

# Matplotlib of Chhattisgarh State Covid-19 district-wise datasets

## By- Somesh Hariharno

```
%matplotlib inline
```

```
In [2]: import matplotlib as mpl
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
```

```
In [3]: data = pd.read_csv('Covid-19 district wise Dataset.csv')
```

```
In [4]: data.head(10)
#Shows top 10 data from dataset
```

```
Out[4]:
```

	District	Confirmed	Active	Recovered	Deceased
0	Balod	102	10	92	0
1	Baloda Bazar	434	70	362	2
2	Balrampur	233	42	191	0
3	Bametara	130	3	127	0
4	Bastar	282	126	155	1
5	Bijapur	88	30	58	0
6	Bilaspur	745	148	591	6
7	Dakshin Bastar Dantewada	135	6	129	0
8	Dhamtari	46	8	37	1
9	Durg	977	301	667	9

```
In [6]: data.describe()
# Describe the statistics of all the the column
```

Out[6]:

	Confirmed	Active	Recovered	Deceased
<b>count</b>	28.000000	28.000000	28.000000	28.000000
<b>mean</b>	393.000000	101.714286	288.642857	2.642857
<b>std</b>	684.652115	246.453490	436.252154	7.253169
<b>min</b>	32.000000	1.000000	31.000000	0.000000
<b>25%</b>	124.000000	15.250000	89.750000	0.000000
<b>50%</b>	203.500000	33.500000	169.500000	0.500000
<b>75%</b>	323.000000	75.500000	260.000000	2.000000
<b>max</b>	3673.000000	1310.000000	2325.000000	38.000000

In [7]:

```
data.tail(10)
#Shows bottom 10 data from dataset
```

Out[7]:

	District	Confirmed	Active	Recovered	Deceased
<b>18</b>	Mungeli	174	6	168	0
<b>19</b>	Narayanpur	188	17	171	0
<b>20</b>	Raigarh	288	84	202	2
<b>21</b>	Raipur	3673	1310	2325	38
<b>22</b>	Rajnandgaon	817	215	597	5
<b>23</b>	Sukma	79	8	71	0
<b>24</b>	Surajpur	85	34	50	1
<b>25</b>	Surguja	244	20	222	2
<b>26</b>	Uttar Bastar Kanker	254	28	226	0
<b>27</b>	Gaurela Pendra Marwahi	32	1	31	0

In [8]:

```
data.shape
#Shows rows and coloums number
```

Out[8]:

(28, 5)

## Simple Line Plot

In [14]:

```
# Line Plot showing active, confirmed, recovered & deceased cases
```

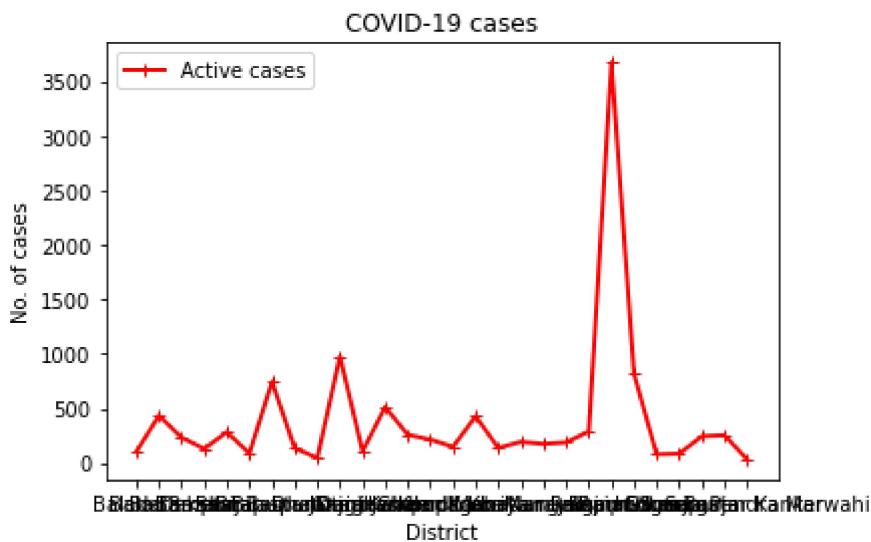
In [19]:

```
A = data.iloc[0:,1].values
R = data.iloc[0:,4].values
D = data.iloc[0:,3].values
C = data.iloc[0:,2].values
X = data.iloc[0:,0]
```

#For designating the data values in a alphabet.

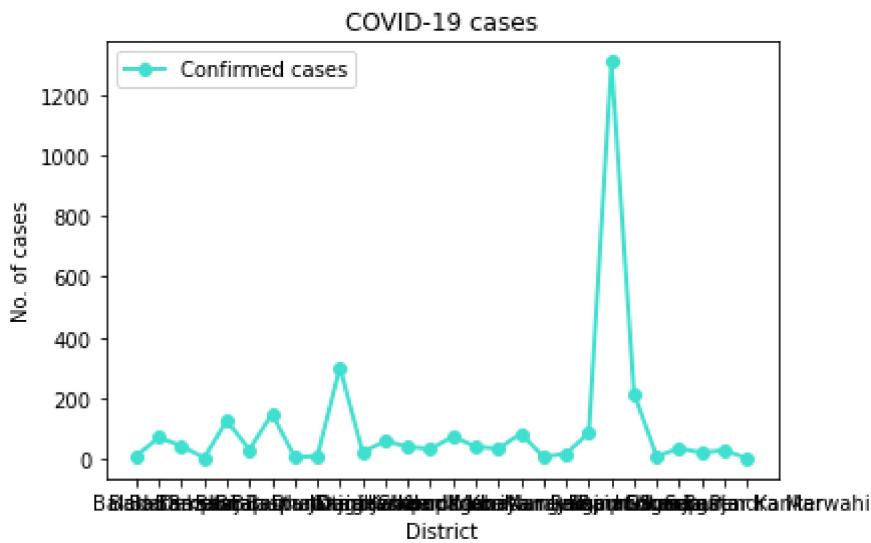
```
In [21]: plt.plot(X,A, label="Active cases", color="red", linewidth=2, marker='+')
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

#Line plot between District(X) and No.of cases(Y) representing Active cases



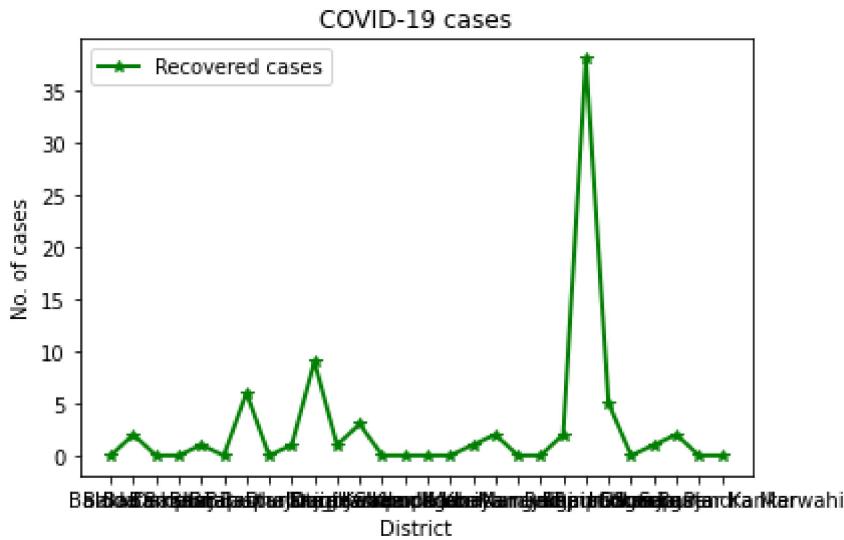
```
In [25]: plt.plot(X,C, label="Confirmed cases", color="turquoise", linewidth=2, marker='o')
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

#Line plot between District(X) and No.of cases(Y) representing Confirmed cases



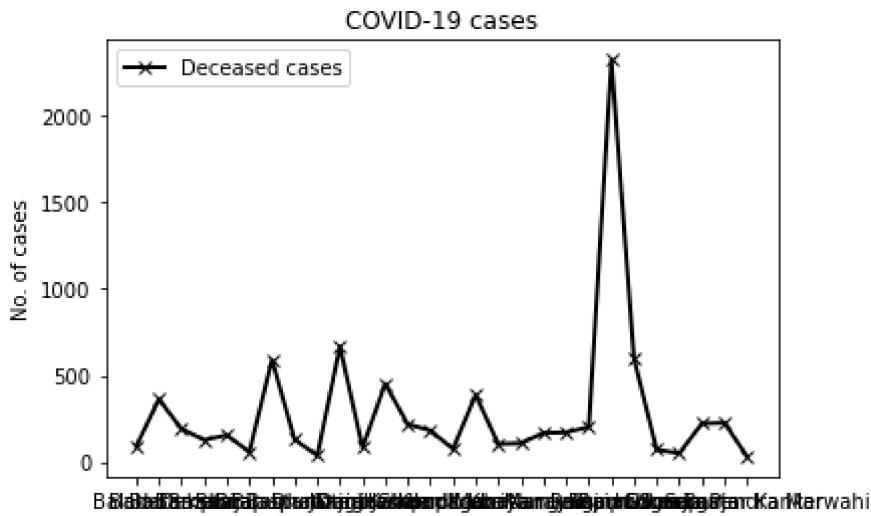
```
In [32]: plt.plot(X,R, label="Recovered cases", color="green", linewidth=2, marker='*')
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

#Line plot between District(X) and No.of cases(Y) representing Recovered cases



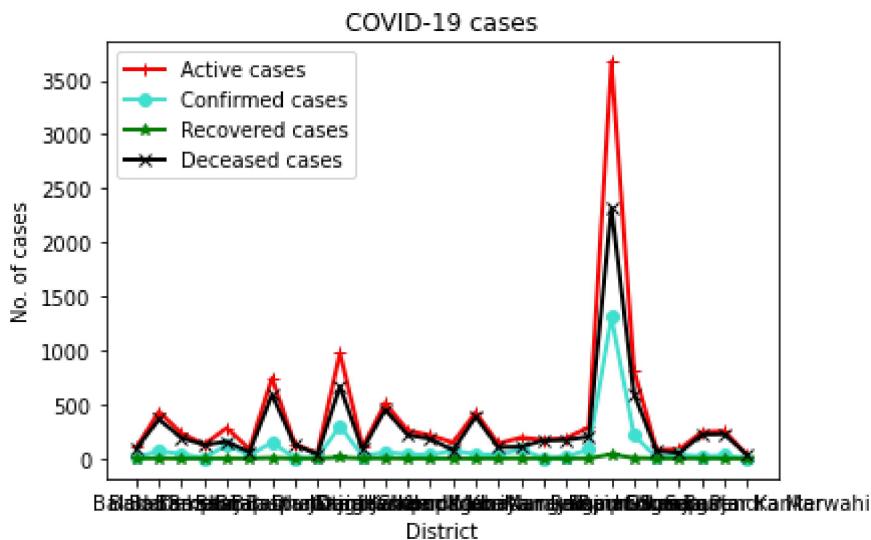
```
In [33]: plt.plot(X,D, label="Deceased cases", color="black", linewidth=2, marker='x')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

#Line plot between District(X) and No.of cases(Y) representing Deceased cases



```
In [39]: plt.plot(X,A, label="Active cases", color="red", linewidth=2, marker='+')
plt.plot(X,C, label="Confirmed cases", color="turquoise", linewidth=2, marker='o')
plt.plot(X,R, label="Recovered cases", color="green", linewidth=2, marker='*')
plt.plot(X,D, label="Deceased cases", color="black", linewidth=2, marker='x')
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

#Single line plot between District(X) and No.of cases(Y) representing all active, conf

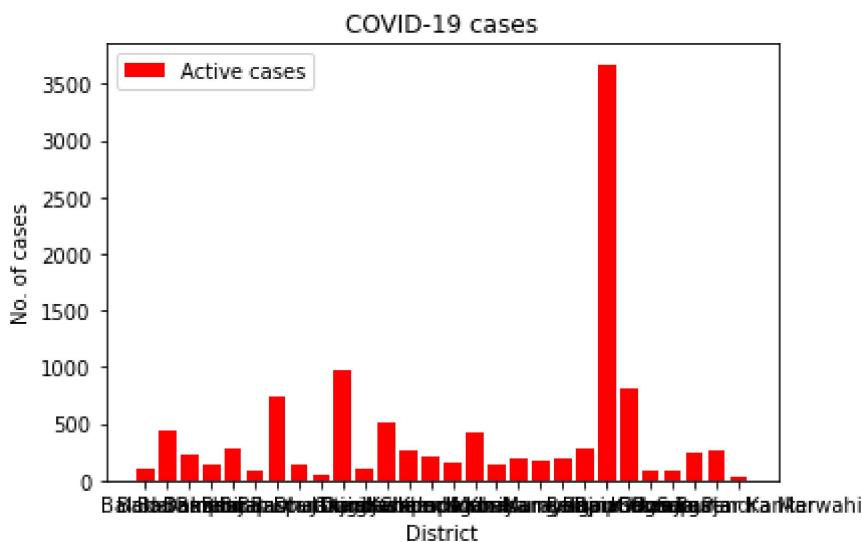


## Bar Graph plot

```
In [37]: #Bar Plot diagram showing active, confirmed, recovered & deceased cases
```

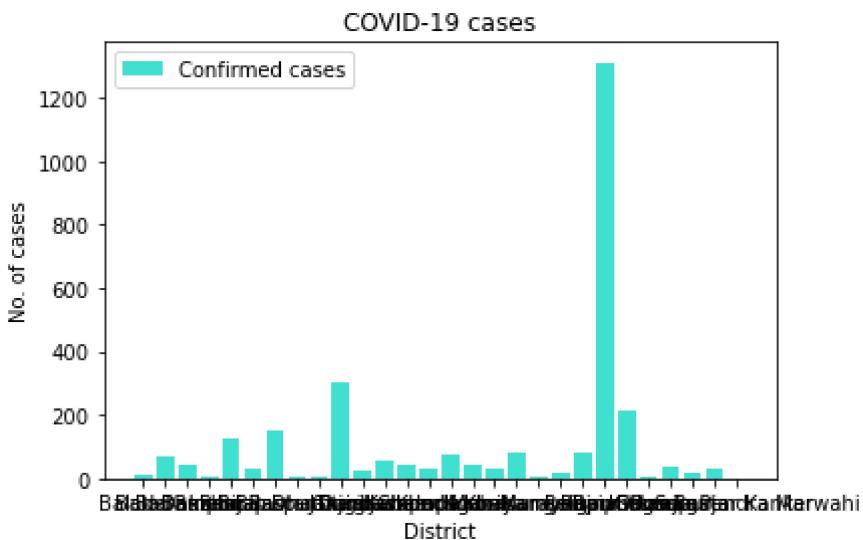
```
In [40]: plt.bar(X, A, label="Active cases", color="red")
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

#Bar plot between District(X) and No.of cases(Y) representing Active cases



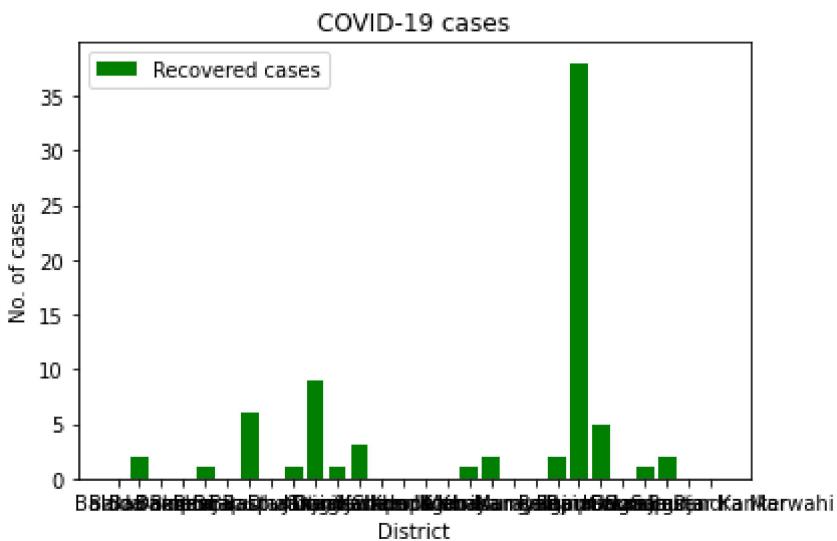
```
In [43]: plt.bar(X, C, label="Confirmed cases", color="turquoise")
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

#Bar plot between District(X) and No. of cases(Y) representing Confirmed cases



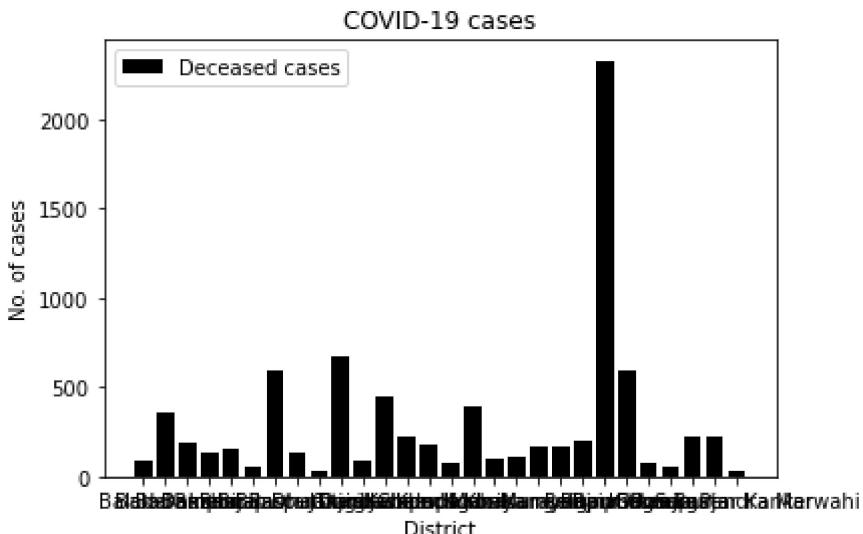
```
In [45]: plt.bar(X, R, label="Recovered cases", color="green")
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

#Bar plot between District(X) and No.of cases(Y) representing Recovered cases



```
In [53]: plt.bar(X, D, label="Deceased cases", color="black")
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
plt.show()
```

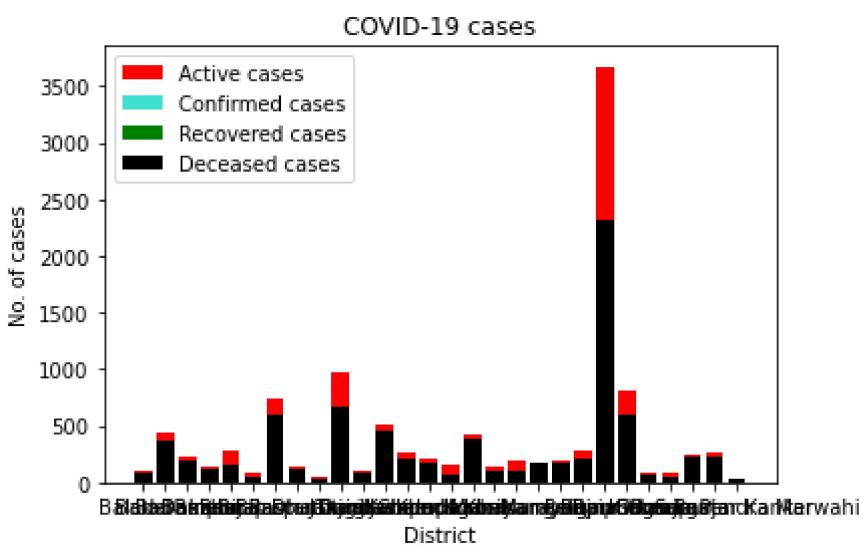
#Bar plot between District(X) and No.of cases(Y) representing Deceased cases



```
In [52]: plt.bar(X, A, label="Active cases", color="red")
plt.bar(X, C, label="Confirmed cases", color="turquoise")
plt.bar(X, R, label="Recovered cases", color="green")
plt.bar(X, D, label="Deceased cases", color="black")
plt.xlabel('District')
plt.ylabel('No. of cases')
plt.title('COVID-19 cases')
plt.legend()
```

#Single Bar plot between District(X) and No.of cases(Y) representing all active, confi

```
Out[52]: <matplotlib.legend.Legend at 0x243d3237e80>
```



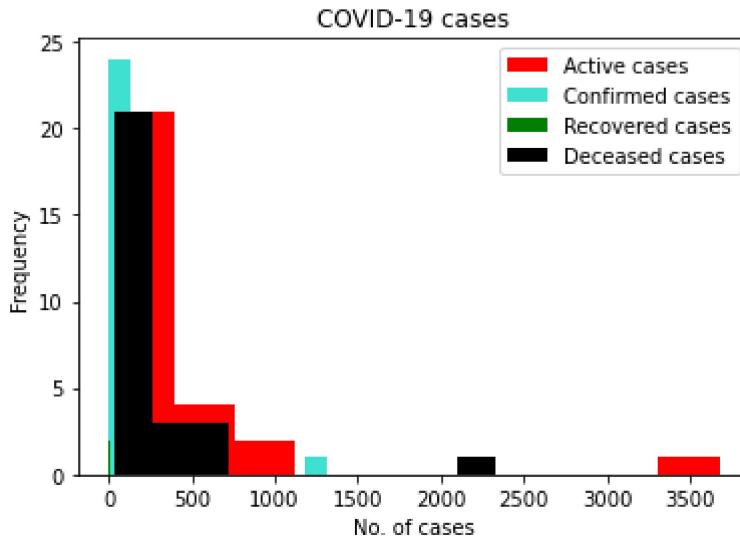
# Histogram

```
In [55]: #Histogram showing active, confirmed, recovered & deceased cases
```

```
In [57]: plt.hist(A, label="Active cases", color="red")
plt.hist(C, label="Confirmed cases", color="turquoise")
plt.hist(R, label="Recovered cases", color="green")
plt.hist(D, label="Deceased cases", color="black")
plt.xlabel("No. of cases")
```

```
plt.ylabel("Frequency")
plt.title("COVID-19 cases")
plt.legend()
plt.show()
```

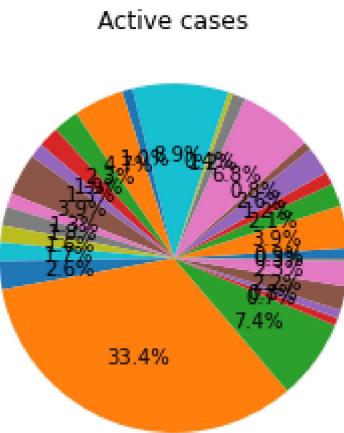
#Histogram between District(X) and No.of cases(Y) representing all active, confirmed,



## Pie Chart

```
In [59]: plt.pie(A, autopct='%1.1f%%')
plt.title('Active cases')
plt.show()
```

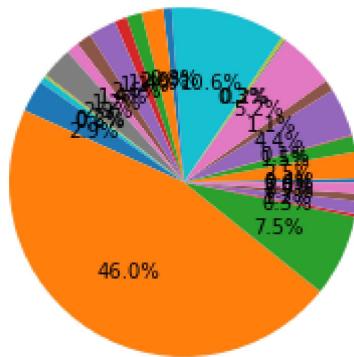
#Pie chart showing Active cases



```
In [60]: plt.pie(C, autopct='%1.1f%%')
plt.title('Confirmed cases')
plt.show()
```

#Pie chart showing Confirmed cases

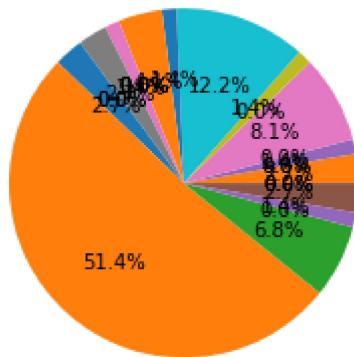
### Confirmed cases



```
In [61]: plt.pie(R, autopct='%1.1f%%')
plt.title('Recovered cases')
plt.show()
```

*#Pie chart showing Recovered cases*

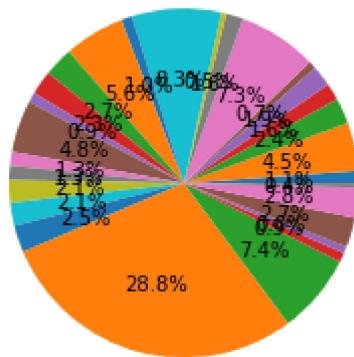
### Recovered cases



```
In [62]: plt.pie(D, autopct='%1.1f%%')
plt.title('Deceased cases')
plt.show()
```

*#Pie chart showing Deceased cases*

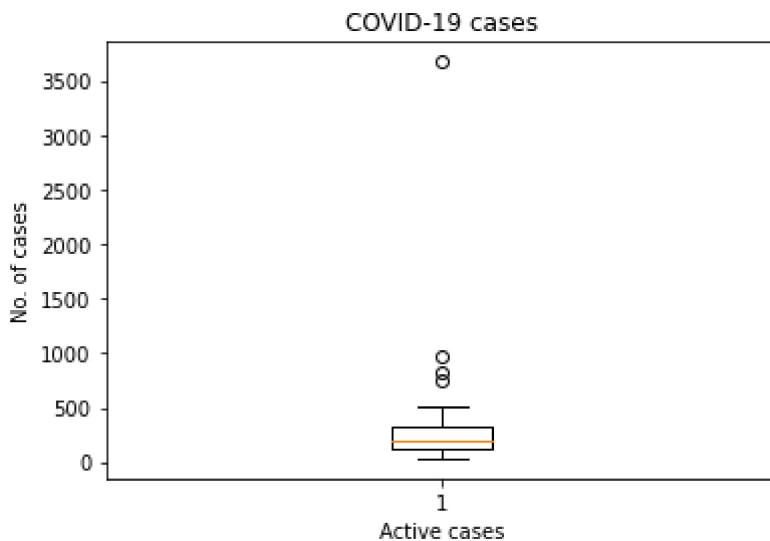
### Deceased cases



## Box Plot

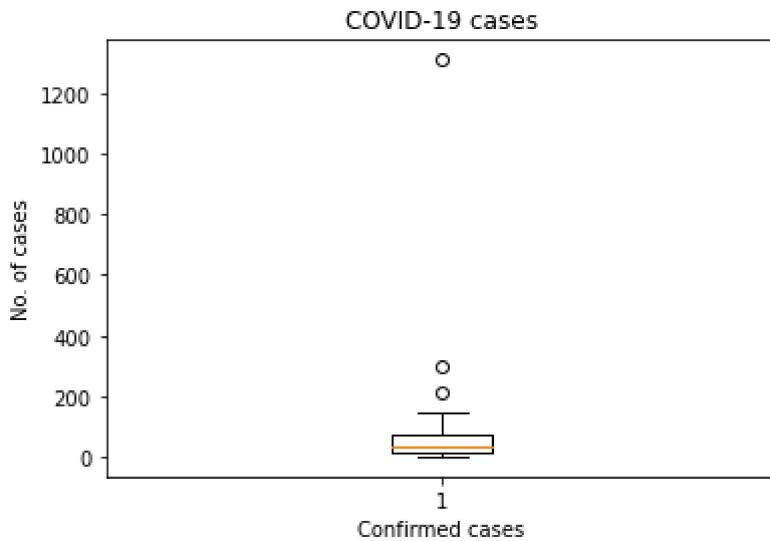
```
In [72]: plt.boxplot(A)
plt.title('COVID-19 cases')
plt.xlabel('Active cases')
plt.ylabel('No. of cases')
plt.show()
```

#Box plot between Active cases(X) and No.of cases(Y)



```
In [75]: plt.boxplot(C)
plt.title('COVID-19 cases')
plt.xlabel('Confirmed cases')
plt.ylabel('No. of cases')
plt.show()
```

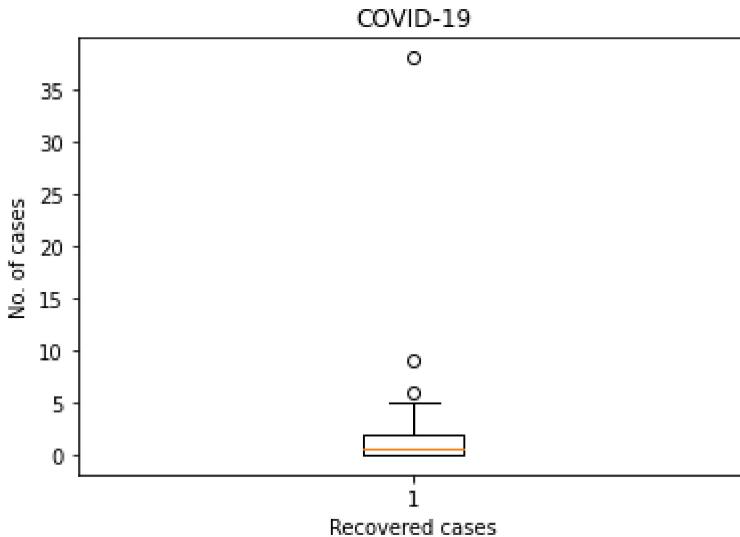
#Box plot between Confirmed cases(X) and No.of cases(Y)



```
In [76]: plt.boxplot(R)
plt.title('COVID-19')
plt.xlabel('Recovered cases')
plt.ylabel('No. of cases')
```

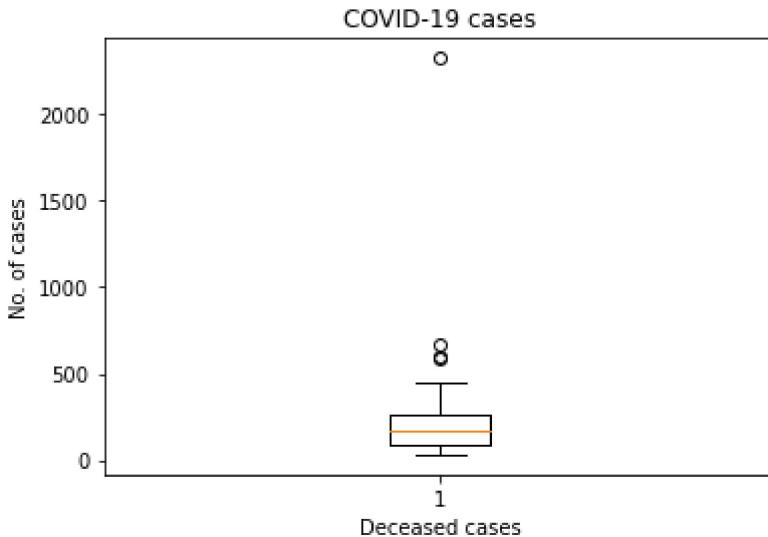
```
plt.show()
```

#Box pLot between Recovered cases(X) and No.of cases(Y)



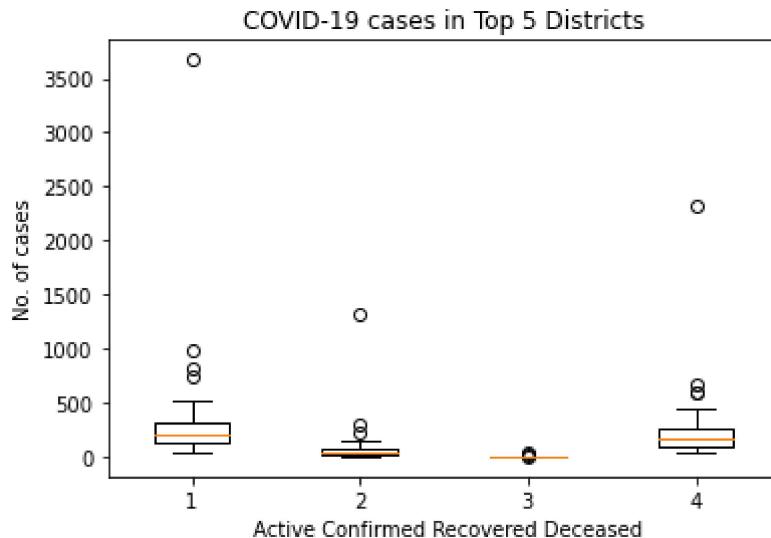
```
In [78]: plt.boxplot(D)
plt.title('COVID-19 cases')
plt.xlabel('Deceased cases')
plt.ylabel('No. of cases')
plt.show()
```

#Box plot between Deceased cases(X) and No.of cases(Y)



```
In [89]: COVID_cases = [A,C,R,D]
plt.boxplot(COVID_cases)
plt.title('COVID-19 cases')
plt.title('COVID-19 cases in Top 5 Districts')
plt.xlabel(' Active Confirmed Recovered Deceased')
plt.ylabel('No. of cases')
plt.show()
```

#Box plot showing COVID-19 cases in Top 5 Districts between Active, Confirmed, Recover



```
In [90]: #Plot active vs recovered cases for top 5 districts having highest number of cases
```

```
In [100... data.sort_values(['Active','District'], ascending = False)
```

```
#Sort all the Active data with district in descending order
```

Out[100]:

	District	Confirmed	Active	Recovered	Deceased
21	Raipur	3673	1310	2325	38
9	Durg	977	301	667	9
22	Rajnandgaon	817	215	597	5
6	Bilaspur	745	148	591	6
4	Bastar	282	126	155	1
20	Raigarh	288	84	202	2
17	Mahasamund	193	83	108	2
14	Kondagaon	148	73	75	0
1	Baloda Bazar	434	70	362	2
11	Janjgir Champa	513	58	452	3
2	Balrampur	233	42	191	0
15	Korba	428	40	388	0
12	Jashpur	258	40	218	0
24	Surajpur	85	34	50	1
16	Koriya	138	33	104	1
13	Kabeerdham	214	32	182	0
5	Bijapur	88	30	58	0
26	Uttar Bastar Kanker	254	28	226	0
10	Gariaband	106	22	83	1
25	Surguja	244	20	222	2
19	Narayanpur	188	17	171	0
0	Balod	102	10	92	0
23	Sukma	79	8	71	0
8	Dhamtari	46	8	37	1
18	Mungeli	174	6	168	0
7	Dakshin Bastar Dantewada	135	6	129	0
3	Bametara	130	3	127	0
27	Gaurela Pendra Marwahi	32	1	31	0

In [96]: `sortcases= data.sort_values(['Active','District'], ascending = False)  
sortcases.head(5)`

#Sort 5 Active data with district in descending order

Out[96]:

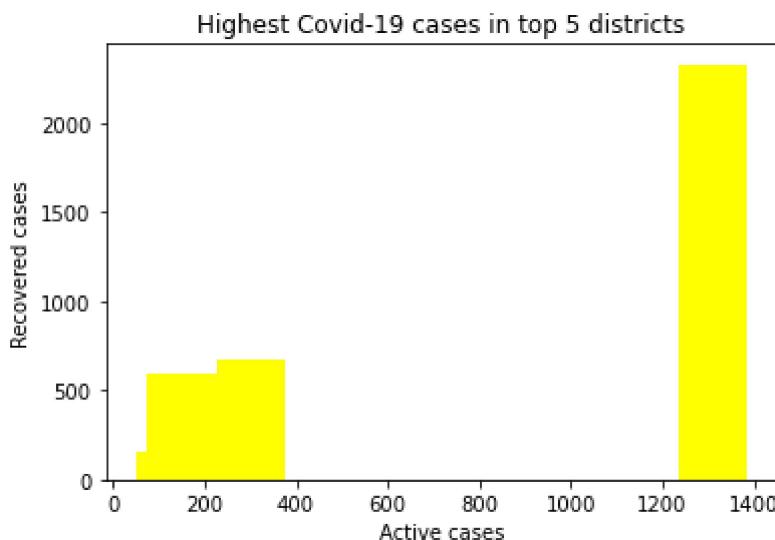
	District	Confirmed	Active	Recovered	Deceased
21	Raipur	3673	1310	2325	38
9	Durg	977	301	667	9
22	Rajnandgaon	817	215	597	5
6	Bilaspur	745	148	591	6
4	Bastar	282	126	155	1

In [104...]

```
highestcases = sortcases.head(5)

a = highestcases.loc[:, "Active"]
r = highestcases.loc[:, "Recovered"]
plt.bar(a,r, width = 150, color='yellow')
plt.xlabel("Active cases")
plt.ylabel("Recovered cases")
plt.title("Highest Covid-19 cases in top 5 districts")
plt.show()
```

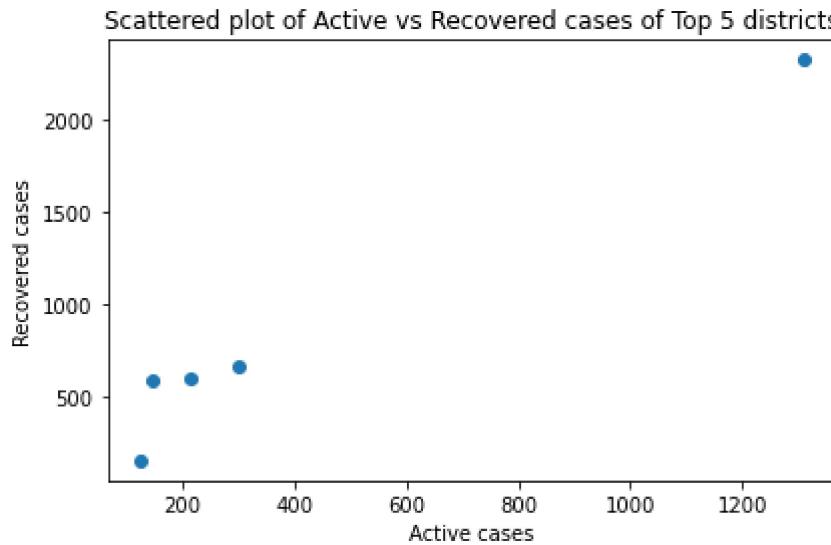
#Bar plot between Active vs Recovered cases of top 5 highest active COVID-19 cases



In [105...]

```
plt.scatter(a,r)
plt.xlabel("Active cases")
plt.ylabel("Recovered cases")
plt.title("Scattered plot of Active vs Recovered cases of Top 5 districts")
plt.tight_layout()
plt.show()
```

#Scatter plot between Active vs Recovered cases of top 5 highest active COVID-19 cases



```
In [125]:  
labels= ['Raipur', 'Durg', 'Rajnandgaon', 'Bilaspur', 'Bastar']  
colors = ['red', 'crimson', 'palevioletred', 'lightpink', 'mistyrose']  
plt.pie(a, labels=labels, colors=colors, autopct='%1.1f%')  
plt.title('Active cases in Top 5 Districts')  
plt.show()  
  
#Pie chart of top 5 highest active COVID-19 cases district
```

Active cases in Top 5 Districts

