

```
In [2]: #!/ pip install seaborn
import pandas as pd
import numpy as np
import seaborn as sns
```

Collecting seaborn  
  Downloading seaborn-0.11.1-py3-none-any.whl (285 kB)  
Requirement already satisfied: matplotlib>=2.2 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from seaborn) (3.4.2)  
Requirement already satisfied: pandas>=0.23 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from seaborn) (1.2.4)  
Requirement already satisfied: numpy>=1.15 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from seaborn) (1.20.2)  
Collecting scipy>=1.0  
  Downloading scipy-1.7.1-cp39-cp39-win\_amd64.whl (33.8 MB)  
Requirement already satisfied: python-dateutil>=2.7 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from matplotlib>=2.2->seaborn) (2.8.1)  
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from matplotlib>=2.2->seaborn) (1.3.1)  
Requirement already satisfied: cycler>=0.10 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from matplotlib>=2.2->seaborn) (0.10.0)  
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from matplotlib>=2.2->seaborn) (2.4.7)  
Requirement already satisfied: pillow>=6.2.0 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from matplotlib>=2.2->seaborn) (8.3.1)  
Requirement already satisfied: pytz>=2017.3 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from pandas>=0.23->seaborn) (2021.1)  
Requirement already satisfied: six>=1.5 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from python-dateutil>=2.7->matplotlib>=2.2->seaborn) (1.15.0)  
Installing collected packages: scipy, seaborn  
Successfully installed scipy-1.7.1 seaborn-0.11.1

WARNING: You are using pip version 20.2.3; however, version 21.2.3 is available.  
You should consider upgrading via the 'c:\users\hp\appdata\local\programs\python\python39\python.exe -m pip install --upgrade pip' command.

## Descriptive analysis

```
In [3]: data = pd.read_csv('E:\Latest Covid-19 India Status.csv')
```

```
In [4]: data.head()
```

```
Out[4]:
```

	State/UTs	Total Cases	Active	Discharged	Deaths	Active Ratio (%)	Discharge Ratio (%)	Death Ratio (%)
0	Andaman and Nicobar	7539	6	7404	129	0.08	98.21	1.71
1	Andhra Pradesh	1970008	20582	1936016	13410	1.04	98.27	0.68
2	Arunachal Pradesh	48565	3508	44823	234	7.22	92.29	0.48
3	Assam	568257	12429	550534	5294	2.19	96.88	0.93
4	Bihar	724917	401	714872	9644	0.06	98.61	1.33

```
In [6]: data.columns
```

```
Out[6]: Index(['State/UTs', 'Total Cases', 'Active', 'Discharged', 'Deaths',
      'Active Ratio (%)', 'Discharge Ratio (%)', 'Death Ratio (%)'],
      dtype='object')
```

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

```
Out[7]: (36, 8)
```

```
In [8]: data.describe()
```

```
Out[8]:
```

	Total Cases	Active	Discharged	Deaths	Active Ratio (%)	Discharge Ratio (%)	Death Ratio (%)
count	3.600000e+01	36.000000	3.600000e+01	36.000000	36.000000	36.000000	36.000000
mean	8.812919e+05	11248.833333	8.582321e+05	11810.972222	2.679722	96.059167	1.261667
std	1.273276e+06	29957.488684	1.231874e+06	22842.176185	5.706423	5.592844	0.567604
min	7.539000e+03	6.000000	7.404000e+03	4.000000	0.020000	68.910000	0.040000
25%	6.494425e+04	363.500000	6.058375e+04	797.000000	0.080000	96.822500	0.937500
50%	4.577400e+05	1277.000000	4.461945e+05	5211.500000	0.670000	98.035000	1.320000
75%	9.854172e+05	10061.250000	9.670425e+05	13439.500000	1.575000	98.615000	1.605000
max	6.315063e+06	165834.000000	6.103325e+06	133038.000000	30.700000	99.820000	2.720000

```
In [9]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 36 entries, 0 to 35
Data columns (total 8 columns):
#   Column                Non-Null Count  Dtype
---  --
0   State/UTs              36 non-null     object
1   Total Cases            36 non-null     int64
2   Active                 36 non-null     int64
3   Discharged             36 non-null     int64
4   Deaths                36 non-null     int64
5   Active Ratio (%)       36 non-null     float64
6   Discharge Ratio (%)    36 non-null     float64
7   Death Ratio (%)        36 non-null     float64
dtypes: float64(3), int64(4), object(1)
memory usage: 2.4+ KB
```

```
In [11]: data.isnull().sum()
```

```
Out[11]: State/UTs      0
Total Cases    0
Active          0
Discharged      0
Deaths          0
Active Ratio (%) 0
Discharge Ratio (%) 0
Death Ratio (%) 0
dtype: int64
```

if there is any null value replace it with mean value of that column to reduce redundancies using data=fillna()

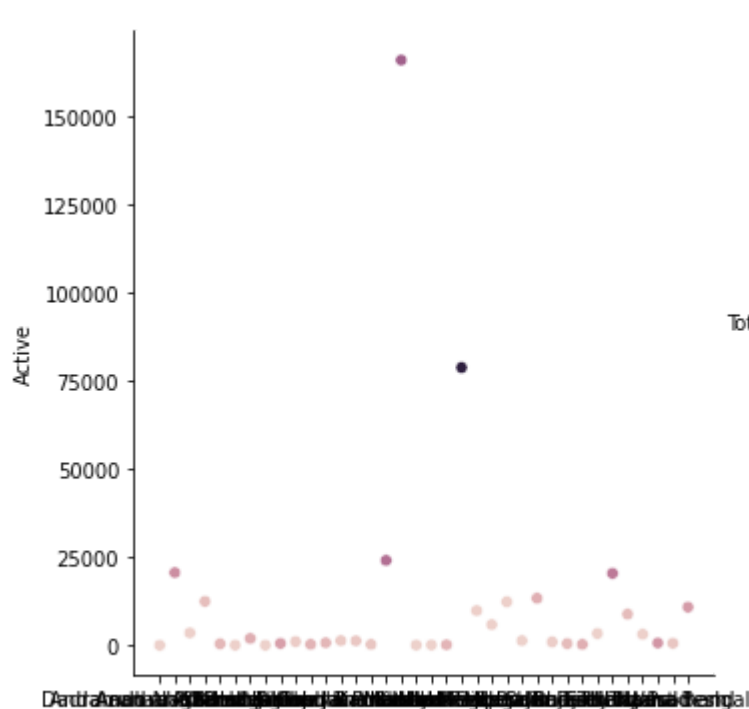
```
In [12]: data.head()
```

```
Out[12]:
```

	State/UTs	Total Cases	Active	Discharged	Deaths	Active Ratio (%)	Discharge Ratio (%)	Death Ratio (%)
0	Andaman and Nicobar	7539	6	7404	129	0.08	98.21	1.71
1	Andhra Pradesh	1970008	20582	1936016	13410	1.04	98.27	0.68
2	Arunachal Pradesh	48565	3508	44823	234	7.22	92.29	0.48
3	Assam	568257	12429	550534	5294	2.19	96.88	0.93
4	Bihar	724917	401	714872	9644	0.06	98.61	1.33

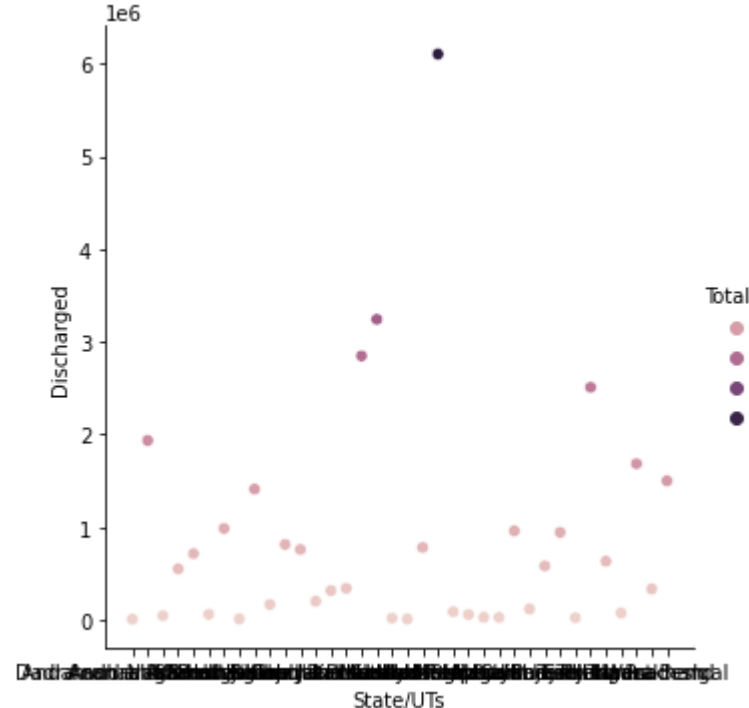
```
In [20]: sns.relplot(x='State/UTs', y='Active',hue='Total Cases',data=data)
```

```
Out[20]: <seaborn.axisgrid.FacetGrid at 0x15b06be4070>
```



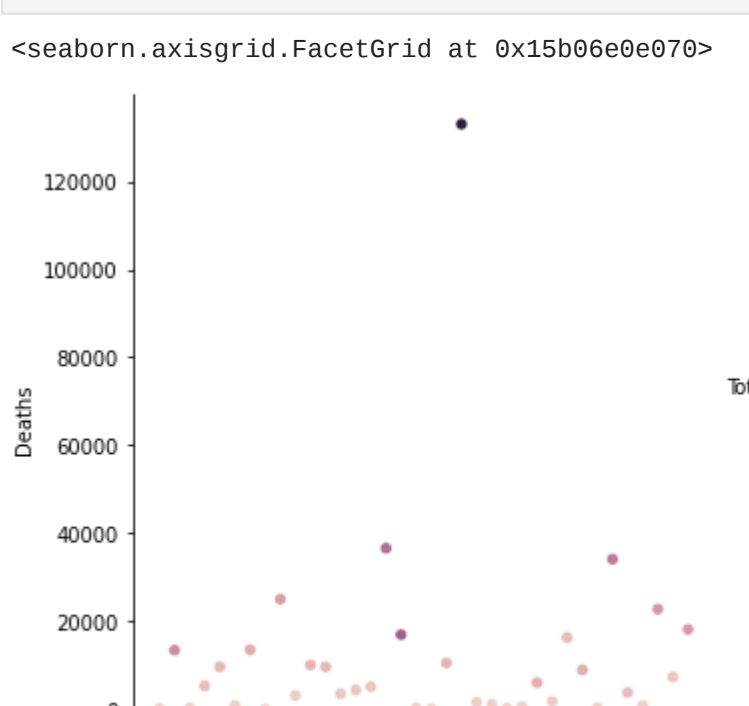
```
In [21]: sns.relplot(x='State/UTs', y='Discharged',hue='Total Cases', data=data)
```

```
Out[21]: <seaborn.axisgrid.FacetGrid at 0x15b06d43070>
```



```
In [22]: sns.relplot(x='State/UTs', y='Deaths',hue='Total Cases', data=data)
```

```
Out[22]: <seaborn.axisgrid.FacetGrid at 0x15b06e0e070>
```

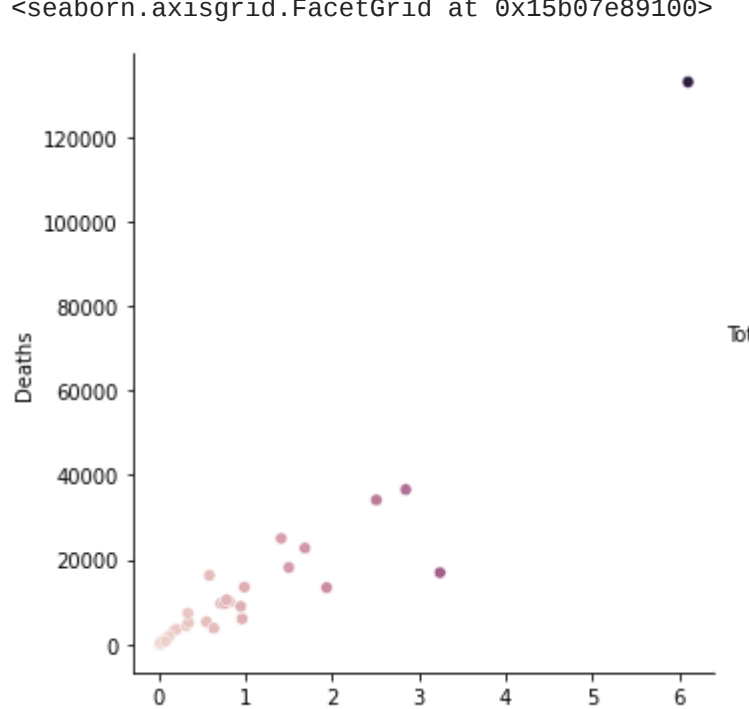


from above graphs you can see that there is a linearity in active cases and deaths but not in discharged rate. That shows infection has been spread and affected all over country but only some states were prepared and controlled it by discharge rates we can conclude which states might have better medical infrastructure.

## Exploratory analysis

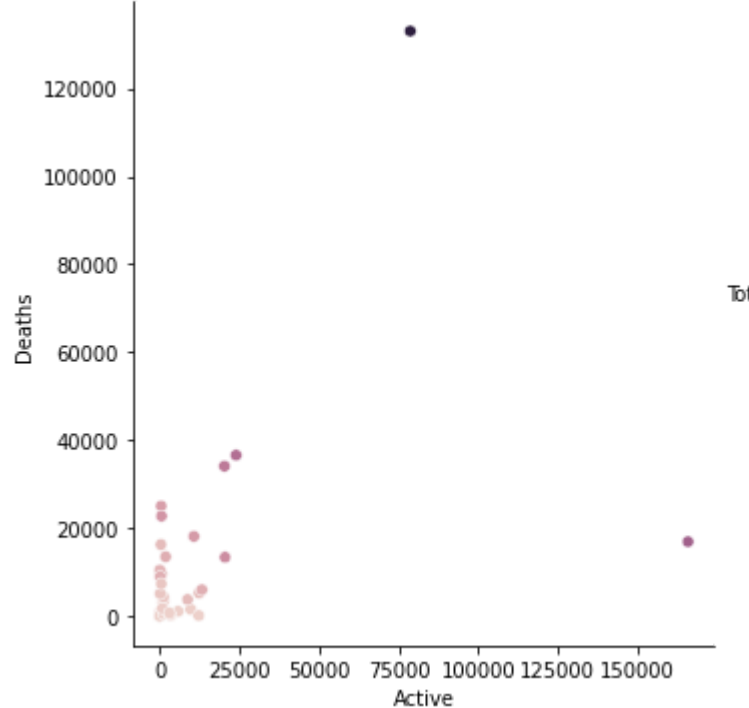
```
In [23]: sns.relplot(x='Discharged', y='Deaths',hue='Total Cases', data=data)
```

```
Out[23]: <seaborn.axisgrid.FacetGrid at 0x15b07e09100>
```



```
In [24]: sns.relplot(x='Active', y='Deaths',hue='Total Cases', data=data)
```

```
Out[24]: <seaborn.axisgrid.FacetGrid at 0x15b07f009d0>
```



from above graphs we can conclude that many states having poor infrastructure or strategy, has linear death and discharge ratio, meaning people have same chances as dying or surviving which points that even with less amount of cases comparing to others they are struggling with handling the pandemic. Active vs death ratio is even worse where bunch of states totally failed to handle the situation as the deaths per active cases seems very high.

## Predictive analysis

```
In [26]: #!/pip install sklearn
```

Collecting sklearn  
WARNING: You are using pip version 20.2.3; however, version 21.2.3 is available.  
You should consider upgrading via the 'c:\users\hp\appdata\local\programs\python\python39\python.exe -m pip install --upgrade pip' command.  
Collecting sklearn-0.0.tar.gz (1.1 kB)  
  Downloading scikit\_learn-0.24.2-cp39-cp39-win\_amd64.whl (6.9 MB)  
Collecting joblib>=0.11  
  Downloading joblib-1.0.1-py3-none-any.whl (303 kB)  
Requirement already satisfied: numpy>=1.13.3 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from scikit\_learn->sklearn) (1.20.2)  
Collecting threadpoolctl>=2.0.0  
  Downloading threadpoolctl-2.2.0-py3-none-any.whl (12 kB)  
Requirement already satisfied: scipy>=0.19.1 in c:\users\hp\appdata\local\programs\python\python39\lib\site-packages (from scikit\_learn->sklearn) (1.7.1)  
Building wheels for collected packages: sklearn  
  Building wheel for sklearn (setup.py): started  
  Building wheel for sklearn (setup.py): finished with status 'done'  
  Created wheel for sklearn: filename=sklearn-0.0-py2.py3-none-any.whl size=1316 sha256=378b7da32b89836e191fa6755e2f076c8dbab6d93ef52e5081de2e8b3ba8dfae  
  Stored in directory: c:\users\hp\appdata\local\pip\cache\wheels\ea\7b\98\b6466d71b8d738a0c54708b9eb39bf676d1ff6ca4b22af1c  
Successfully built sklearn  
Installing collected packages: joblib, threadpoolctl, scikit\_learn, sklearn  
Successfully installed joblib-1.0.1 scikit\_learn-0.24.2 sklearn-0.0 threadpoolctl-2.2.0

```
In [27]: from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
```

```
In [28]: train = data.drop(['Total Cases', 'State/UTs'], axis=1)
```

```
In [29]: test = data['Total Cases']
```

```
In [30]: x_train, x_test, y_train, y_test = train_test_split(train, test, test_size=0.3, random_state=2)
```

```
In [31]: regr = LinearRegression()
```

```
In [32]: regr.fit(x_train, y_train)
```

```
Out[32]: LinearRegression()
```

```
In [33]: pred = regr.predict(x_test)
```

```
In [35]: pred
```

```
Out[35]: array([[ 286369.,   321725.,   953704.,  2563544.,   28004.,   171295.,
        599162.,    7539.,   979737.,   347223.,  1970008.]])
```

```
In [ ]:
```