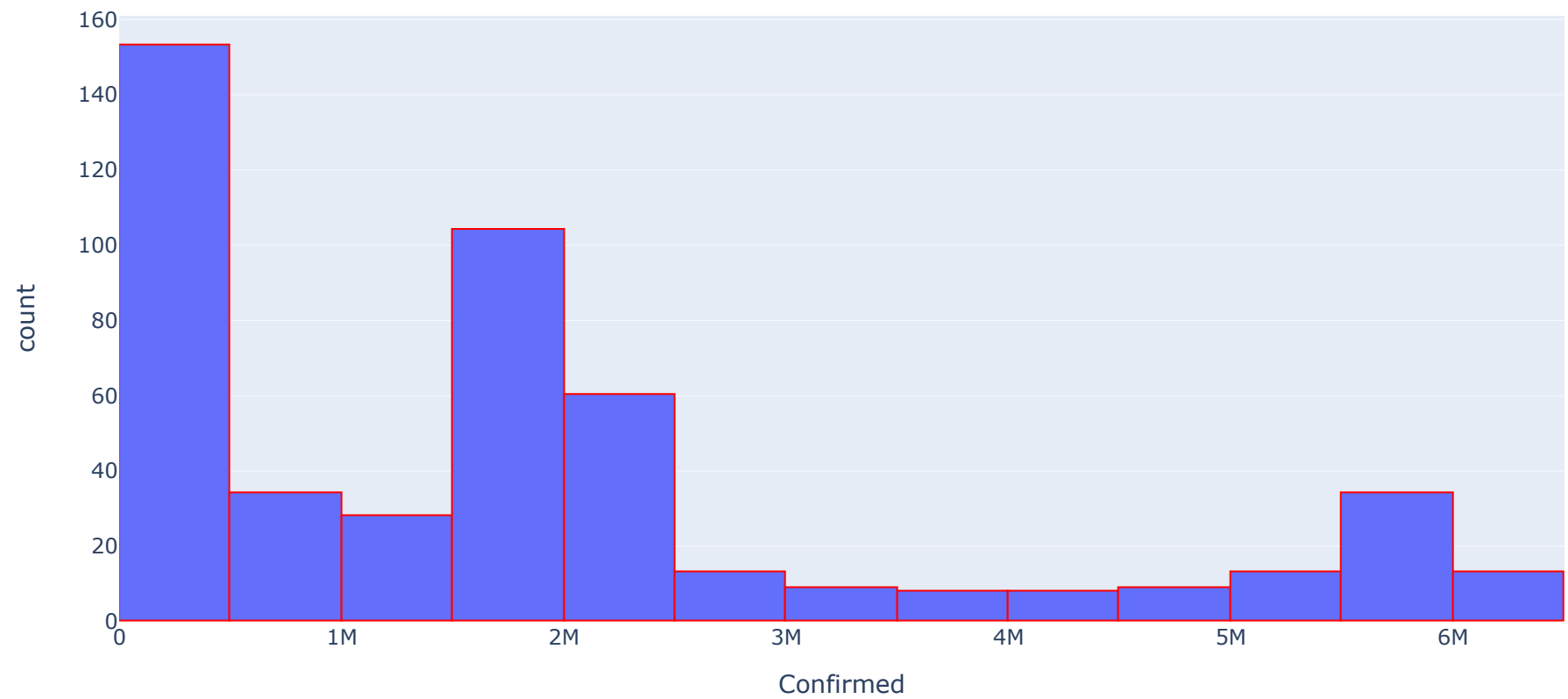

```
In [60]: fig = plt.figure(figsize=(10,6))  
sns.lineplot(y='Active',x='Confirmed',data=maha)  
plt.show()
```



```
In [ ]:
```

```
In [61]: fig=px.histogram(x='Confirmed',data_frame=maha,title='Total Confirmed Count of Maharashtra')  
fig.update_traces(marker_line_width=1,marker_line_color='red')
```

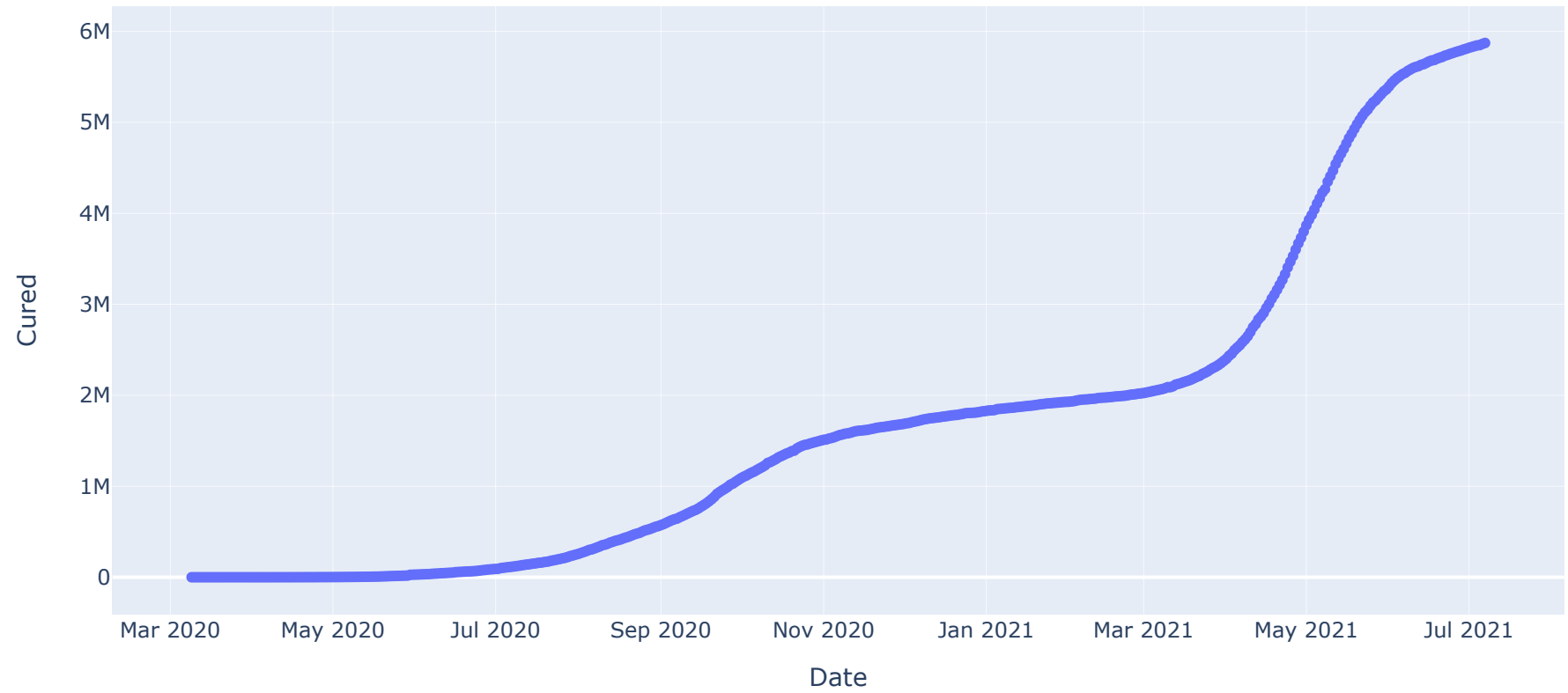
Total Confirmed Count of Maharashtra



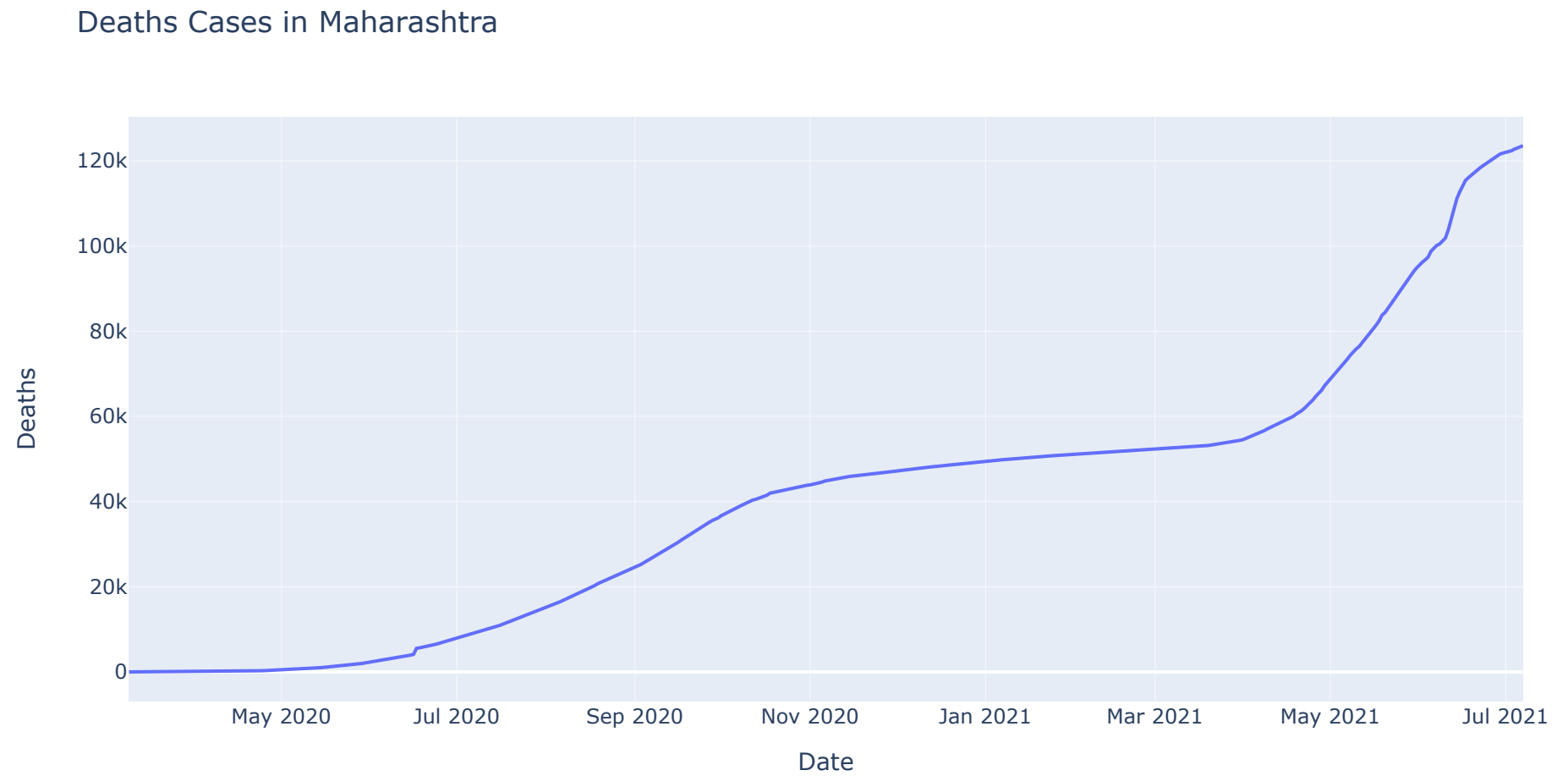
In [62]:

```
fig = px.scatter(data_frame=maha, x="Date", y="Cured", title='Cured Cases in Maharashtra')  
fig.show()
```

Cured Cases in Maharashtra

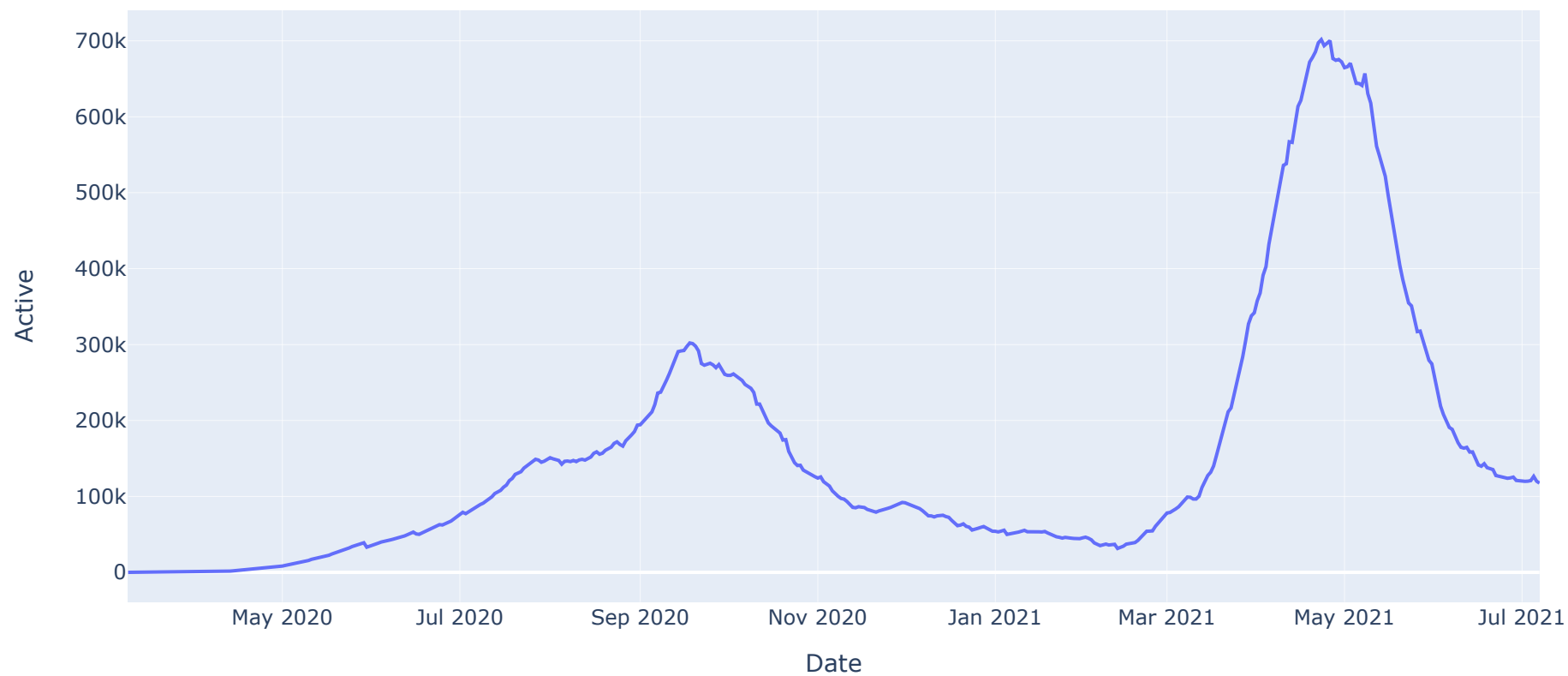


```
In [63]: fig = px.line(data_frame=maha, x="Date", y="Deaths", title='Deaths Cases in Maharashtra')  
fig.show()
```



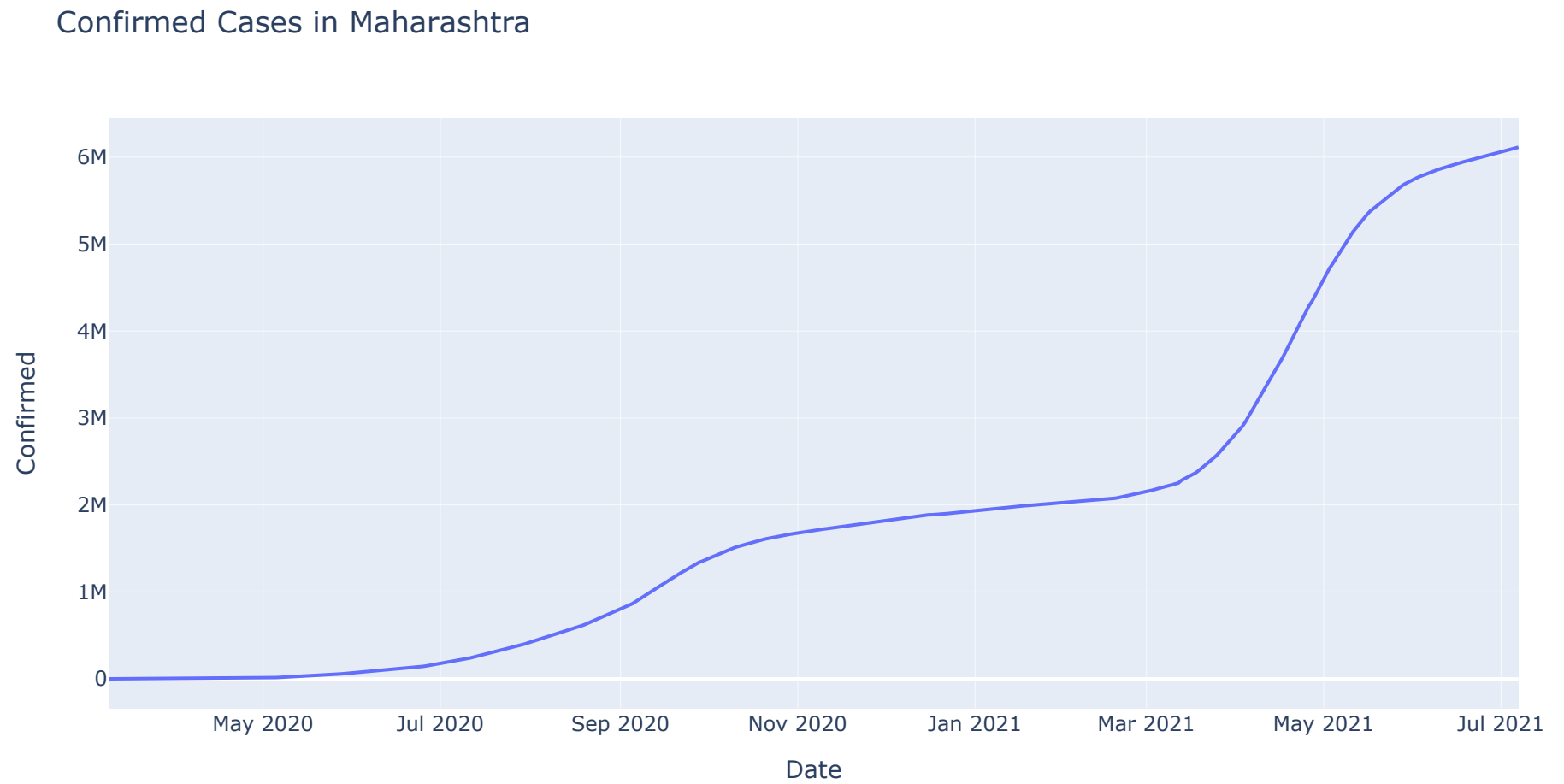
```
In [64]: fig = px.line(data_frame=maha, x="Date", y="Active", title='Active Cases in Maharashtra')  
fig.show()
```

Active Cases in Maharashtra



In []:

```
In [65]: fig = px.line(data_frame=maha, x="Date", y="Confirmed", title='Confirmed Cases in Maharashtra')  
fig.show()
```



In []:

```
In [66]: maha_cured_sum = maha['Cured'].sum()  
maha_cured_sum
```

```
Out[66]: 813788907
```

```
In [67]: maha_deaths_sum = maha['Deaths'].sum()  
maha_deaths_sum
```

```
Out[67]: 19314532
```

```
In [68]: maha_confirmed_sum = maha['Confirmed'].sum()  
maha_confirmed_sum
```

```
Out[68]: 908892470
```

```
In [69]: maha_active_sum = maha['Active'].sum()  
maha_active_sum
```

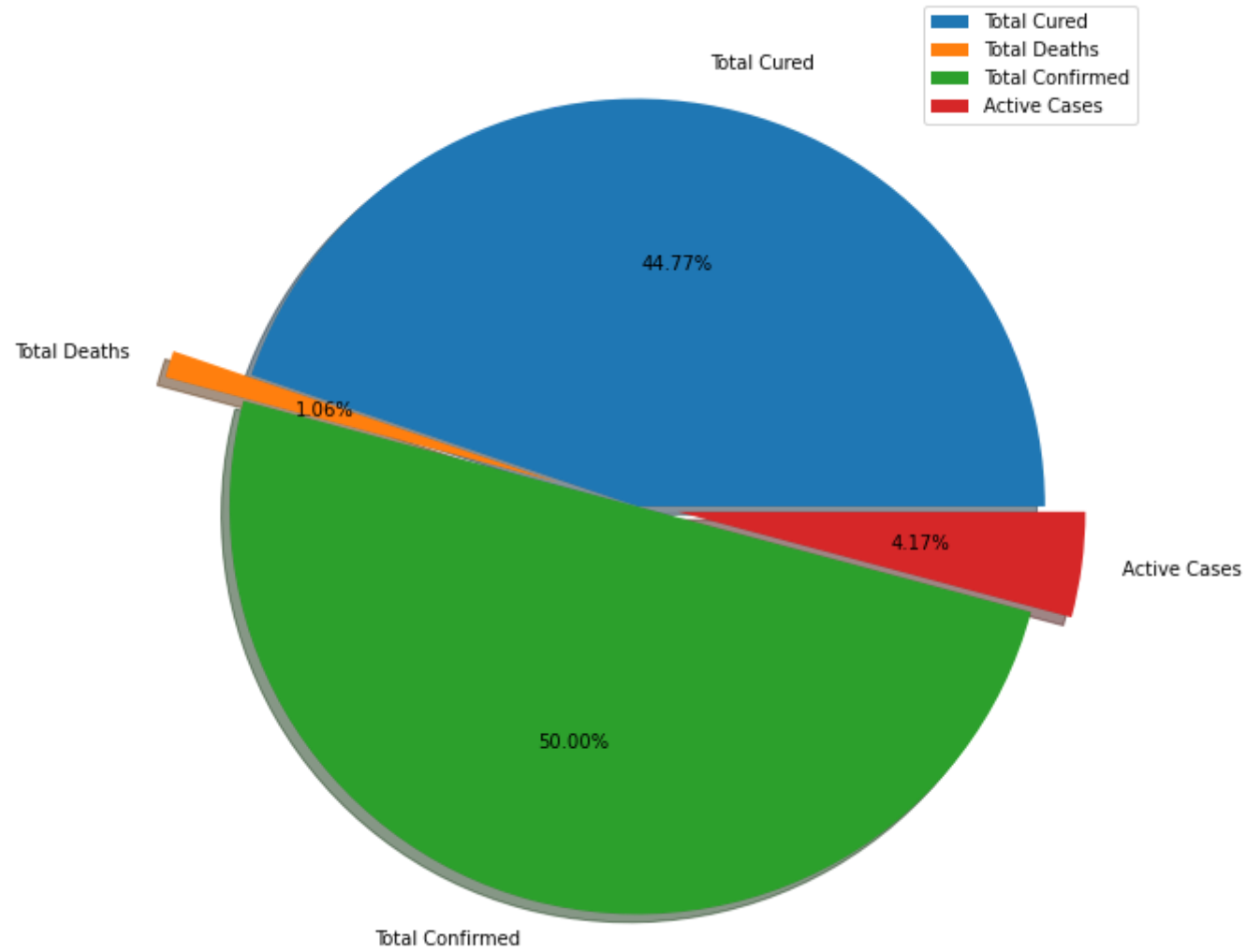
```
Out[69]: 75789031
```

```
In [70]: result=np.array([maha_cured_sum,maha_deaths_sum,maha_confirmed_sum,maha_active_sum])  
lable = ["Total Cured","Total Deaths","Total Confirmed","Active Cases"]  
e = [0.0,0.2,0.0,0.1]
```

```
In [71]: fig = plt.figure()
fig.set_figheight(10)
fig.set_figwidth(12)

plt.pie(result, labels=lable, autopct='%1.2f%%', explode=e, shadow=True)
plt.title("Maharashtra COVID-19 Data", size=20, color='k')
plt.legend(lable)
plt.show()
```


Maharashtra COVID-19 Data



In []:

In []:

Date Wise Covid-19 Data of Maharashtra

```
In [72]: maha['Year'].unique()
```

```
Out[72]: array([2020, 2021], dtype=int64)
```

```
In [73]: maha_yearly_data=maha.groupby('Year')[['Cured', 'Deaths', 'Confirmed']].sum()  
maha_yearly_data
```

```
Out[73]:
```

	Cured	Deaths	Confirmed
Year			
2020	187034270	6184938	222900632
2021	626754637	13129594	685991838

```
In [ ]:
```

```
In [74]: maha_monthly_data=maha.groupby('Month')[['Cured','Deaths','Confirmed']].sum()  
maha_monthly_data
```

Out[74]:

	Cured	Deaths	Confirmed
Month			
1	58313365	1559536	61433195
2	55303793	1442941	57992941
3	67054190	1644594	73874996
4	89859366	1800692	109201594
5	146246891	2554642	164573396
6	171002307	3450855	180776368
7	45819418	1201894	51391124
8	12695192	609995	18227851
9	23832012	908307	32562680
10	40815564	1269079	48150932
11	48214136	1372935	52401197
12	54632673	1499062	58306196

In []:

```
In [75]: maha_daily_data=maha.groupby('Day')[['Cured','Deaths','Confirmed']].sum()  
maha_daily_data
```

Out[75]:

	Cured	Deaths	Confirmed
Day			
1	28502795	671418	31691952
2	28705621	675239	31894681
3	28891775	678691	32095468
4	29099470	682181	32295652
5	29300242	686996	32506537
6	29483671	690680	32709622
7	29677537	694557	32933806
8	23967103	574368	27020193
9	24178446	577963	27224924
10	24377326	581289	27426078
11	24575007	585843	27602865
12	24762640	591370	27804743
13	24972625	596086	28013925
14	25128349	601768	28212428
15	25309260	605938	28403236
16	25510779	610594	28590275
17	25696275	616353	28784636
18	25881287	620316	28979438
19	26074231	624381	29182733
20	26273293	627784	29383810
21	26467406	631671	29589229
22	26685958	635536	29794745

	Cured	Deaths	Confirmed
Day			
23	26887754	638880	29995950
24	27086246	643384	30200531
25	27281221	646918	30405658
26	27472257	651119	30613288
27	27657098	654846	30809396
28	27839463	658807	31018350
29	26005105	610463	29063743
30	26204715	614151	29256181
31	13833952	334942	15388397

In []:

In []: