Project Title : Covid-19 using cognos

Team members :

Y.SHANKAR

D.PAYEELAVAN

S.SUVENTHAN

PHASE 4 : DEVELOPMENT PART 2

INTRODUCTION:

The project involves analyzing COVID-19 cases and deaths data using IBM Cognos. The objective is to compare and contrast the mean values and standard deviations of cases and associated deaths per day and by country in the EU/EEA. This project encompasses defining analysis objectives, collecting COVID-19 data, designing relevant visualizations in IBM Cognos, and deriving insights from the data.

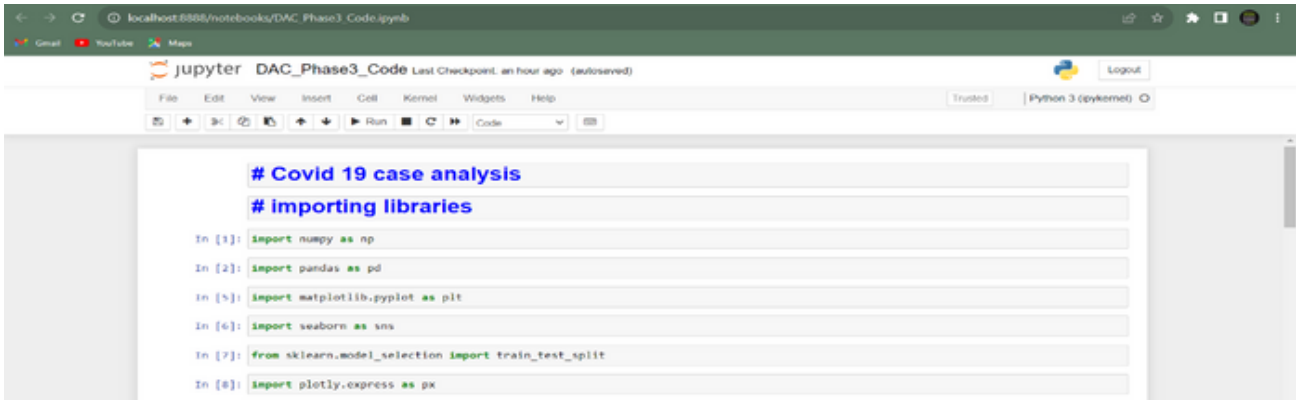
Data Collection and Preprocessing

Collect Covid -19 data which include date,month,year,cases,death,countries and territories and any other relevant data.

- Preprocess the data by handling missing values, encoding categorical variables, and scaling numerical features.

- Split the data into training and testing sets.

- **Importing Libraries**



- **Importing COVID-19 Case DataSet**

DataSet - https://www.kaggle.com/datasets/chakradharmattapalli/covid-19-cases

```
# importing covid-19 case Dataset
```

```
In [19]: cd_data= pd.read_csv ("Downloads/Covid_19_cases4.csv")
```

```
In [21]: cd_data
```

Out[21]:

	dateRep	day	month	year	cases	deaths	countriesAndTerritories
0	31-05-2021	31	5	2021	366	5	Austria
1	30-05-2021	30	5	2021	570	6	Austria
2	29-05-2021	29	5	2021	538	11	Austria
3	28-05-2021	28	5	2021	639	4	Austria
4	27-05-2021	27	5	2021	405	19	Austria
...	...	...	...	...	...	...	...
2725	06-03-2021	6	3	2021	3455	17	Sweden
2726	05-03-2021	5	3	2021	4069	12	Sweden
2727	04-03-2021	4	3	2021	4884	14	Sweden
2728	03-03-2021	3	3	2021	4876	19	Sweden
2729	02-03-2021	2	3	2021	6191	19	Sweden

2730 rows × 7 columns

Data Preprocessing

- **Head , Tail and Shape of the data**

```
In [22]: cd_data.head()
```

Out[22]:

	dateRep	day	month	year	cases	deaths	countriesAndTerritories
0	31-05-2021	31	5	2021	366	5	Austria
1	30-05-2021	30	5	2021	570	6	Austria
2	29-05-2021	29	5	2021	538	11	Austria
3	28-05-2021	28	5	2021	639	4	Austria
4	27-05-2021	27	5	2021	405	19	Austria

```
In [23]: cd_data.tail()
```

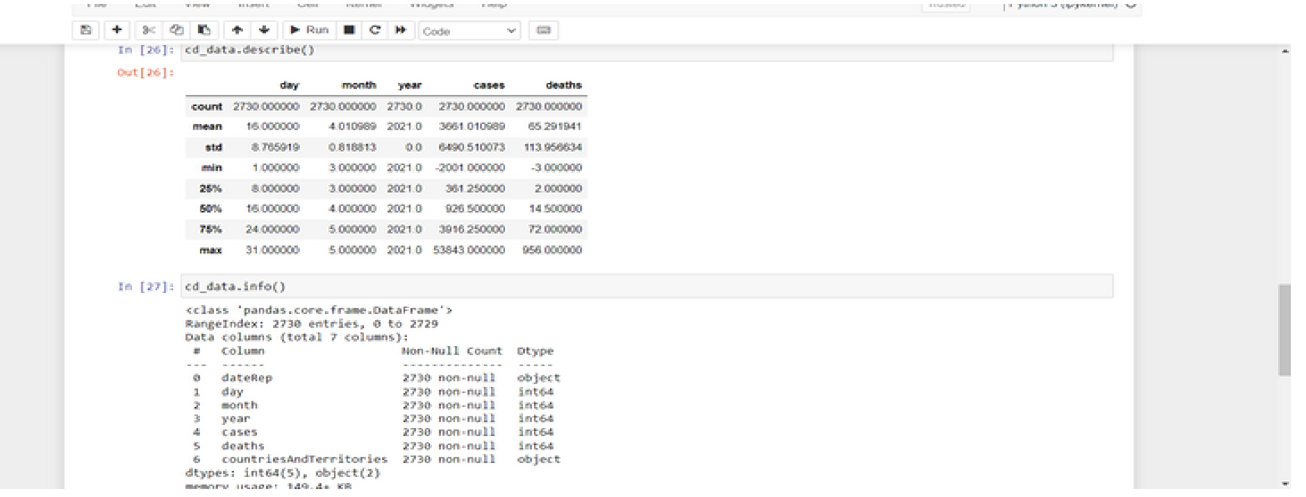
Out[23]:

	dateRep	day	month	year	cases	deaths	countriesAndTerritories
2725	06-03-2021	6	3	2021	3455	17	Sweden
2726	05-03-2021	5	3	2021	4069	12	Sweden
2727	04-03-2021	4	3	2021	4884	14	Sweden
2728	03-03-2021	3	3	2021	4876	19	Sweden
2729	02-03-2021	2	3	2021	6191	19	Sweden

```
In [24]: cd_data.shape
```

```
Out[24]: (2730, 7)
```

- **Describe and Information of the data**



In [26]: cd_data.describe()

Out[26]:

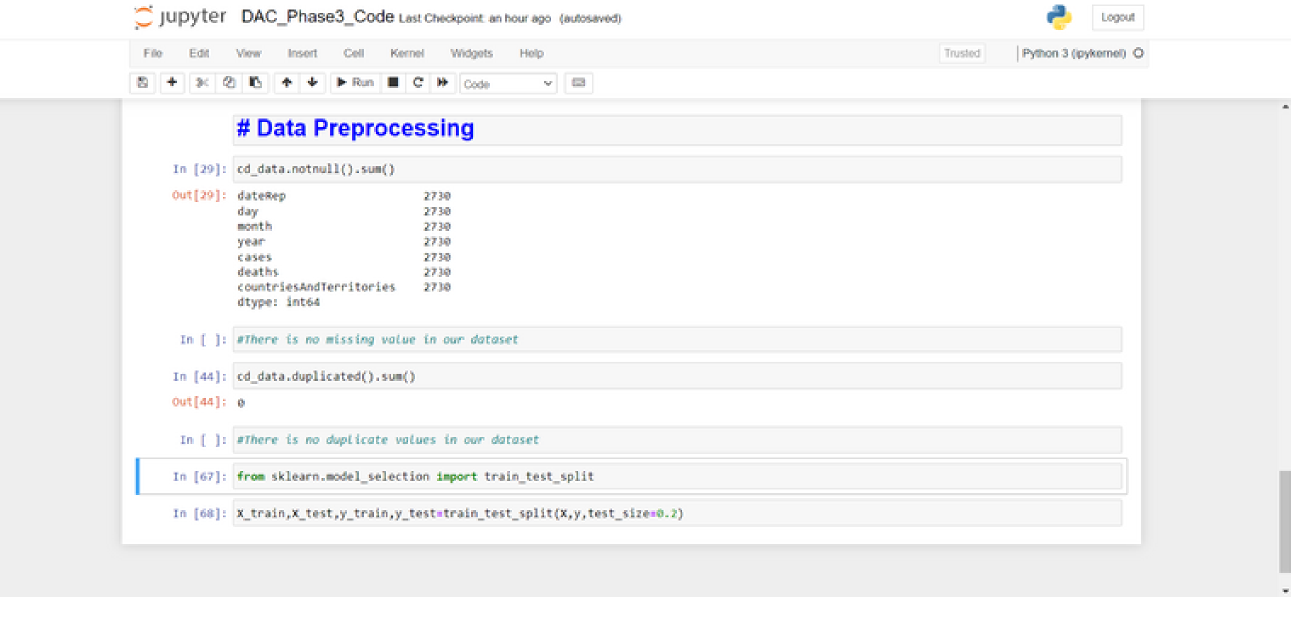
	day	month	year	cases	deaths
count	2730.000000	2730.000000	2730.0	2730.000000	2730.000000
mean	16.000000	4.010989	2021.0	3661.010989	65.291941
std	8.765919	0.818813	0.0	6490.510073	113.956634
min	1.000000	3.000000	2021.0	-2001.000000	-3.000000
25%	8.000000	3.000000	2021.0	361.250000	2.000000
50%	16.000000	4.000000	2021.0	926.500000	14.500000
75%	24.000000	5.000000	2021.0	3916.250000	72.000000
max	31.000000	5.000000	2021.0	53843.000000	956.000000

```
In [27]: cd_data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2730 entries, 0 to 2729
Data columns (total 7 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   dateRep                 2730 non-null   object
 1   day                     2730 non-null   int64
 2   month                   2730 non-null   int64
 3   year                    2730 non-null   int64
 4   cases                   2730 non-null   int64
 5   deaths                  2730 non-null   int64
 6   countriesAndTerritories 2730 non-null   object
dtypes: int64(5), object(2)
```

- **Null Values and Duplicates**

The dataset does not contain duplicates and missing values.

The data are split into **train and test dataset** for further development.



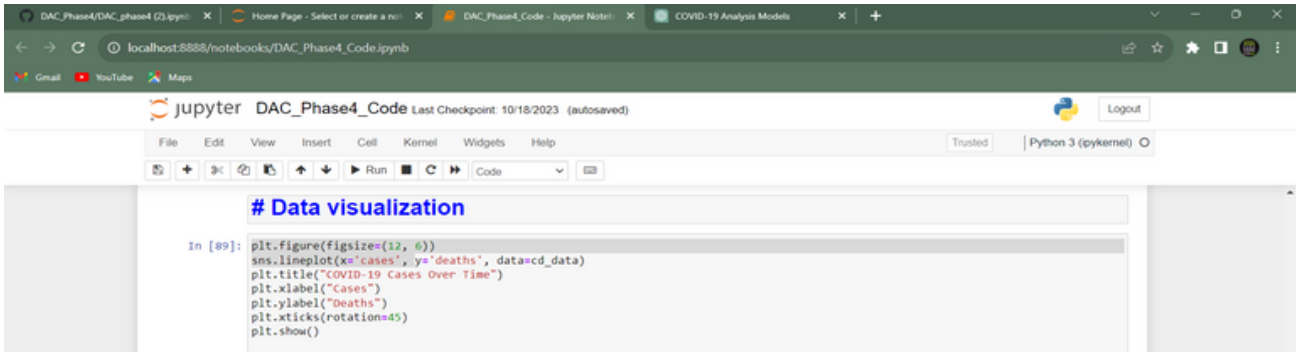
jupyter DAC_Phase3_Code Last Checkpoint: an hour ago (autosaved)

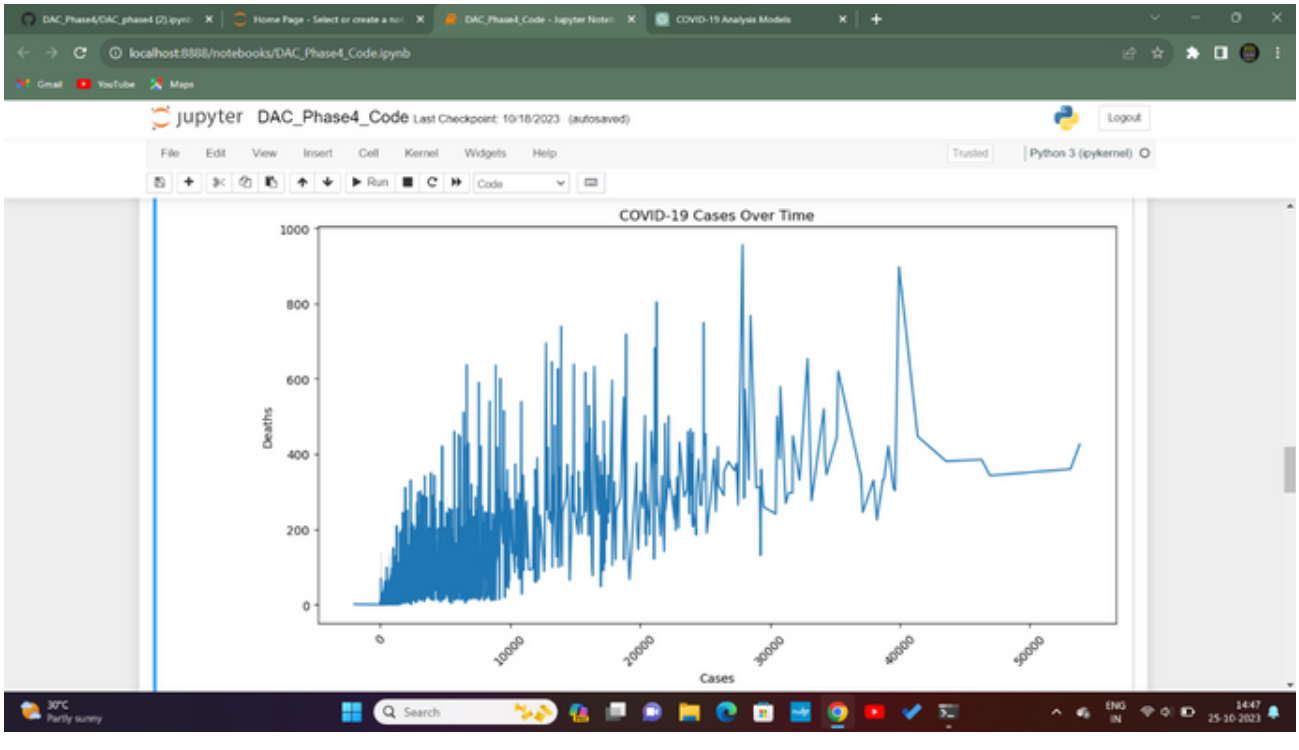
Data Preprocessing

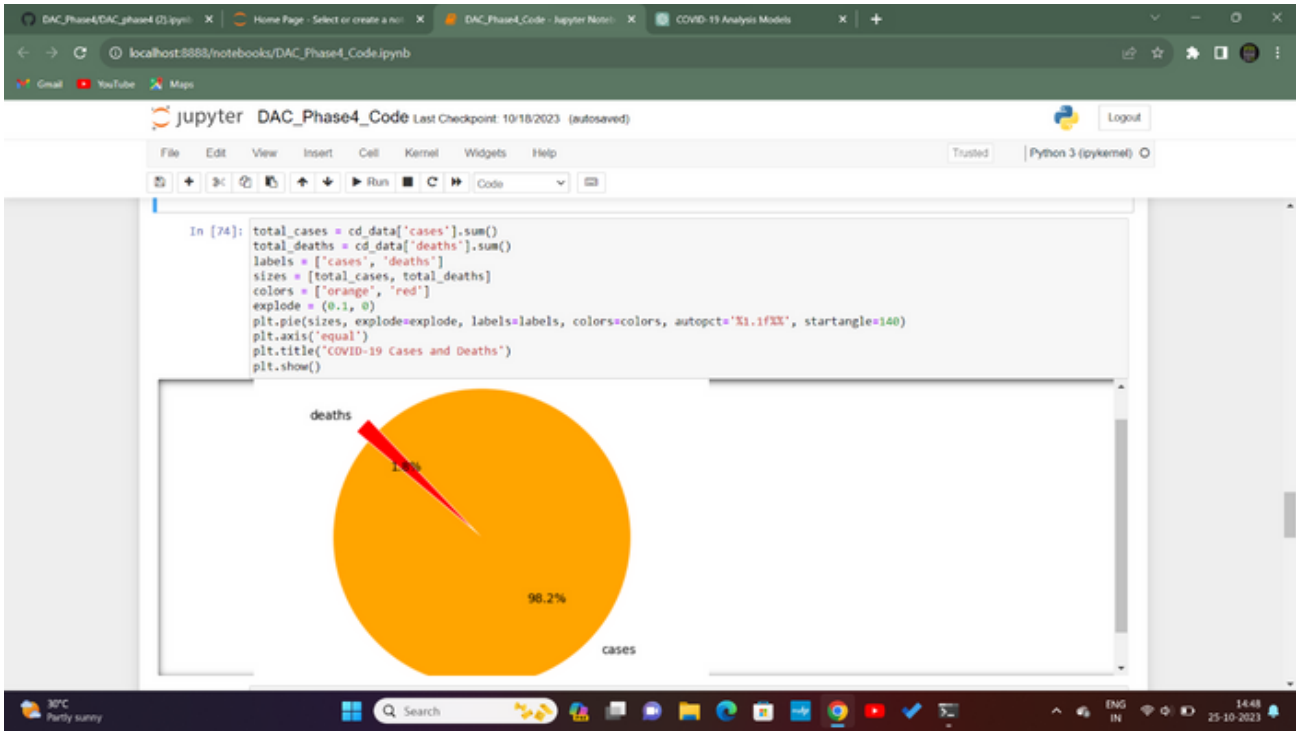
```
In [29]: cd_data.notnull().sum()
Out[29]: dateRep                    2730
         day                        2730
         month                      2730
         year                       2730
         cases                      2730
         deaths                     2730
         countriesAndTerritories    2730
         dtype: int64

In [ ]: #There is no missing value in our dataset

In [44]: cd_data.duplicated().sum()
Out[44]: 0

In [ ]: #There is no duplicate values in our dataset

In [67]: from sklearn.model_selection import train_test_split

In [68]: x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2)
```

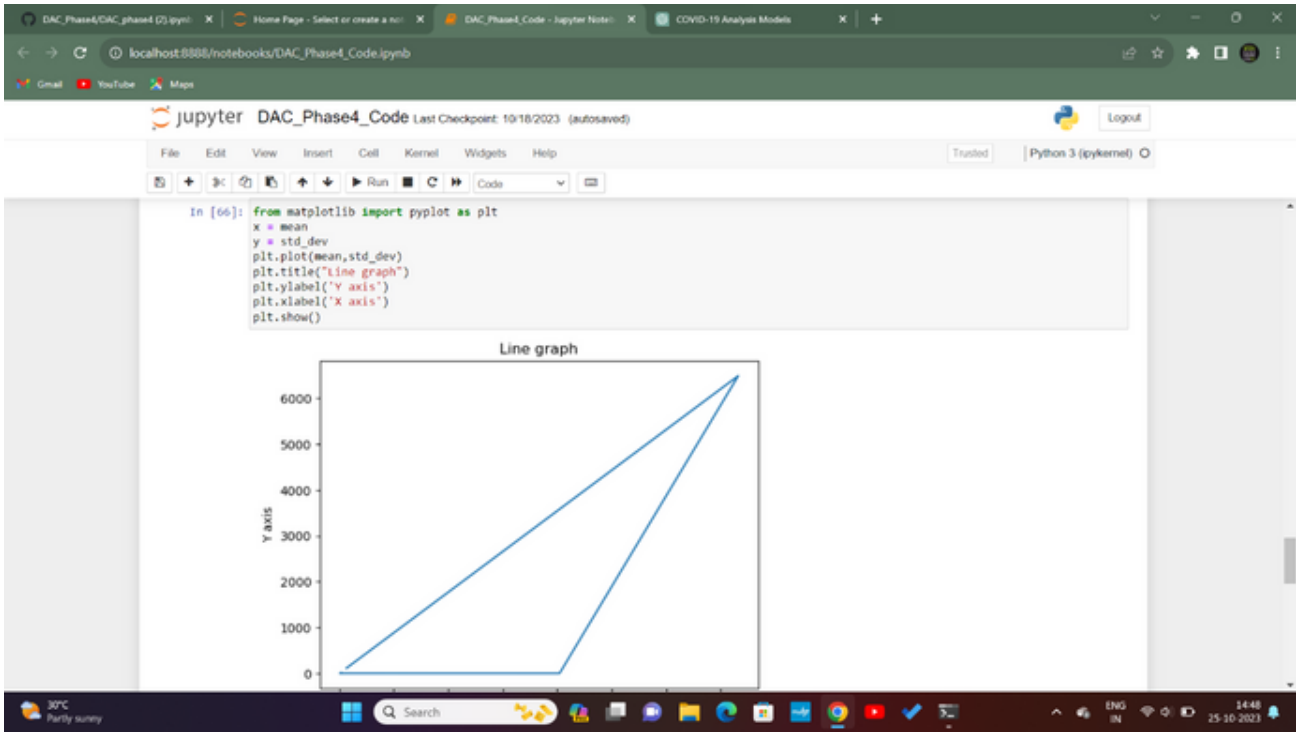
- **Data Visualization**

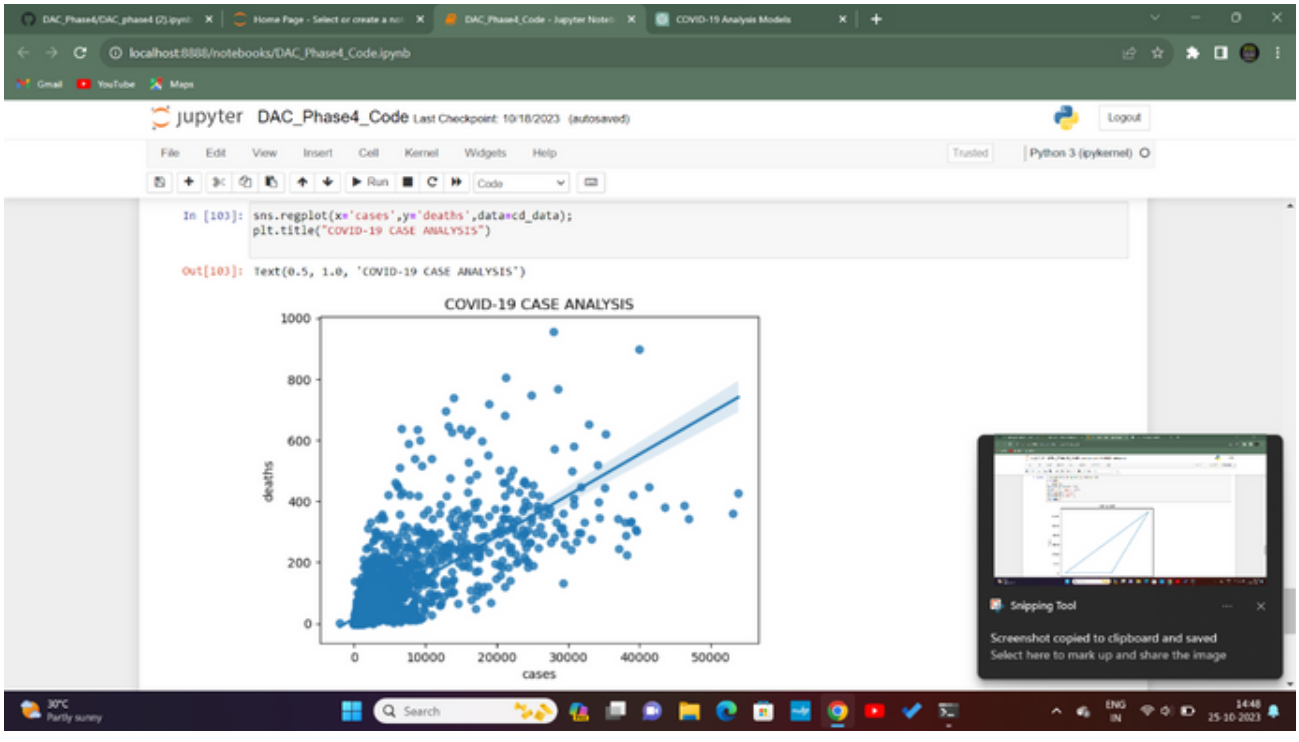
Creating a data visualization for COVID-19 case analysis typically involves plotting various aspects of the data to provide insights into the spread of the virus.

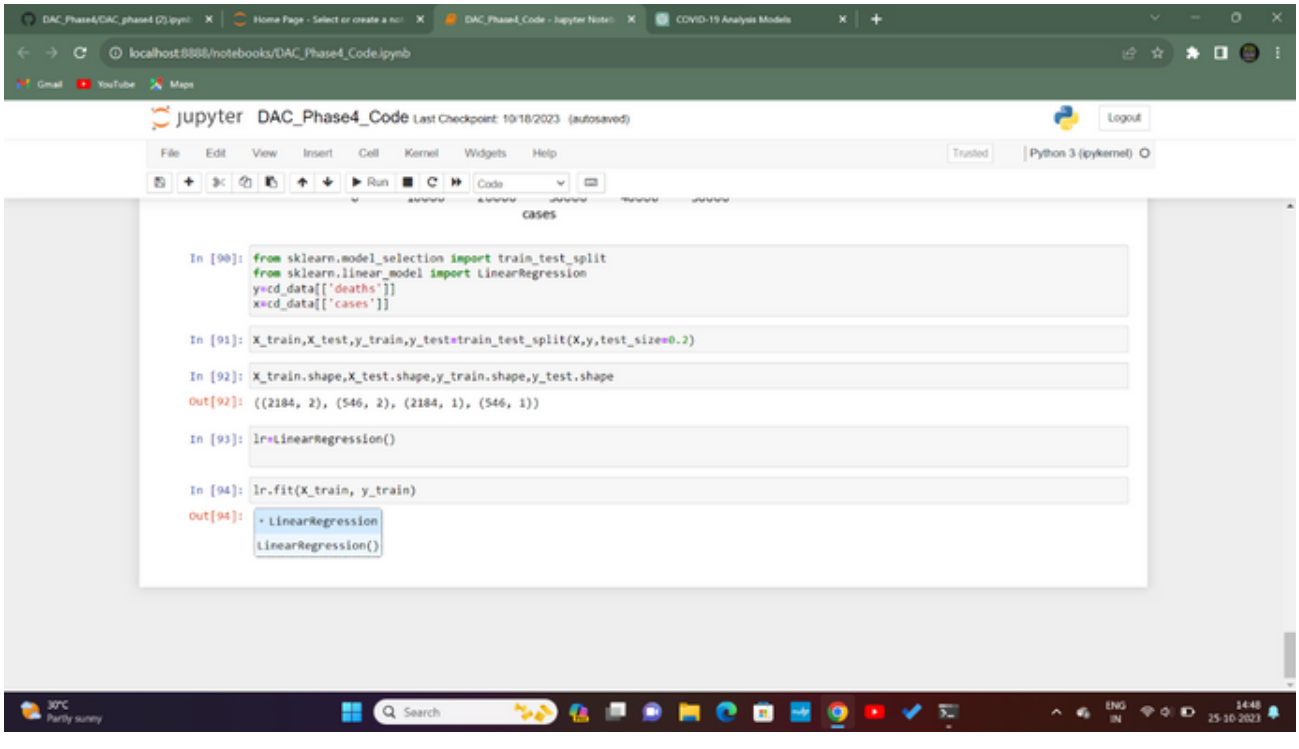





jupyter DAC_Phase4_Code Last Checkpoint: 10/18/2023 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
In [74]:
total_cases = cd_data['cases'].sum()
total_deaths = cd_data['deaths'].sum()
labels = ['cases', 'deaths']
sizes = [total_cases, total_deaths]
colors = ['orange', 'red']
explode = (0.1, 0)
plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=140)
plt.axis('equal')
plt.title('COVID-19 Cases and Deaths')
plt.show()
deaths
1.8%
98.2%
cases


jupyter DAC_Phase4_Code Last Checkpoint: 10/18/2023 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
In [66]:
from matplotlib import pyplot as plt
x = mean
y = std_dev
plt.plot(mean,std_dev)
plt.title("Line graph")
plt.ylabel('Y axis')
plt.xlabel('X axis')
plt.show()
Line graph
Y axis
6000
5000
4000
3000
2000
1000
0


DAC_Phase4_Code - Jupyter Notebook
COVID-19 Analysis Models
localhost:8888/notebooks/DAC_Phase4_Code.ipynb
jupyter DAC_Phase4_Code Last Checkpoint: 10/18/2023 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
In [103]:
sns.regplot(x='cases',y='deaths',data=cd_data);
plt.title("COVID-19 CASE ANALYSIS")
Out[103]:
Text(0.5, 1.0, 'COVID-19 CASE ANALYSIS')
COVID-19 CASE ANALYSIS
deaths
cases
0
200
400
600
800
1000
0
10000
20000
30000
40000
50000
Snipping Tool
Screenshot copied to clipboard and saved
Select here to mark up and share the image
Search
ENG
IN
14:48
25-10-2023


DAC_Phase4_Code - Jupyter Notebook
COVID-19 Analysis Models
localhost:8888/notebooks/DAC_Phase4_Code.ipynb
jupyter DAC_Phase4_Code Last Checkpoint: 10/18/2023 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
cases
In [90]:
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
y=cd_data[['deaths']]
x=cd_data[['cases']]
In [91]:
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2)
In [92]:
X_train.shape,X_test.shape,y_train.shape,y_test.shape
Out[92]:
((2184, 2), (546, 2), (2184, 1), (546, 1))
In [93]:
lr=LinearRegression()
In [94]:
lr.fit(X_train, y_train)
Out[94]:
LinearRegression
LinearRegression()
Search
ENG
IN
14:48
25-10-2023

CONCLUSION:

COVID-19 Case Analysis insights aid decision-makers in understanding current scenarios of , predicting future trends, and making informed choices.These insights guide healthcare professionals in allocating resources, implementing containment strategies, and adjusting public health measures to manage and mitigate the impact of COVID-19 effectively.