Registration Number: 21MDT0068

Name- Arnab Dey

Abstract

COVID-19 pandemic represents an unprecedented global health crisis in the last 100 years. Its economic, social and health impact continues to grow and is likely to end up as one of the worst global disasters since the 1918 pandemic and the World Wars. Mathematical models have played an important role in the ongoing crisis; they have been used to inform public policies and have been instrumental in many of the social distancing measures that were instituted worldwide. In this work, I performed analysis on latest Indian state wise covid-1 data of using important Exploratory Data Analysis techniques to find important findings and arrive at conclusions based on them, so that we can make important decisions.

Introduction:

The ongoing COVID-19 pandemic is the most significant pandemic since the 1918 Influenza pandemic. It has already caused over 21 Million confirmed cases and 758,000 deaths.[1] The economic impact is already in trillions of dollars. As in other pandemics, researchers and public health policy makers are interested in questions such as,[2] (i) How did it start? (ii) How is it likely to progress and how can we control it? (iii) Which states having large population handled their situation well? (iv) How related active cases and deaths in a state were? (v) Which states had largest death cases?

To answer these questions we leverage EDA which as defined by Tukey in 1961 as:

"Procedures for analyzing data, techniques for interpreting the results of such procedures, ways of planning the gathering of data to make its analysis easier, more precise or more accurate, and all the machinery and results of (mathematical) statistics which apply to analyzing data."[

We perform EDA on a dataset containing current data on covid-19 from https://www.mohfw.gov.in/ and uncover hidden insights regarding the spread of the disease, through extensive studies and visualizations.

Preliminaries:

We have considered the dataset on current covid-19 scenario as of May 22, 2022 taken from https://www.mohfw.gov.in/ which shows statewise covid-19 data .

The structure of the dataset is as follows:

	State/UTs	Total Cases	Active	Discharged	Deaths	Active Ratio	Discharge Ratio	Death Ratio	Population
0	Andaman and Nicobar	8397	479	7789	129	5.70	92.76	1.54	399001
1	Andhra Pradesh	2087879	10119	2063255	14505	0.48	98.82	0.69	91702478
2	Arunachal Pradesh	56010	665	55063	282	1.19	98.31	0.50	1711947
3	Assam	635050	13139	615722	6189	2.07	96.96	0.97	35998752
4	Bihar	762458	28660	721684	12114	3.76	94.65	1.59	128500364

consisting of the first 5 rows.

In our analysis we have given primary focus on visualisations and correlations between the features.

For visualisation purposes, we used plots as:-

(i) Bar Plots:

Bar graphs are the pictorial representation of data (generally grouped), in the form of vertical or horizontal rectangular bars, where the length of bars is proportional to the measure of data.

(ii) Scatter plots

Scatter plots are the graphs that present the relationship between two variables in a data-set. It represents data points on a two-dimensional plane or on a **Cartesian system**. The independent variable or attribute is plotted on the X-axis, while the dependent variable is plotted on the Y-axis.

Another tool used in the analysis was **Correlation coefficient**, which helped in uncovering linear relationships between variables in the dataset.

It is usually represented by ρ (rho).

Mathematically, it is defined as:

$$\rho(X,Y) = \text{cov}(X,Y) / \sigma X.\sigma Y.$$

where cov(X,Y) gives covariance between X and Y.

Implementation:

The analysis was performed on a 64-bit machine with Windows 10 Operating System and 8 G.B of RAM.

Python language was used for the work. Modules as pandas and matplotlib were used extensively.

Pandas module was used to read the dataset in python and do various manipulations on it to prepare the data for further analysis.

Matplotlib was used for visualising the data and to present the patterns uncovered from it for making inferences regarding the patterns.

Procedure:

Firstly, the dataset was read and we scanned through the data to decide upon the variables for study as:

In [81]:
```
df=pd.read_csv("datatrue.csv")
df.head()
```

Out[81]:

	State/UTs	Total Cases	Active	Discharged	Deaths	Active Ratio	Discharge Ratio	Death Ratio	Population
0	Andaman and Nicobar	8397	479	7789	129	5.70	92.76	1.54	399001
1	Andhra Pradesh	2087879	10119	2063255	14505	0.48	98.82	0.69	91702478
2	Arunachal Pradesh	56010	665	55063	282	1.19	98.31	0.50	1711947
3	Assam	635050	13139	615722	6189	2.07	96.96	0.97	35998752
4	Bihar	762458	28660	721684	12114	3.76	94.65	1.59	128500364

In [82]:
```
for i in df.columns:
    print(i)
```

```
State/UTs
Total Cases
Active
Discharged
Deaths
Active Ratio
Discharge Ratio
Death Ratio
Population
```

We then observed that none of the variables had missing values, so no imputation was required.

Therefore, we could proceed and find the descriptive summary of the dataset as:

In [83]:
```
df.isnull().sum()
```

Out[83]:
```
State/UTs          0
Total Cases        0
Active             0
Discharged         0
Deaths             0
Active Ratio       0
Discharge Ratio    0
Death Ratio        0
Population         0
dtype: int64
```

In [84]:
```
df.describe()
```

Out[84]:

	Total Cases	Active	Discharged	Deaths	Active Ratio	Discharge Ratio	Death Ratio	Population
count	3.600000e+01	36.000000	3.600000e+01	36.000000	36.000000	36.000000	36.000000	3.600000e+01
mean	1.008831e+06	31042.527778	9.643156e+05	13473.194444	3.151944	95.590833	1.257222	3.971861e+07
std	1.512173e+06	47482.150114	1.448209e+06	25045.352736	1.977721	2.158088	0.536383	5.050913e+07
min	8.397000e+03	15.000000	7.789000e+03	4.000000	0.140000	90.760000	0.040000	6.600100e+04
25%	8.756775e+04	2332.000000	8.426325e+04	1021.000000	1.787500	94.617500	0.947500	1.695473e+06
50%	5.132880e+05	13636.500000	4.740580e+05	5686.500000	2.965000	95.795000	1.300000	2.410088e+07
75%	1.051432e+06	35371.500000	1.009939e+06	13846.500000	3.940000	96.930000	1.567500	6.979986e+07
max	7.034661e+06	243849.000000	6.649111e+06	141701.000000	7.910000	99.370000	2.630000	2.315026e+08

To prepare the data so that the analysis becomes easier to perform, we first sort the rows in the dataset in descending order of the population of the states.

In [85]:
```
df.sort_values(by="Population",ascending=False,inplace=True)
df.reset_index(drop=False,inplace=True)
df.head()
```

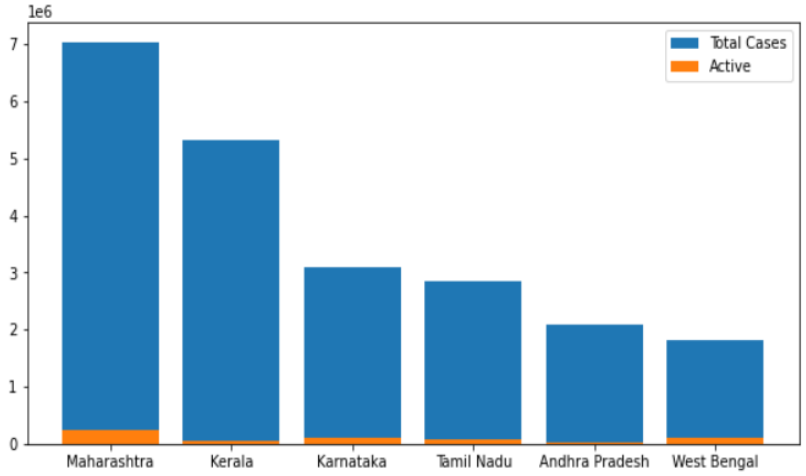
Out[85]:

	index	State/UTs	Total Cases	Active	Discharged	Deaths	Active Ratio	Discharge Ratio	Death Ratio	Population
0	33	Uttar Pradesh	1770521	57355	1690226	22940	3.24	95.46	1.30	231502578
1	4	Bihar	762458	28660	721684	12114	3.76	94.65	1.59	128500364
2	20	Maharashtra	7034661	243849	6649111	141701	3.47	94.52	2.01	124904071
3	35	West Bengal	1817585	116251	1681375	19959	6.40	92.51	1.10	100896618
4	1	Andhra Pradesh	2087879	10119	2063255	14505	0.48	98.82	0.69	91702478

Now we visualise how many active cases are there in the top five states sorted according to population to see what proportion of the population is getting exposed to the spread of the disease.

```
In [143]: plt.figure(figsize=(10,5))
          dff=df.sort_values(by="Total Cases",ascending=False)
          dff.reset_index(drop=True,inplace=True)
          plt.bar(dff.loc[:5,["State/UTs"]]["State/UTs"],dff.loc[:5,["Total Cases"]]["Total Cases"],label="Total Cases")
          plt.bar(dff.loc[:5,["State/UTs"]]["State/UTs"],dff.loc[:5,["Active"]]["Active"],label="Active")
          plt.legend()
```

```
Out[143]: <matplotlib.legend.Legend at 0x20ad1c9e5e0>
```



We observe that Out of the 6 most populous states Maharashtra has maximum covid cases as of May 22, followed by Andhra Pradesh and West Bengal, respectively.

Now we find correlations between the variables to check whether one has an influence over the other.

In [144]:
```
dff.corr()
```

Out[144]:

	Total Cases	Active	Discharged	Deaths	Active Ratio	Discharge Ratio	Death Ratio	Population
Total Cases	1.000000	0.860484	0.999776	0.935526	-0.025761	-0.000841	0.100452	0.525734
Active	0.860484	1.000000	0.849776	0.920939	0.249863	-0.279725	0.206191	0.566020
Discharged	0.999776	0.849776	1.000000	0.929357	-0.035694	0.009900	0.093860	0.522039
Deaths	0.935526	0.920939	0.929357	1.000000	0.034865	-0.092946	0.246828	0.483275
Active Ratio	-0.025761	0.249863	-0.035694	0.034865	1.000000	-0.969990	0.215939	0.103380
Discharge Ratio	-0.000841	-0.279725	0.009900	-0.092946	-0.969990	1.000000	-0.446854	-0.106384
Death Ratio	0.100452	0.206191	0.093860	0.246828	0.215939	-0.446854	1.000000	0.047525
Population	0.525734	0.566020	0.522039	0.483275	0.103380	-0.106384	0.047525	1.000000

There is a strong positive correlation between Total Cases and Active Cases (0.86), Total Cases and Deaths (0.93), Total Cases and Discharged (0.99) and Active Cases vs Deaths (0.92).

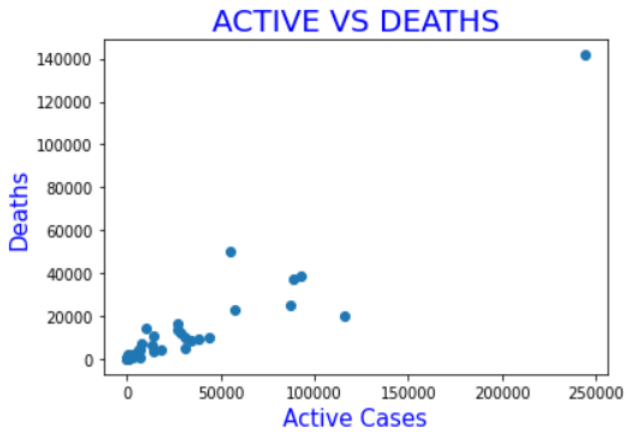
Therefore, in states like Maharashtra,Karnataka,West Bengal and Tamil Nadu where total cases are highest, the number of deaths is high and discharge is also high.

Thus, the correlations obtained between the variables as above are further proved by the following scatter plots:

ACTIVE VS DEATHS

In [140]:
```
plt.scatter(df["Active"],df["Deaths"])
plt.title("ACTIVE VS DEATHS",color="b",size=20)
plt.xlabel("Active Cases",color="blue",size=15)
plt.ylabel("Deaths",color="blue",size=15)
```

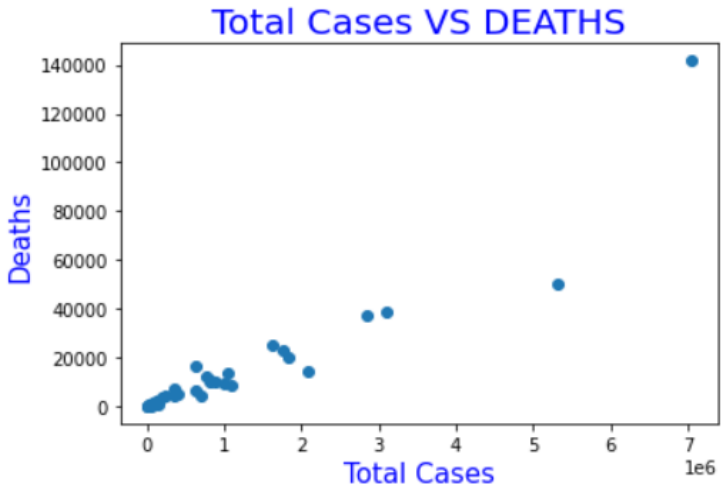
Out[140]:
```
Text(0, 0.5, 'Deaths')
```



TOTAL CASES vs DEATHS

In [138]:
```
plt.scatter(df["Total Cases"],df["Deaths"])
plt.title("Total Cases VS DEATHS",color="b",size=20)
plt.xlabel("Total Cases",color="blue",size=15)
plt.ylabel("Deaths",color="blue",size=15)
```

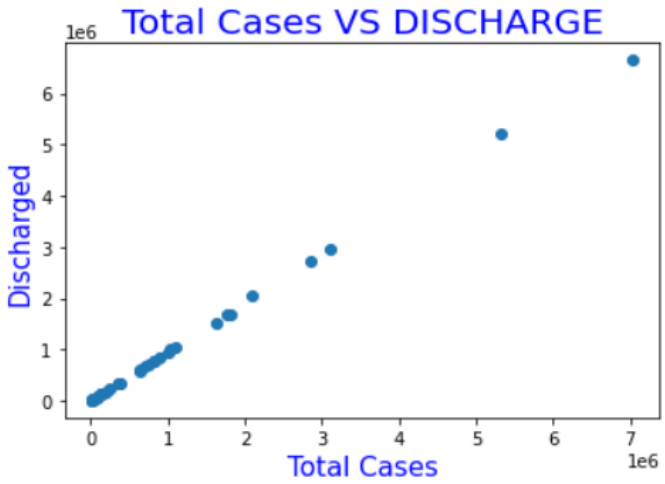
Out[138]:
```
Text(0, 0.5, 'Deaths')
```



TOTAL CASES vs. DISCHARGE

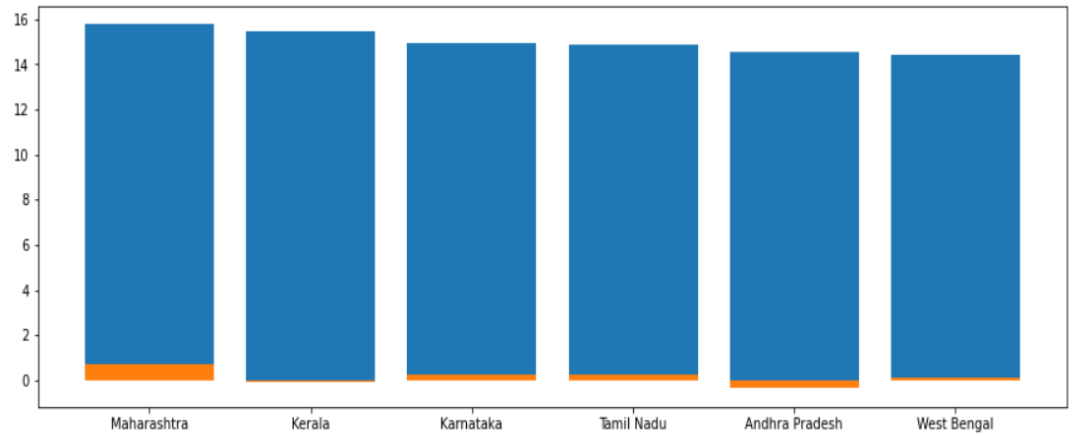
In [141]:
```
plt.scatter(df["Total Cases"],df["Discharged"])
plt.title("Total Cases VS DISCHARGE",color="b",size=20)
plt.xlabel("Total Cases",color="blue",size=15)
plt.ylabel("Discharged",color="blue",size=15)
```

Out[141]:
```
Text(0, 0.5, 'Discharged')
```



Due to this observed correlation between Total Cases and Death Ratio, we plot the bar chart showing Death Ratio in the top six states having maximum number of Total Cases, with respect to their Total Cases.

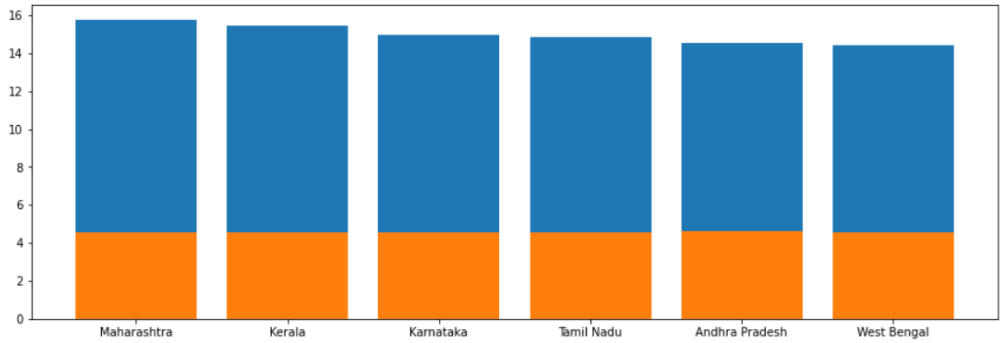
```
In [203]: plt.figure(figsize=(15,5))

          tot=df.sort_values(by=["Total Cases"],ascending=False)

          tot.reset_index(drop=True,inplace=True)
          city=tot.loc[:5,["State/UTs"]]["State/UTs"].to_list()
          plt.bar(city,np.log(tot.loc[:5,["Total Cases"]]["Total Cases"]))
          plt.bar(city,np.log(tot.loc[:5,["Death Ratio"]]["Death Ratio"]))

Out[203]: <BarContainer object of 6 artists>
```



We see that the proportion of Death Ratio with respect to Total cases is in maximum in states as Maharashtra , Karnataka, Tamil Nadu and Andhra Pradesh.

In terms of discharge ratio, considering that there is a strong correlation between discharge ratio and Total Cases, we plot the same :

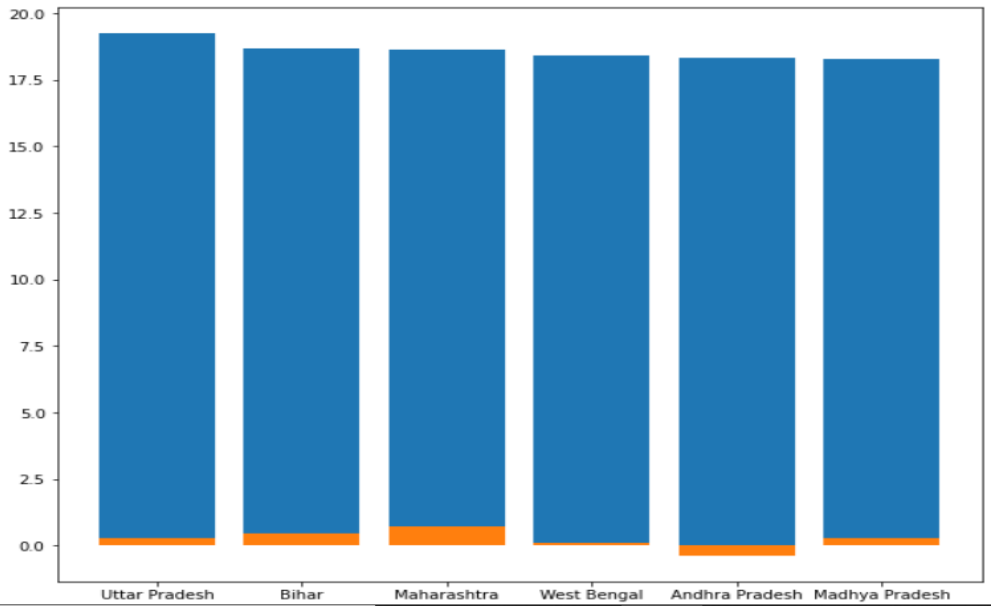
```
In [204]: plt.figure(figsize=(15,5))

          tot=df.sort_values(by=["Total Cases"],ascending=False)

          tot.reset_index(drop=True,inplace=True)
          city=tot.loc[:5,["State/UTs"]]["State/UTs"].to_list()
          plt.bar(city,np.log(tot.loc[:5,["Total Cases"]]["Total Cases"]))
          plt.bar(city,np.log(tot.loc[:5,["Discharge Ratio"]]["Discharge Ratio"]))

Out[204]: <BarContainer object of 6 artists>
```



And we conclude from the observation that more or less the discharge ratio is uniform among the States considered in terms of maximum Total Cases. (Maharashtra, Kerala, Karnataka, Tamil Nadu, Andhra Pradesh and West Bengal)

While considering the most populous states, it is observable that Maharashtra has the most discharge ratio out of all the 6 states where covid cases are maximum.

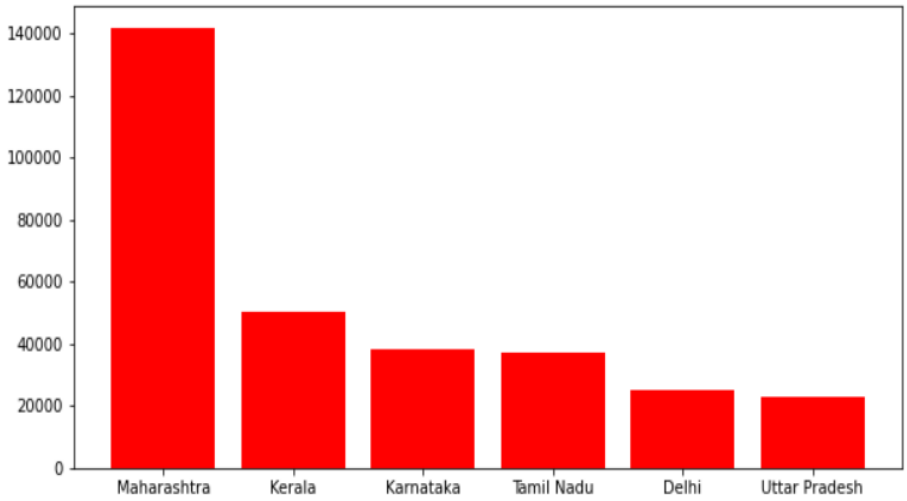
```
          tot.reset_index(drop=True,inplace=True)
          city=tot.loc[:5,["State/UTs"]]["State/UTs"].to_list()
          plt.bar(city,np.log(tot.loc[:5,["Population"]]["Population"]))
          plt.bar(city,np.log(tot.loc[:5,["Death Ratio"]]["Death Ratio"]))

Out[213]: <BarContainer object of 6 artists>
```



Now when we focus on the states having the most number of deaths, Maharashtra, Kerala, Karnataka, Tamil Nadu, Delhi and Uttar Pradesh come into prominence.

```
In [180]: plt.figure(figsize=(10,5))
          da=df.sort_values(by=["Deaths"],ascending=False)
          da.reset_index(drop=True,inplace=True)
          a1=da.loc[:5,["State/UTs"]]["State/UTs"].to_list()
          plt.bar(a1,da.loc[:5,["Deaths"]]["Deaths"].to_list(),color="r")
```

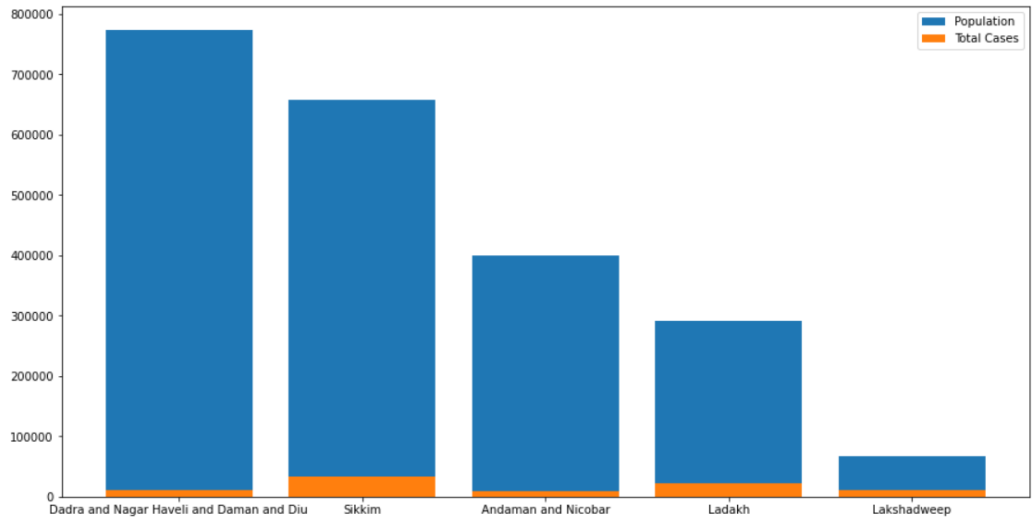
```
Out[180]: <BarContainer object of 6 artists>
```



Now when we consider that least populous states of India, Sikkim shows most number of Total Cases with respect to its population and also this data tallies with the number of deaths occurred.

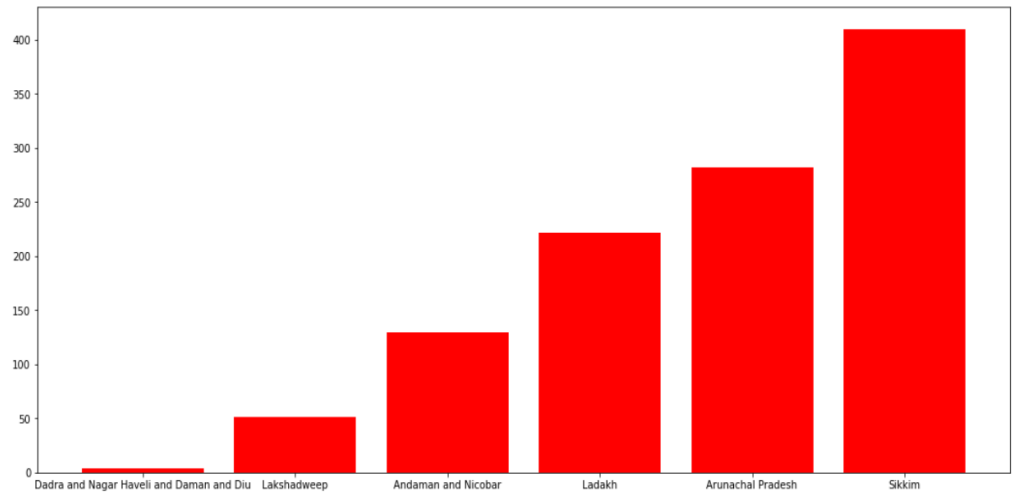
```
In [256]: k1=df.loc[31:35,["State/UTs"]]
          k2=df.loc[31:35,["Population"]]
          plt.figure(figsize=(15,8))
          plt.bar(k1["State/UTs"].to_list(),k2["Population"].to_list(),label="Population")
          plt.bar(k1["State/UTs"].to_list(),df.loc[31:35,["Total Cases"]]["Total Cases"].to_list(),label="Total Cases")
          plt.legend()
```

```
Out[256]: <matplotlib.legend.Legend at 0x20aded3f0a0>
```



Whereas, Dadra and Nagar Haveli shows minimum number of deaths due to Covid-19.

```
In [255]: plt.figure(figsize=(18,8))
          da=df.sort_values(by=["Deaths"],ascending=True)
          da.reset_index(drop=True,inplace=True)
          a1=da.loc[:5,["State/UTs"]]["State/UTs"].to_list()
          plt.bar(a1,da.loc[:5,["Deaths"]]["Deaths"].to_list(),color="r")

Out[255]: <BarContainer object of 6 artists>
```



The number of deaths due to covid-19 is minimum in Dadra and Nagar Haveli. Sikkim shows a large number of deaths compared to the other states considered here.

Results:

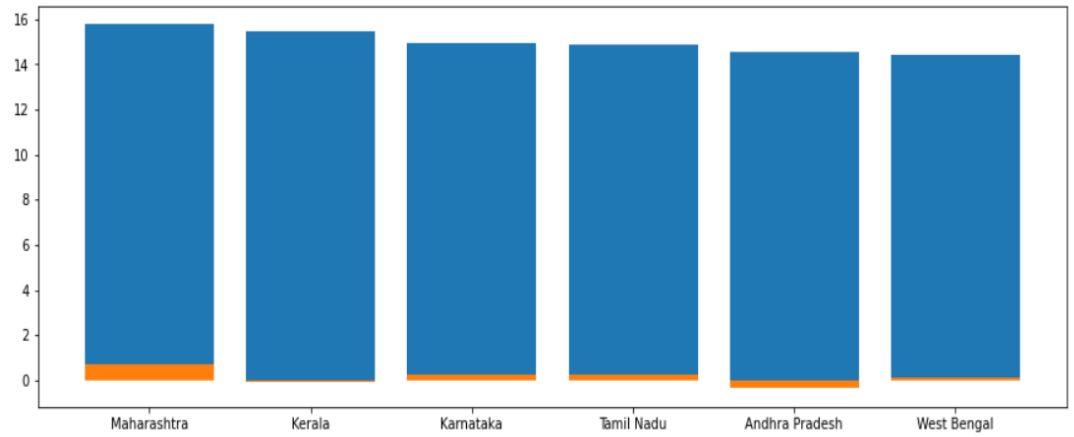
From our results we see that there is a strong linear correlation between variables like Active cases and Deaths, Total cases and active cases, Population and Active cases.

This tallies well with our observations from the dataset.

For example, Maharashtra having the greatest number of Total cases has most number of Deaths owing to the correlations between Total cases and Deaths.

In [203]:
```
plt.figure(figsize=(15,5))

tot=df.sort_values(by=["Total Cases"],ascending=False)

tot.reset_index(drop=True,inplace=True)
city=tot.loc[:5,["State/UTs"]]["State/UTs"].to_list()
plt.bar(city,np.log(tot.loc[:5,["Total Cases"]]["Total Cases"]))
plt.bar(city,np.log(tot.loc[:5,["Death Ratio"]]["Death Ratio"]))
```

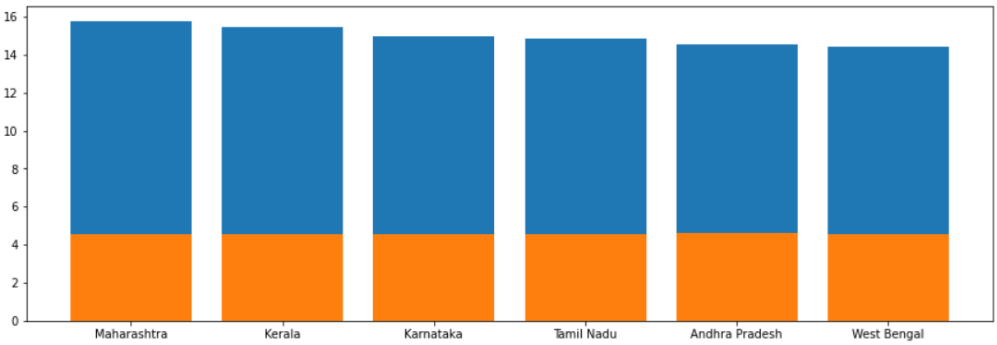
Out[203]:
```
<BarContainer object of 6 artists>
```



But when it comes to discharge ratio, which indicates the recovery rate of the state, it is observable that it is more or less the same in the States where total cases are maximum. (Particularly in the states of Maharashtra, Kerala, Karnataka, Tamil Nadu, Andhra Pradesh and West Bengal.)

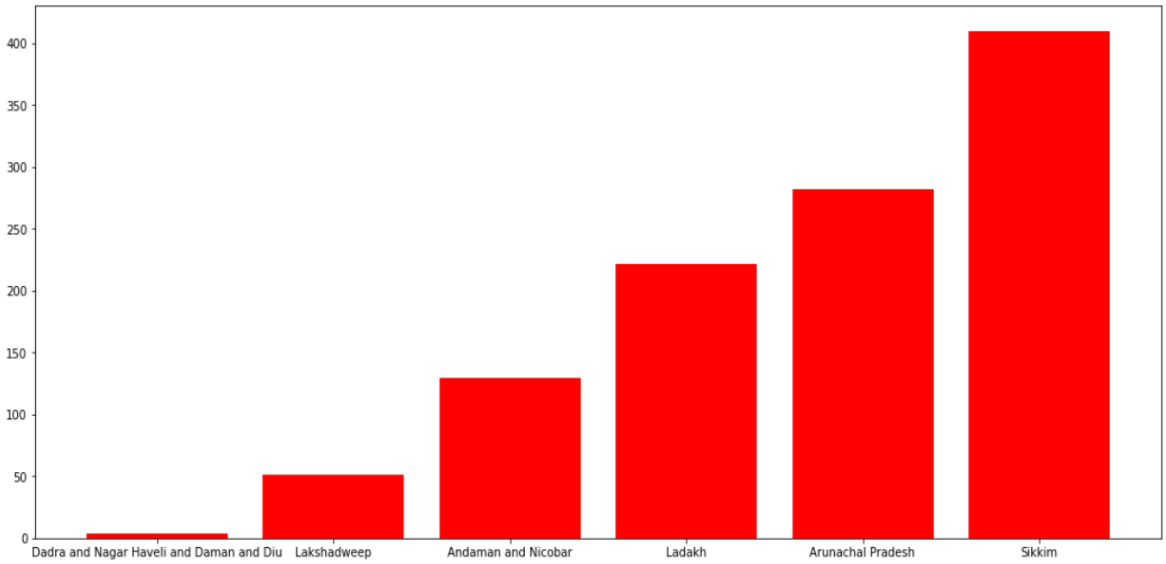
In [204]:
```
plt.figure(figsize=(15,5))

tot=df.sort_values(by=["Total Cases"],ascending=False)

tot.reset_index(drop=True,inplace=True)
city=tot.loc[:5,["State/UTs"]]["State/UTs"].to_list()
plt.bar(city,np.log(tot.loc[:5,["Total Cases"]]["Total Cases"]))
plt.bar(city,np.log(tot.loc[:5,["Discharge Ratio"]]["Discharge Ratio"]))
```

Out[204]:
```
<BarContainer object of 6 artists>
```



Another important insight we got from the data is that when it comes to the least populated states, Sikkim had the most number of cases as well as the most number of deaths. This tallies with the actual information that the first instances Covid-19 cases in India appeared in the state of Sikkim.

```
In [255]: plt.figure(figsize=(18,8))
          da=df.sort_values(by=["Deaths"],ascending=True)
          da.reset_index(drop=True,inplace=True)
          a1=da.loc[:5,["State/UTs"]]["State/UTs"].to_list()
          plt.bar(a1,da.loc[:5,["Deaths"]]["Deaths"].to_list(),color="r")

Out[255]: <BarContainer object of 6 artists>
```



The number of deaths due to covid-19 is minimum in Dadra and Nagar Haveli. Sikkim shows a large number of deaths compared to the other states considered here.

Conclusion:

In our analysis we have been able to answer the questions that we formulated before the analysis began and this helped us in obtaining important insights regarding the status of covid-19 situation in India.

References:

1. The numbers reported are as of August 14, 2020.
   See https://coronavirus.jhu.edu/map.html and https://nssac.bii.virginia.edu/covid-19/dashboard/ for most up to date surveillance information.

2. see https://www.nytimes.com/news-event/coronavirus.