

DAV Team Project

Project Report

SUPERVISOR: Dr. Sharanjit Kaur

SUBMITTED BY:

Sambit Basu , AB-1264 , 19001570034

Sarvesh Sharma , AB-1271 , 19001570036

Tushar Gaurav , AB-1253 , 19001570050

Program Name: B.Sc. (H.) Computer Science

Semester: 5

Title of The Paper: DATA ANALYSIS AND VISUALISATION (DAV)



2021

Department of Computer Science
ACHARYA NARENDRA DEV COLLEGE

Title of the project

Analysis and Visualisation of Covid -19 Data in India

Objective

Analysis and Visualisation of Covid -19 Data for confirmed cases, deaths and recovered individuals of each state in India for each day from 30/01/2020 to 11/08/2021.

Data used along with link from where it is downloaded

https://www.kaggle.com/sudalairajkumar/covid19-in-india?select=covid_19_india.csv

	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
...
18105	18106	2021-08-11	8:00 AM	Telangana	-	-	638410	3831	650353
18106	18107	2021-08-11	8:00 AM	Tripura	-	-	77811	773	80660
18107	18108	2021-08-11	8:00 AM	Uttarakhand	-	-	334650	7368	342462
18108	18109	2021-08-11	8:00 AM	Uttar Pradesh	-	-	1685492	22775	1708812
18109	18110	2021-08-11	8:00 AM	West Bengal	-	-	1506532	18252	1534999

18110 rows × 9 columns

Description about data

We have 18110 records and 9 columns, namely : Sno, Date, Time, State/UnionTerritory, ConfirmedIndianNational, ConfirmedForeignNational, Cured, Deaths, Confirmed.

The values for ConfirmedIndianNational and ConfirmedForeignNational are only updated till 28/03/2020.

We do not require the columns Sno, Time, ConfirmedIndianNational, ConfirmedForeignNational for our analysis, so, we drop them.

Queries performed on the data along with code and DAV concepts used

Code snippet 1: (Importing important libraries)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import matplotlib
import datetime
import seaborn as sns
```

Code snippet 2: (Uploading dataset)

```
from google.colab import files
uploaded=files.upload()
```

Code snippet 3:

```
covid=pd.read_csv('/content/covid_19_india.csv')
covid.head()
```

	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

Code snippet 4: (Setting parameters for plots)

```
plt.rc('figure',figsize=(10,10))
font_options={'family':'serif','weight':'normal','size':15}
plt.rc('font',**font_options)
```

Code snippet 5: (Dropping unwanted attributes)

```
covid.drop(['ConfirmedIndianNational','ConfirmedForeignNational','Sno','Time'],inplace=True,axis=1)
```

```
covid.tail()
```

	Date	State/UnionTerritory	Cured	Deaths	Confirmed
18105	2021-08-11	Telangana	638410	3831	650353
18106	2021-08-11	Tripura	77811	773	80660
18107	2021-08-11	Uttarakhand	334650	7368	342462
18108	2021-08-11	Uttar Pradesh	1685492	22775	1708812
18109	2021-08-11	West Bengal	1506532	18252	1534999

1. For each Indian state, find maximum cases reported for confirmed, deaths and recovered individually along with the date on which these cases were reported for any three months of the year 2020. Display the result in the self-explanatory format.

Code snippet 1:

```
q1=pd.DataFrame(columns=[['August','August','August','August','August','August',  
'September','September','September','September','September','September','October',  
'October','October','October','October','October'],['Confirmed','Confirmed','Deaths','Dea  
ths','Cured','Cured','Confirmed','Confirmed','Deaths','Deaths','Cured','Cured','Confirme  
d','Confirmed','Deaths','Deaths','Cured','Cured'],['Date','Number','Date','Number','Date'  
, 'Number','Date','Number','Date','Number','Date','Number','Date','Number','Date','Num  
ber','Date','Number']])  
q1.index.name='State'
```

Code snippet 2:

```
for sta in covid[(covid['Date']==datetime.date(2020,8,31)) | (covid['Date']==datetime.d  
ate(2020,9,30)) | (covid['Date']==datetime.date(2020,10,31))]['State/UnionTerritory'].u  
nique():  
    Li=[]  
    for z in range(8,11):  
        if z==9:  
            fin=31  
        else:  
            fin=32  
        conf,deat,cure=pd.Series(),pd.Series(),pd.Series()  
  
    for d in range(2,fin):  
        conf.loc[datetime.date(2020,z,d-1)]=int(covid[(covid['State/UnionTerritory']==sta) &  
(covid['Date']==datetime.date(2020,z,d))].Confirmed)-int(covid[(covid['State/UnionTer  
ritory']==sta) & (covid['Date']==datetime.date(2020,z,d-1))].Confirmed)
```

```

deat.loc[datetime.date(2020,z,d-1)]=int(covid[(covid['State/Union Territory']==sta)
& (covid['Date']==datetime.date(2020,z,d))].Deaths)-int(covid[(covid['State/Union Territory']==sta) & (covid['Date']==datetime.date(2020,z,d-1))].Deaths)
cure.loc[datetime.date(2020,z,d-1)]=int(covid[(covid['State/Union Territory']==sta)
& (covid['Date']==datetime.date(2020,z,d))].Cured)-int(covid[(covid['State/Union Territory']==sta) & (covid['Date']==datetime.date(2020,z,d-1))].Cured)
Li.extend([str(conf.idxmax()),conf.max()])
Li.extend([str(deat.idxmax()),deat.max()])
Li.extend([str(cure.idxmax()),cure.max()])
q1.loc[sta]=Li
q1.head(12)

```

State	August						September						October					
	Confirmed		Deaths		Cured		Confirmed		Deaths		Cured		Confirmed		Deaths		Cured	
	Date	Number	Date	Number	Date	Number	Date	Number	Date	Number	Date	Number	Date	Number	Date	Number	Date	Number
Andaman and Nicobar Islands	2020-08-14	149	2020-08-16	4	2020-08-09	148	2020-09-06	40	2020-09-02	1	2020-09-01	67	2020-10-21	27	2020-10-05	1	2020-10-19	24
Andhra Pradesh	2020-08-26	10830	2020-08-08	97	2020-08-01	12750	2020-09-05	10825	2020-09-01	84	2020-09-04	12154	2020-10-01	6751	2020-10-08	42	2020-10-03	7798
Arunachal Pradesh	2020-08-30	157	2020-08-30	2	2020-08-24	130	2020-09-28	328	2020-09-15	2	2020-09-21	235	2020-10-09	281	2020-10-13	4	2020-10-07	431
Assam	2020-08-12	4593	2020-08-26	14	2020-08-23	3259	2020-09-28	3644	2020-09-12	23	2020-09-11	2628	2020-10-03	1632	2020-10-06	18	2020-10-25	2961
Bihar	2020-08-09	4157	2020-08-02	20	2020-08-17	4128	2020-09-02	3092	2020-09-01	39	2020-09-08	2246	2020-10-04	1727	2020-10-10	10	2020-10-05	1537
Chandigarh	2020-08-29	261	2020-08-30	7	2020-08-27	181	2020-09-13	449	2020-09-23	10	2020-09-22	383	2020-10-02	155	2020-10-02	5	2020-10-06	238
Chhattisgarh	2020-08-29	1513	2020-08-14	16	2020-08-29	709	2020-09-26	3896	2020-09-09	70	2020-09-25	8027	2020-10-09	2958	2020-10-23	58	2020-10-06	3484
Dadra and Nagar Haveli and Daman and Diu	2020-08-09	108	2020-08-01	0	2020-08-03	94	2020-09-05	70	2020-09-01	0	2020-09-05	54	2020-10-04	17	2020-10-01	0	2020-10-02	20
Delhi	2020-08-30	2024	2020-08-01	26	2020-08-29	1449	2020-09-16	4473	2020-09-29	48	2020-09-26	4476	2020-10-30	5891	2020-10-26	54	2020-10-30	4433
Goa	2020-08-13	570	2020-08-21	9	2020-08-22	479	2020-09-12	740	2020-09-14	14	2020-09-22	736	2020-10-01	524	2020-10-01	12	2020-10-06	594
Gujarat	2020-08-29	1282	2020-08-06	27	2020-08-07	1370	2020-09-25	1442	2020-09-14	17	2020-09-17	1652	2020-10-07	1473	2020-10-02	15	2020-10-07	1546
Haryana	2020-08-26	1397	2020-08-28	15	2020-08-26	1117	2020-09-12	2783	2020-09-20	29	2020-09-22	2892	2020-10-30	1650	2020-10-03	25	2020-10-01	2161

2. Use appropriate year-month string date conversions.

Code snippet 1:

```
covid.Date=[pd.to_datetime(d) for d in covid.Date]
covid.Date=[datetime.datetime.date(f) for f in covid.Date]
type(covid.Date[0])
```

```
datetime.date
```

Code snippet 2:

2.(a) Find the state that ended 2020 with the most number of cured cases

```
covid.loc[covid[covid['Date']==datetime.date(2020,12,31)].Cured.idxmax()]['State/UnionTerritory']
```

```
'Maharashtra'
```

2.(b) Find the rate of increase in confirmed cases after Durga Pujo in West Bengal and Chhat puja in Bihar

Code snippet 3:

```
wb7oct=int(covid[(covid['State/UnionTerritory']=='West Bengal') & (covid['Date']==datetime.date(2020,10,7))].Confirmed)
wb17oct=int(covid[(covid['State/UnionTerritory']=='West Bengal') & (covid['Date']==datetime.date(2020,10,17))].Confirmed)
```

```
wb27oct=int(covid[(covid['State/UnionTerritory']=='West Bengal') & (covid['Date']==datetime.date(2020,10,27))].Confirmed)
k,y=wb17oct-wb7oct,wb27oct-wb17oct
wbdpr=[(k/wb7oct)*100,(y/wb17oct)*100,((y-k)/k)*100]
```

```
br8nov=int(covid[(covid['State/UnionTerritory']=='Bihar') & (covid['Date']==datetime.date(2020,11,8))].Confirmed)
br13nov=int(covid[(covid['State/UnionTerritory']=='Bihar') & (covid['Date']==datetime.date(2020,11,13))].Confirmed)
br18nov=int(covid[(covid['State/UnionTerritory']=='Bihar') & (covid['Date']==datetime.date(2020,11,18))].Confirmed)
a,b=br13nov-br8nov,br18nov-br13nov
brcpr=[(a/br8nov)*100,(b/br13nov)*100,((b-a)/a)*100]
```

```
pujorates=pd.DataFrame(index=['West Bengal','Bihar'],columns=['Rate Before','Rate After','Increase Rate'])
pujorates.loc['West Bengal']=wbdpr
pujorates.loc['Bihar']=brcpr
```

```
pujorates
```

	Rate Before	Rate After	Increase Rate
West Bengal	13.0443	12.9743	12.4381
Bihar	1.624	1.11079	-30.4905

3. Create subplots (line graph) for showing total number of cured cases month-wise from April 2020 to March 2021 in four states namely Karnataka, Gujarat, Haryana, and Uttar Pradesh.

Code snippet 1:

```
ka,guj,har,up=[],[],[],[]
for i in range(4,13):
    if i in [4,6,9,11]:
        ka.append(int(covid[(covid['State/UnionTerritory']=='Karnataka') & (covid['Date']==datetime.date(2020,i,30))].Cured.values))
        guj.append(int(covid[(covid['State/UnionTerritory']=='Gujarat') & (covid['Date']==datetime.date(2020,i,30))].Cured.values))
        har.append(int(covid[(covid['State/UnionTerritory']=='Haryana') & (covid['Date']==datetime.date(2020,i,30))].Cured.values))
        up.append(int(covid[(covid['State/UnionTerritory']=='Uttar Pradesh') & (covid['Date']==datetime.date(2020,i,30))].Cured.values))
    else:
        ka.append(int(covid[(covid['State/UnionTerritory']=='Karnataka') & (covid['Date']==datetime.date(2020,i,31))].Cured.values))
        guj.append(int(covid[(covid['State/UnionTerritory']=='Gujarat') & (covid['Date']==datetime.date(2020,i,31))].Cured.values))
        har.append(int(covid[(covid['State/UnionTerritory']=='Haryana') & (covid['Date']==datetime.date(2020,i,31))].Cured.values))
        up.append(int(covid[(covid['State/UnionTerritory']=='Uttar Pradesh') & (covid['Date']==datetime.date(2020,i,31))].Cured.values))

for i in range(1,4):
    if i in [1,3]:
        ka.append(int(covid[(covid['State/UnionTerritory']=='Karnataka') & (covid['Date']==datetime.date(2021,i,30))].Cured.values))
        guj.append(int(covid[(covid['State/UnionTerritory']=='Gujarat') & (covid['Date']==datetime.date(2021,i,30))].Cured.values))
        har.append(int(covid[(covid['State/UnionTerritory']=='Haryana') & (covid['Date']==datetime.date(2021,i,30))].Cured.values))
        up.append(int(covid[(covid['State/UnionTerritory']=='Uttar Pradesh') & (covid['Date']==datetime.date(2021,i,30))].Cured.values))
```

```

else:
    ka.append(int(covid[(covid['State/UnionTerritory']=='Karnataka') & (covid['Date']==datetime.date(2021,i,28))].Cured.values))
    guj.append(int(covid[(covid['State/UnionTerritory']=='Gujarat') & (covid['Date']==datetime.date(2021,i,28))].Cured.values))
    har.append(int(covid[(covid['State/UnionTerritory']=='Haryana') & (covid['Date']==datetime.date(2021,i,28))].Cured.values))
    up.append(int(covid[(covid['State/UnionTerritory']=='Uttar Pradesh') & (covid['Date']==datetime.date(2021,i,28))].Cured.values))

```

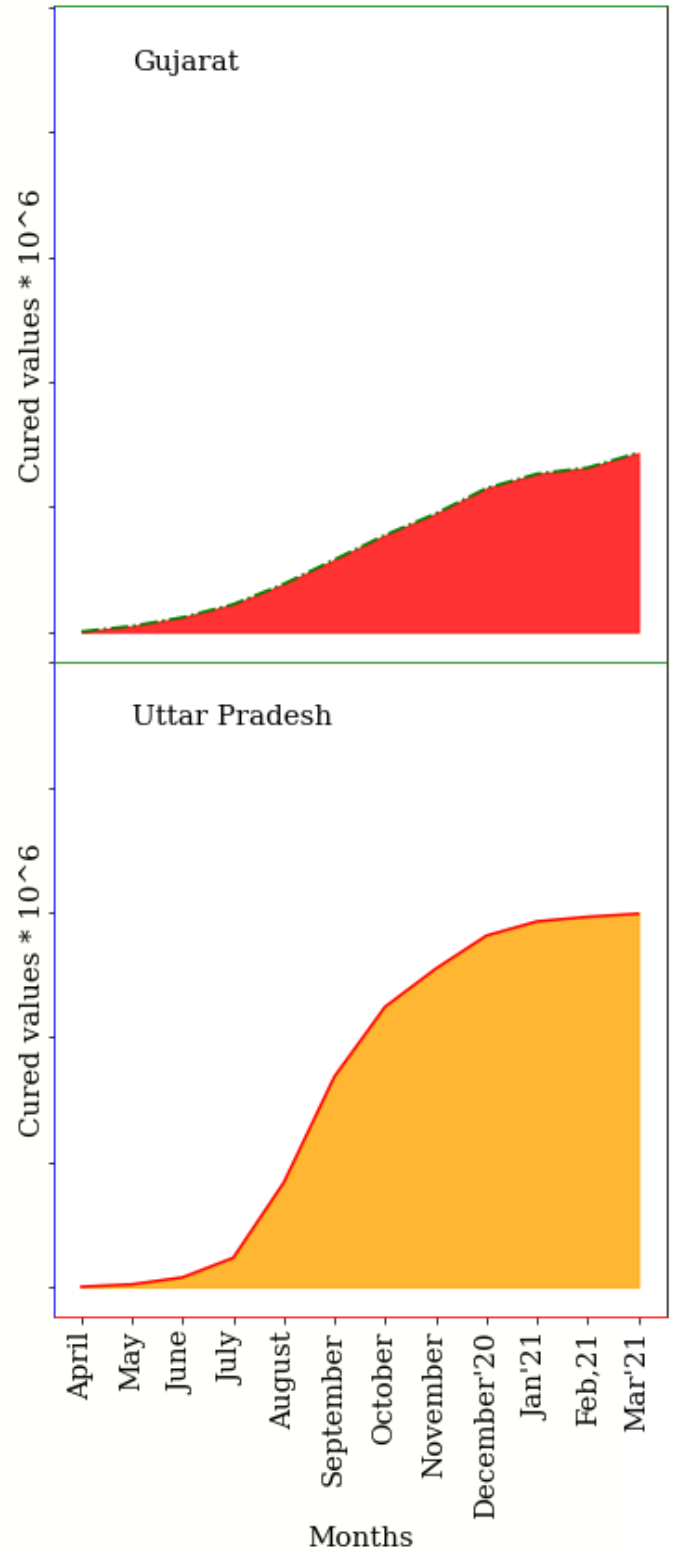
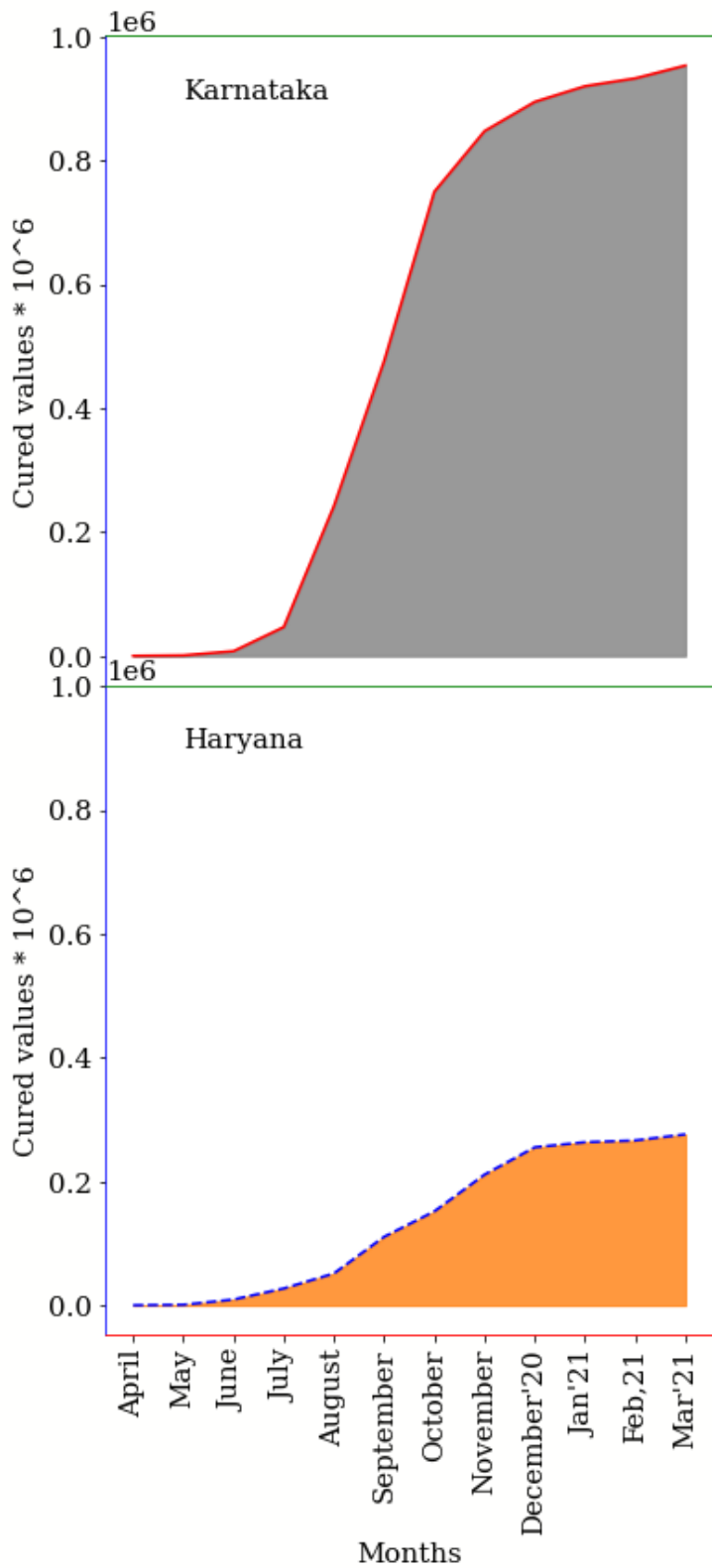
Code snippet 2:

```

fig,axes=plt.subplots(2,2,sharey=True,sharex=True)
states=['Karnataka','Gujarat','Haryana','Uttar Pradesh']
months=["April","May","June",
        "July","August","September",
        "October","November","December'20",
        "Jan'21","Feb,21","Mar'21"]
c=0
hy=[ka,guj,har,up]
colours=['r-','g-','b--','r-']
col=['grey','r','C1','orange']
for f in range(2):
    for t in range(2):
        axes[f,t].plot(months,hy[c],colours[c])
        axes[f,t].text(1,900000,states[c])
        axes[f,t].set_xticklabels(months,rotation='vertical')
        axes[f,t].fill_between(months,hy[c],color=col[c],alpha=0.8)
        axes[f,t].spines['top'].set_color('green')
        axes[f,t].spines['bottom'].set_color('red')
        axes[f,t].spines['right'].set_color('black')
        axes[f,t].spines['left'].set_color('blue')
        axes[f,t].set_xlabel('Months')
        axes[f,t].set_ylabel('Cured values * 10^6')
        c+=1
fig.suptitle('Cured values of 4 states for 12 months')
plt.subplots_adjust(wspace=0.2,hspace=0)

```

Cured values of 4 states for 12 months



4. Compare the deaths due to Covid-19 in the months of May 2020 and May 2021 for the states namely Karnataka, Delhi, and Madhya Pradesh using stacked bars.

Code snippet 1:

```
types=['May-2020','May-2021']
karn,delhi,mad=pd.Series(dtype='int64',index=types),pd.Series(dtype='int64',index=types),pd.Series(dtype='int64',index=types)
```

```
karn['May-2020']=int(covid[(covid['State/UnionTerritory']=='Karnataka') & (covid.Date==datetime.date(2020,5,31))].Deaths.values)-int(covid[(covid['State/UnionTerritory']=='Karnataka') & (covid.Date==datetime.date(2020,4,30))].Deaths.values)
```

```
karn['May-2021']=int(covid[(covid['State/UnionTerritory']=='Karnataka') & (covid.Date==datetime.date(2021,5,31))].Deaths.values)-int(covid[(covid['State/UnionTerritory']=='Karnataka') & (covid.Date==datetime.date(2021,4,30))].Deaths.values)
```

```
delhi['May-2020']=int(covid[(covid['State/UnionTerritory']=='Delhi') & (covid.Date==datetime.date(2020,5,31))].Deaths.values)-int(covid[(covid['State/UnionTerritory']=='Delhi') & (covid.Date==datetime.date(2020,4,30))].Deaths.values)
```

```
delhi['May-2021']=int(covid[(covid['State/UnionTerritory']=='Delhi') & (covid.Date==datetime.date(2021,5,31))].Deaths.values)-int(covid[(covid['State/UnionTerritory']=='Delhi') & (covid.Date==datetime.date(2021,4,30))].Deaths.values)
```

```
mad['May-2020']=int(covid[(covid['State/UnionTerritory']=='Madhya Pradesh') & (covid.Date==datetime.date(2020,5,31))].Deaths.values)-int(covid[(covid['State/UnionTerritory']=='Madhya Pradesh') & (covid.Date==datetime.date(2020,4,30))].Deaths.values)
mad['May-2021']=int(covid[(covid['State/UnionTerritory']=='Madhya Pradesh') & (covid.Date==datetime.date(2021,5,31))].Deaths.values)-int(covid[(covid['State/UnionTerritory']=='Madhya Pradesh') & (covid.Date==datetime.date(2021,4,30))].Deaths.values)
```

Code snippet 2:

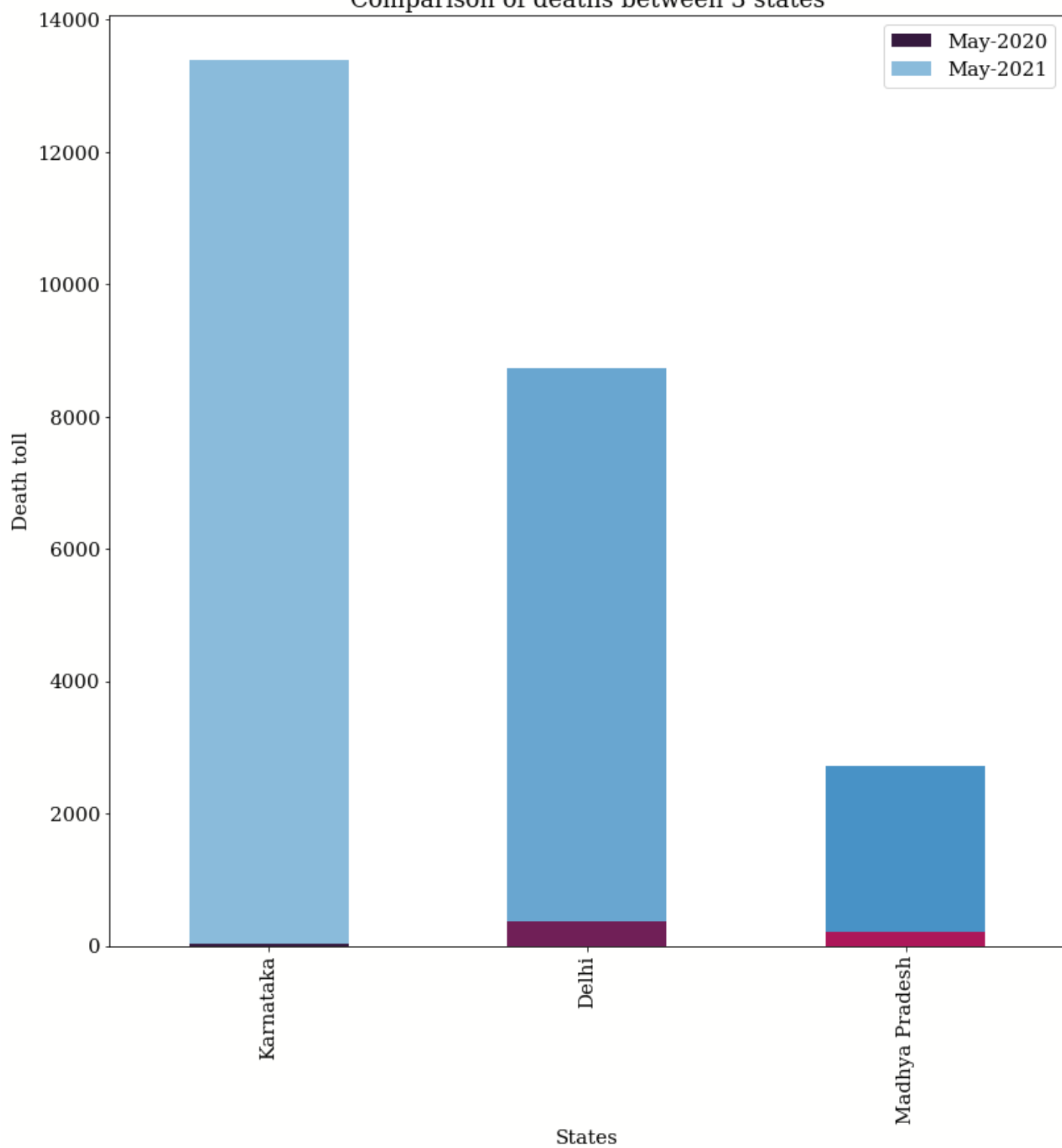
```
q4=pd.DataFrame([karn,delhi,mad])
q4.index=['Karnataka','Delhi','Madhya Pradesh']
q4
```

	May-2020	May-2021
Karnataka	27	13373
Delhi	360	8379
Madhya Pradesh	213	2500

Code snippet 3:

```
q4.plot.bar(stacked=True,xlabel='States',ylabel='Death toll',title='Comparison of death  
s between 3 states',color=[sns.color_palette('rocket'),sns.color_palette('Blues_d')])
```

Comparison of deaths between 3 states



5. Make a graph to show the month wise relation between number of confirmed Covid-19 cases and Deaths in Uttar Pradesh.

Code snippet 1:

```
d,c=list(),list()
for f in range(len(covid)):
    if covid.loc[f]['State/UnionTerritory']=='Uttar Pradesh':
        if covid.loc[f].Date.month in [1,3,5,7,8,10,12]:
            if covid.loc[f].Date.day==31:
                d.append(covid.loc[f].Deaths)
                c.append(covid.loc[f].Confirmed)
        elif covid.loc[f].Date.month==2:
            if covid.loc[f].Date.day==28:
                d.append(covid.loc[f].Deaths)
                c.append(covid.loc[f].Confirmed)
        else:
            if covid.loc[f].Date.day==30:
                d.append(covid.loc[f].Deaths)
                c.append(covid.loc[f].Confirmed)
```

Code snippet 2:

```
c.append(int(covid[(covid['State/UnionTerritory']=='Uttar Pradesh') & (covid['Date']==d
atetime.date(2021,8,11))].Confirmed.values))
d.append(int(covid[(covid['State/UnionTerritory']=='Uttar Pradesh') & (covid['Date']==d
atetime.date(2021,8,11))].Deaths.values))
len(c),len(d)

(18, 18)
```

Code snippet 3:

```
fig=plt.figure(figsize=[10,10])
ax=fig.add_subplot(1,1,1)
months=["March","April","May","June",
```

```
"July","August","September",  
"October","November","December'20",  
"Jan'21","Feb'21","Mar'21","Apr'21",  
"May'21","Jun'21","Jul'21","Aug'21"]  
ax.set_xticklabels(months,rotation='vertical')  
ax.bar(months,c,color=sns.color_palette('rocket_r'))  
ax.bar(months,d,color=sns.color_palette('Greens_d'))  
ax.legend(['Confirmed','Deaths'])  
ax.set_xlabel('Months')  
ax.set_ylabel('Count*10^6')  
ax.set_title('Relation between confirmed and deaths in Uttar Pradesh')
```

