```
import pandas as pd
import seaborn as sns # informative statistical graphics.
import statsmodels.api as sm #for ARIMA and SARIMAX
import datetime
from datetime import timedelta

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

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-2-4ce35ee21619> in <cell line: 7>()
      5 from datetime import timedelta
      6
----> 7 df = pd.read_csv('/content/covid_19_india.csv')

                                 6 frames
/usr/local/lib/python3.10/dist-packages/pandas/io/common.py in get_handle(path_or_buf, mode, encoding, compression, memory_map,
is_text, errors, storage_options)
    854         if ioargs.encoding and "b" not in ioargs.mode:
    855             # Encoding
--> 856             handle = open(
    857                 handle,
    858                 ioargs.mode,

FileNotFoundError: [Errno 2] No such file or directory: '/content/covid_19_india.csv'
```

SEARCH STACK OVERFLOW

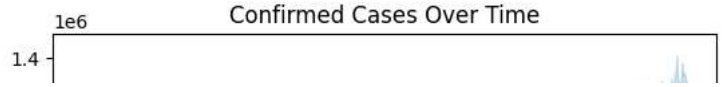
Double-click (or enter) to edit

```
df= df.drop(labels = ["Sno","State","Time","Cured","Deaths"], axis= 1, inplace= False)
```

```
df.head()
```

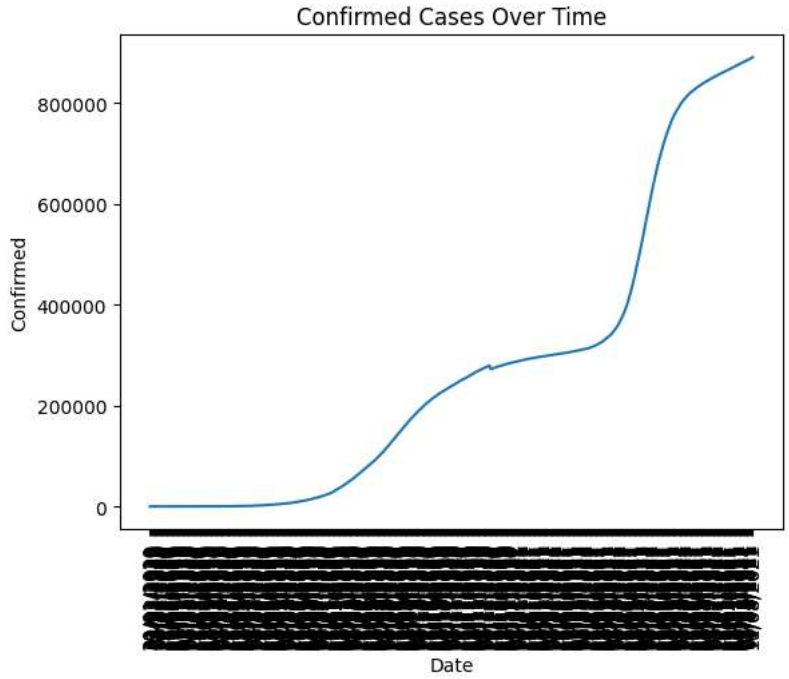
	Date	Confirmed
0	30/01/2020	1
1	31/01/2020	1
2	01/02/2020	2
3	02/02/2020	3
4	03/02/2020	3

```
import matplotlib.pyplot as plt
sns.lineplot(x="Date", y="Confirmed",legend = 'full' , data=df)
plt.title("Confirmed Cases Over Time")
plt.xticks(rotation=45)
plt.show()
```

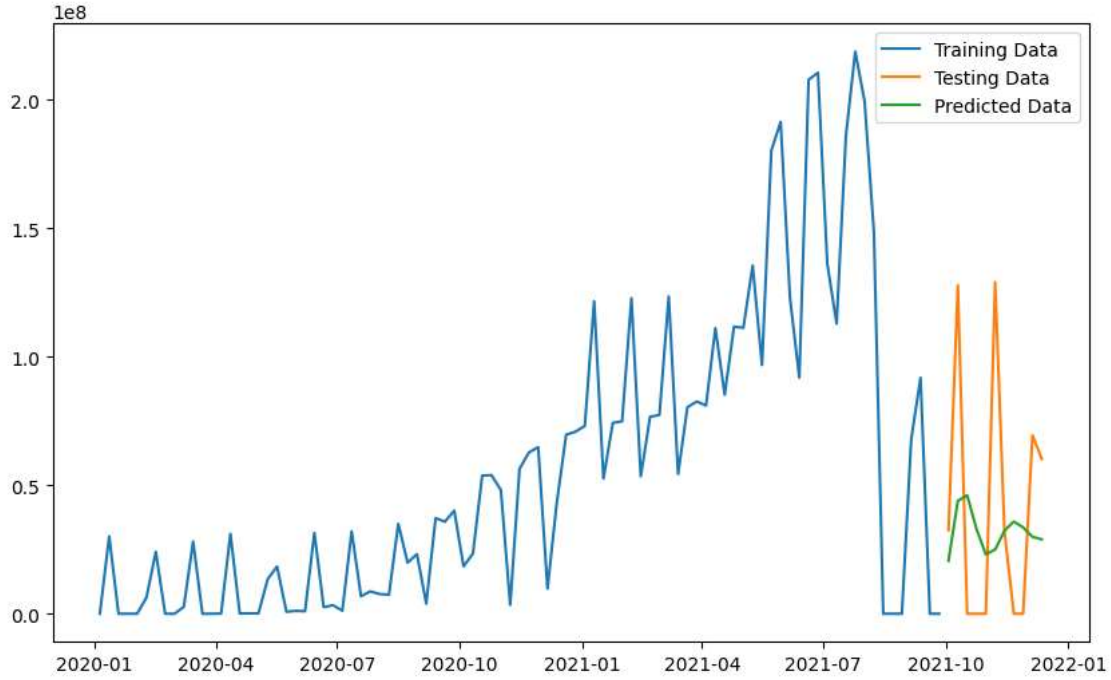



```
import matplotlib.pyplot as plt
sns.lineplot(x="Date", y="Confirmed",legend = 'full' , data=df, ci=None)
plt.title("Confirmed Cases Over Time")
plt.xticks(rotation=90)
plt.show()
```

```
<ipython-input-5-af2afa873440>:2: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

  sns.lineplot(x="Date", y="Confirmed",legend = 'full' , data=df, ci=None)
```




```
df['Date'] = pd.to_datetime(df['Date'])
df.set_index('Date', inplace=True)


df = df.resample('W').sum()


train_data = df[:int(0.9*(len(df)))]
test_data = df[int(0.9*(len(df))):]


import statsmodels.api as sm


model = sm.tsa.arima.ARIMA(train_data, order=(2,1,2))
model_fit = model.fit()


predictions = model_fit.predict(start=len(train_data), end=len(train_data)+len(test_data)-1, typ='levels')


plt.figure(figsize=(10,6))
plt.plot(train_data, label='Training Data')
plt.plot(test_data, label='Testing Data')
plt.plot(predictions, label='Predicted Data')
plt.legend()
plt.show()
```



Colab paid products - Cancel contracts here