

Project Title : Covid-19 case analysis using cognos

Project I'd : Proj_229797_Team_2

Team members :

Y . Shankar

D . Payeelavan

S . Suventhan

PHASE 5:PROJECT DOCUMENTATION AND SUBMISSION

Abstract:

Project Definition: 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.

Design Thinking:

1. Analysis Objectives:

- ❖ To specify the objectives of analyzing COVID-19 cases and deaths data, such as comparing mean values and standard deviations and understand the patterns of COVID-19 cases and deaths over time, identifying spikes, declines, and potential outbreaks.

2. Data Collection:

- ❖ *To obtain the provided data file containing COVID-19 cases and deaths information per day and by country in the EU/EEA.*
- ❖ *This data is then used to generate insights and trends, assess the effectiveness of interventions, and predict future outbreaks.*

3. Visualization Strategy:

- ❖ *To visualize the mean values and standard deviations using IBM Cognos to create informative charts and graphs.*
- ❖ *Interactive dashboards with dynamic visualizations like line charts, heat maps, and geospatial representations to display trends over time and geographic regions of COVID-19 cases.*

4. Insights Generation:

- ❖ *To identify potential insights from the comparison of mean values and standard deviations of cases and deaths.*
- ❖ *These insights aid decision-makers in understanding current scenarios of , predicting future trends, and making informed choices.*
- ❖ *These insights guide policymakers and healthcare professionals in allocating resources, implementing containment strategies, and adjusting public health measures to manage and mitigate the impact of COVID-19 effectively.*

PHASE 2 : INNOVATION

STEP 1: *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.*

STEP 2: *Model Selection and Development*

- *We need to choose an appropriate machine learning algorithm for COVID-19 Case Analysis.*
- *Common choices include logistic regression, decision trees, random forests, support vector machines, or gradient boosting methods like XGBoost or LightGBM.*
- *If the COVID-19 dataset is large we can use advance technique like neural network.*

STEP 3: *Data Spiting*

- *Split the data into training, validation, and test sets.*
- *This allows you to train the model, tune hyperparameters, and evaluate its performance on unseen data.*

STEP 4: *Model Training*

- *Train the selected machine learning model on the training data.*

- Optimize hyperparameters using techniques like grid search or random search.

STEP 5: Model Evaluation

- Evaluate the model's performance using appropriate metrics, such as accuracy, precision, recall, F1-score, and ROC AUC.

STEP 6: Model Deployment

- Once satisfied with the model's performance, deploy it to your production environment.
- Implement a mechanism to regularly retrain the model as new data becomes available to ensure it remains accurate.

STEP 7: Monitor and Action

- Continuously monitor the model's predictions and act on them.
- Covid-19 case and death rate is monitored according to the Countries and Territories which helps to get valuable insight to take valuable decision

STEP 8: Documentation and Training

- Describe the problem and the goal of the COVID-19 Case Analysis
- Explain the data sources and features used in the model.
- Document the machine learning algorithm and its parameters.
- Describe the model evaluation metrics and results.
- Prepare the data by cleaning, preprocessing, and engineering features.
- Train the machine learning model using the prepared data.
- Evaluate the trained model on a holdout test set.

- *Deploy the trained model to production to predict COVID-19 case for newly affected people.*

PHASE 3& 4 : DEVELOPMENT PART 1

DEVELOPMENT PART 2

INTRODUCTION:

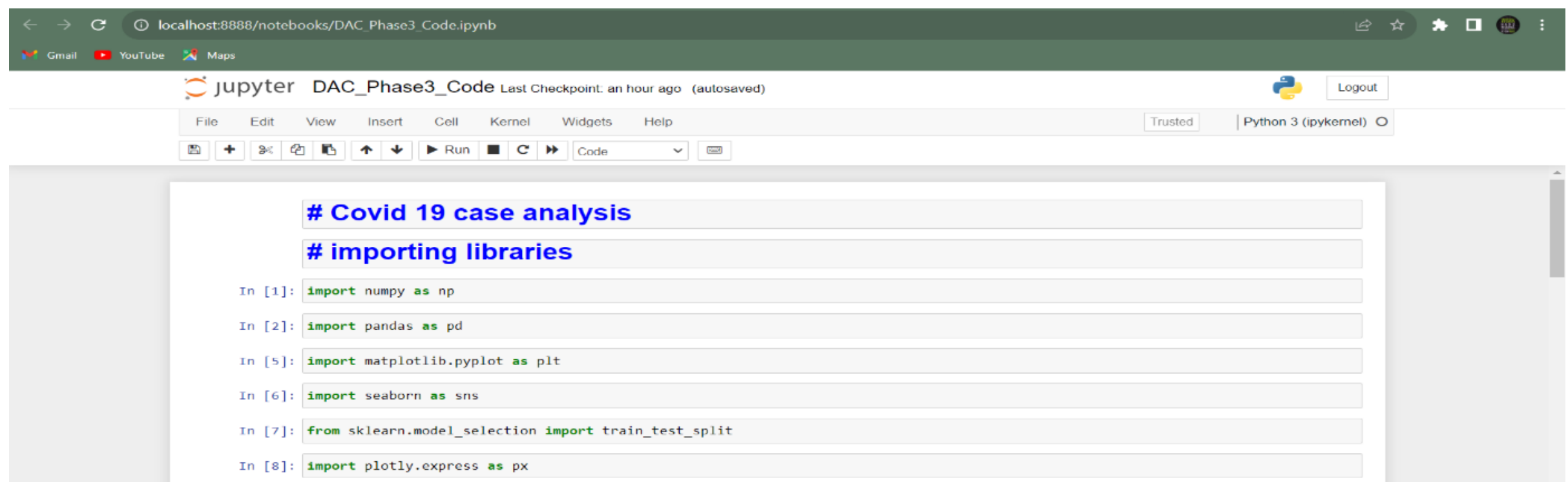
The project involves analyzing COVID-19 cases and deaths data using IBM Cognos. The objective is to compare and contrast the mean values and standard deviations of cases and associated deaths per day and by country in the EU/EEA. This project encompasses defining analysis objectives, collecting COVID-19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



```
# Covid 19 case analysis
# importing libraries

In [1]: import numpy as np

In [2]: import pandas as pd

In [5]: import matplotlib.pyplot as plt

In [6]: import seaborn as sns

In [7]: from sklearn.model_selection import train_test_split

In [8]: import plotly.express as px
```

► 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

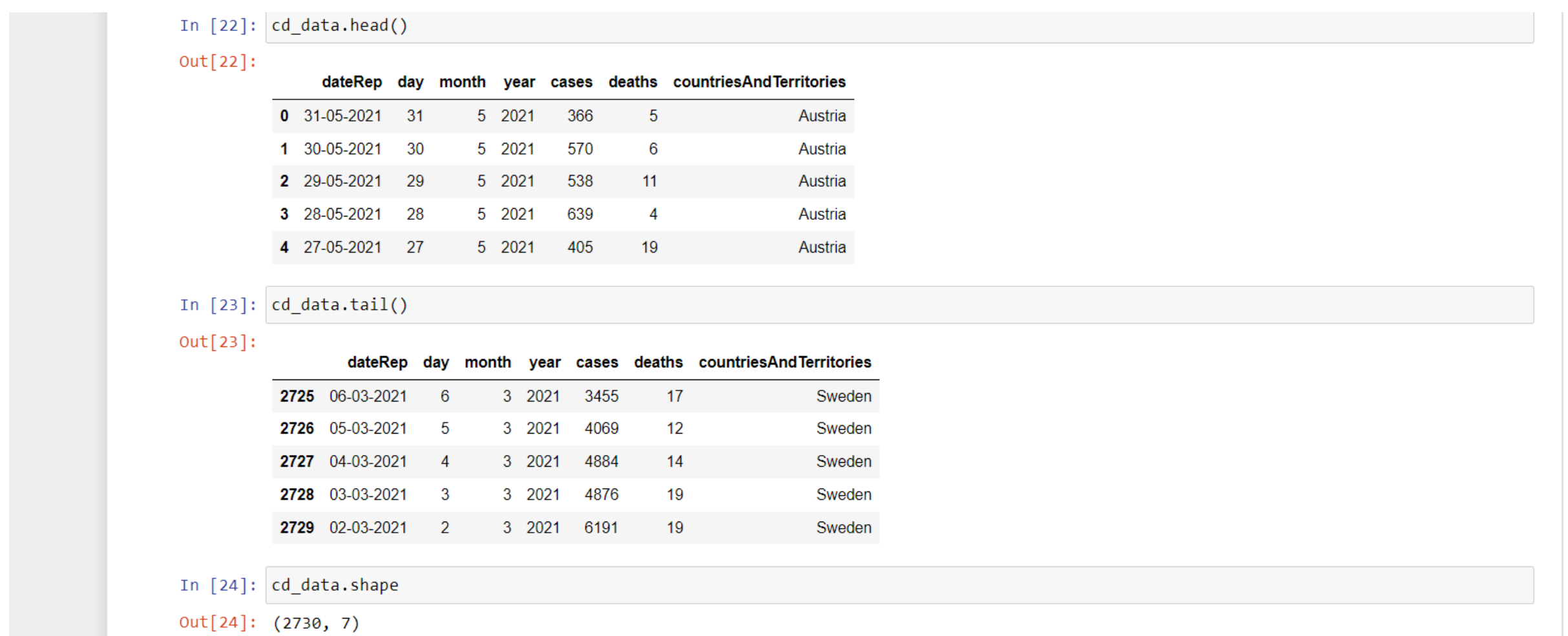
Out[21]:
```

	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

```
File Edit View Insert Cell Help Trusted Python 3 (ipykernel)
+ < > Run C Code
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)
memory usage: 149.4+ KB
```

► 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) Python 3 (ipykernel) Logout
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)
+ < > Run C Code
# 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.

DAC_Phase4/DAC_phase4 (2).ipynb | Home Page - Select or create a no | DAC_Phase4_Code - Jupyter Noteb | COVID-19 Analysis Models | +

localhost:8888/notebooks/DAC_Phase4_Code.ipynb

Gmail | YouTube | Maps

jupyter DAC_Phase4_Code Last Checkpoint: 10/18/2023 (autosaved)

Logout

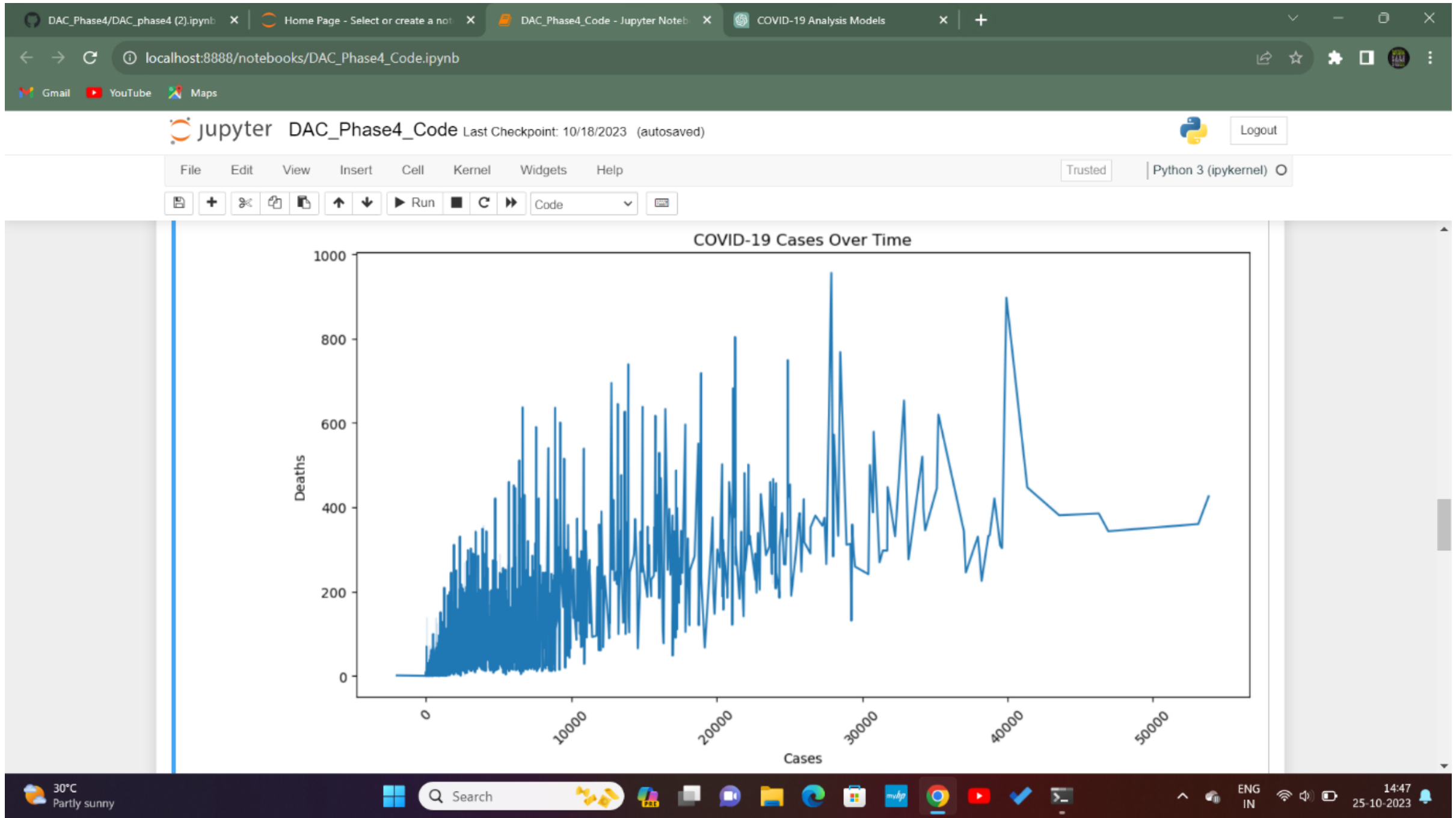
File | Edit | View | Insert | Cell | Kernel | Widgets | Help

Trusted | Python 3 (ipykernel)

Run

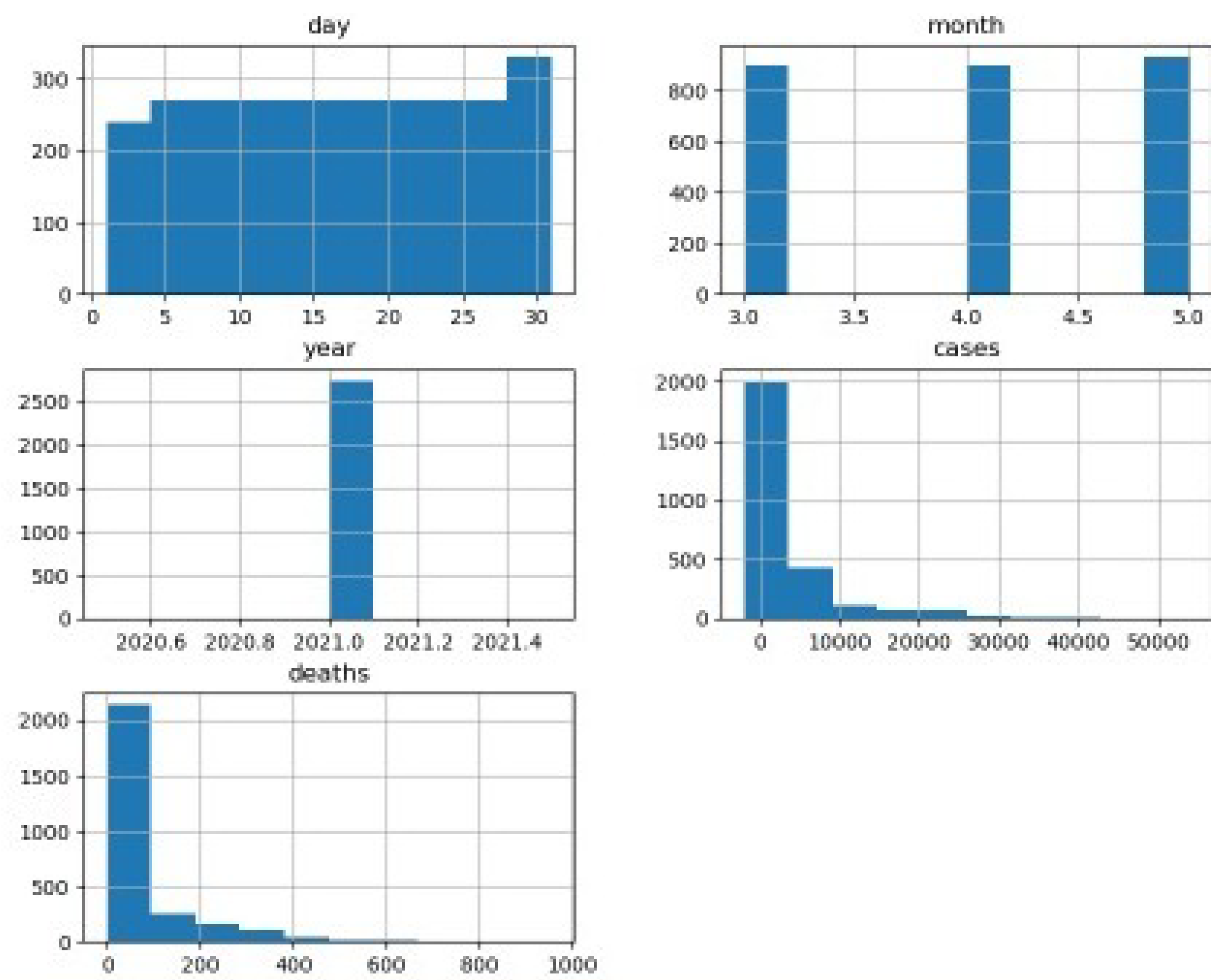
```
# Data visualization

In [89]: plt.figure(figsize=(12, 6))
sns.lineplot(x='cases', y='deaths', data=cd_data)
plt.title("COVID-19 Cases Over Time")
plt.xlabel("Cases")
plt.ylabel("Deaths")
plt.xticks(rotation=45)
plt.show()
```



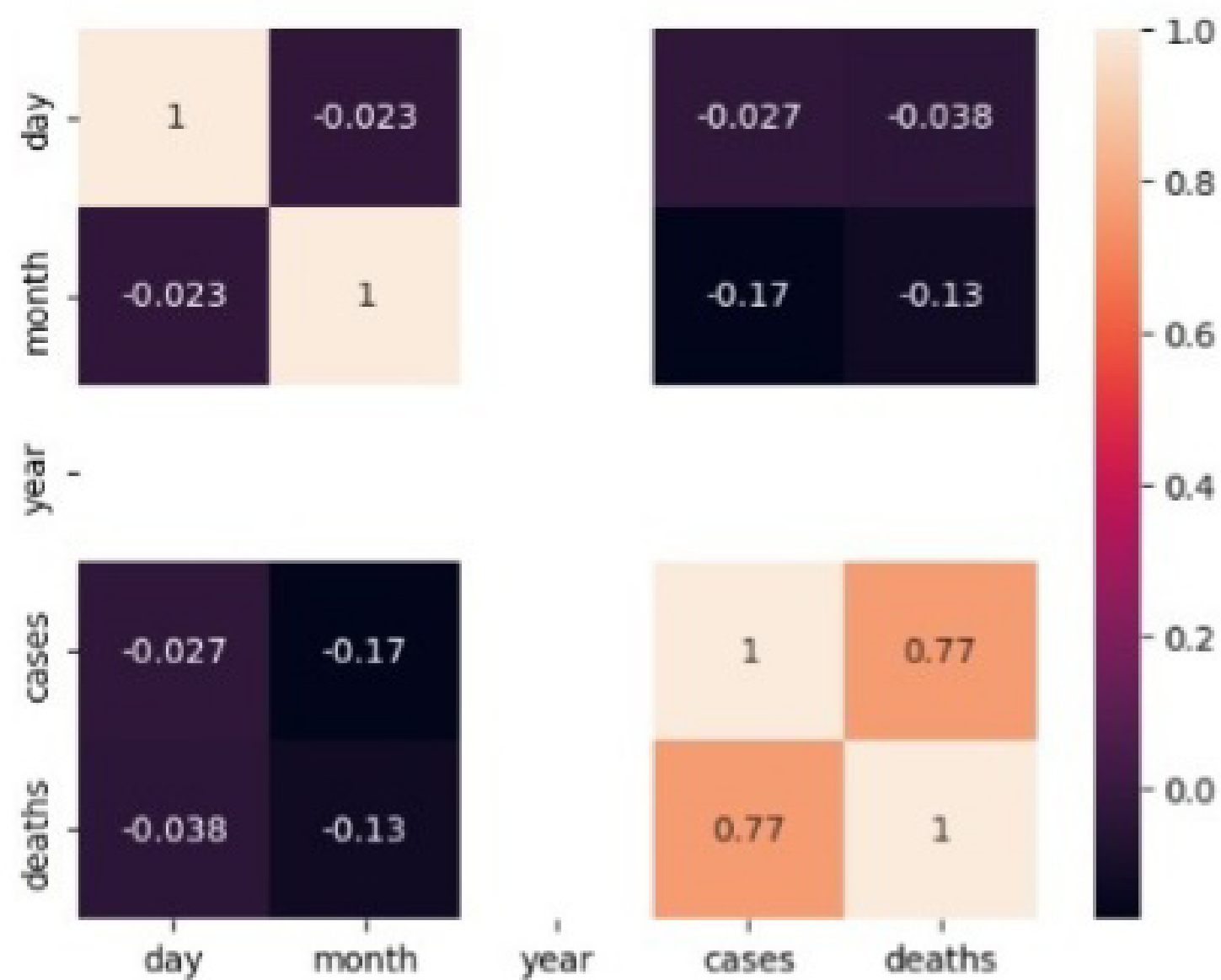

```
In [38]: ds.hist(figsize=(10,8))
```

```
Out[38]: array([[<Axes: title=["center": 'day']>,  
  <Axes: title=["center": 'month']>],  
  [<Axes: title=["center": 'year']>,  
  <Axes: title=["center": 'cases']>],  
  [<Axes: title=["center": 'deaths']>], dtype=object)
```

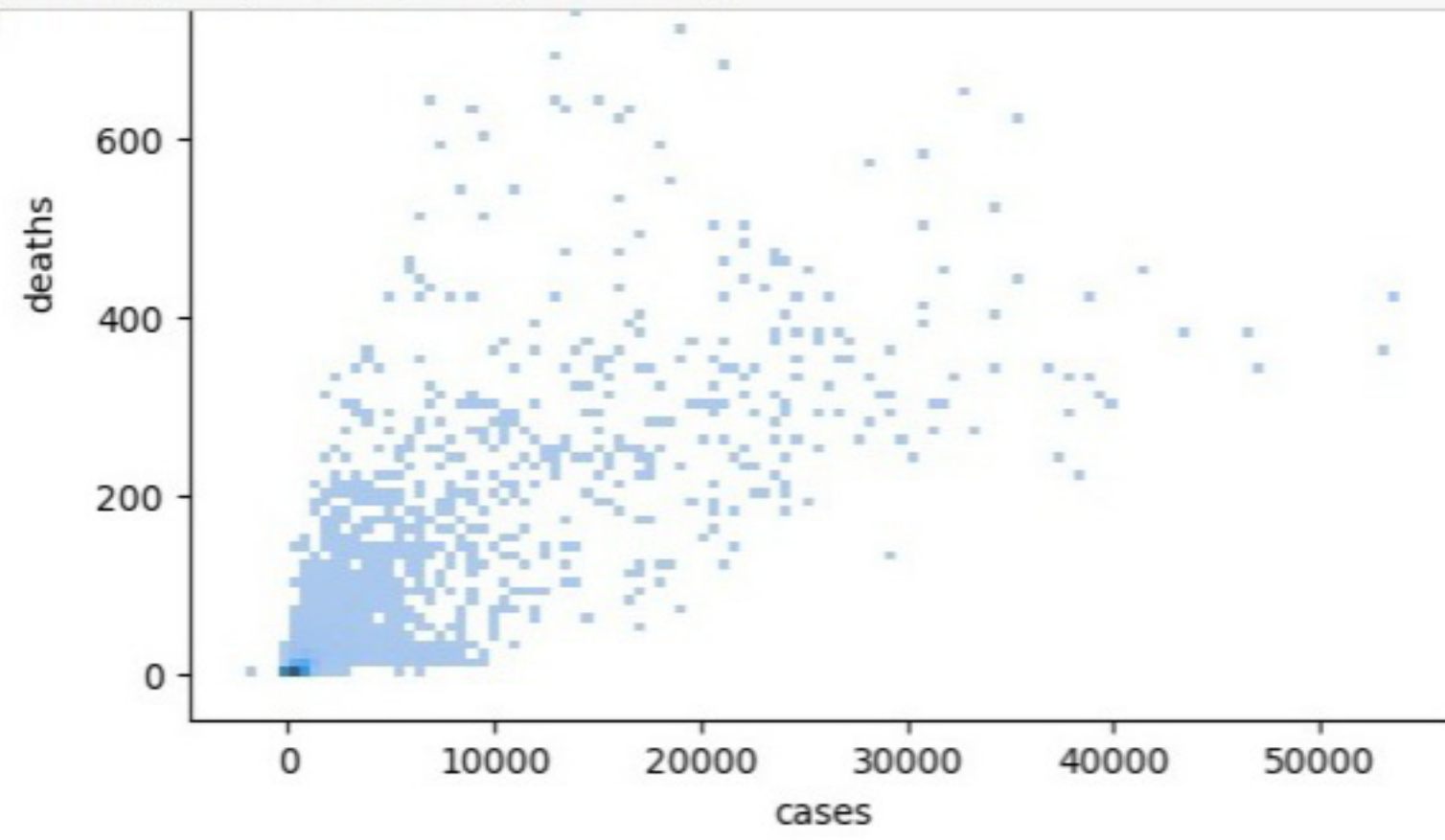


```
In [34]: sns.heatmap(ds.corr(numeric_only=True),annot=True)
```

```
Out[34]: <Axes: >
```

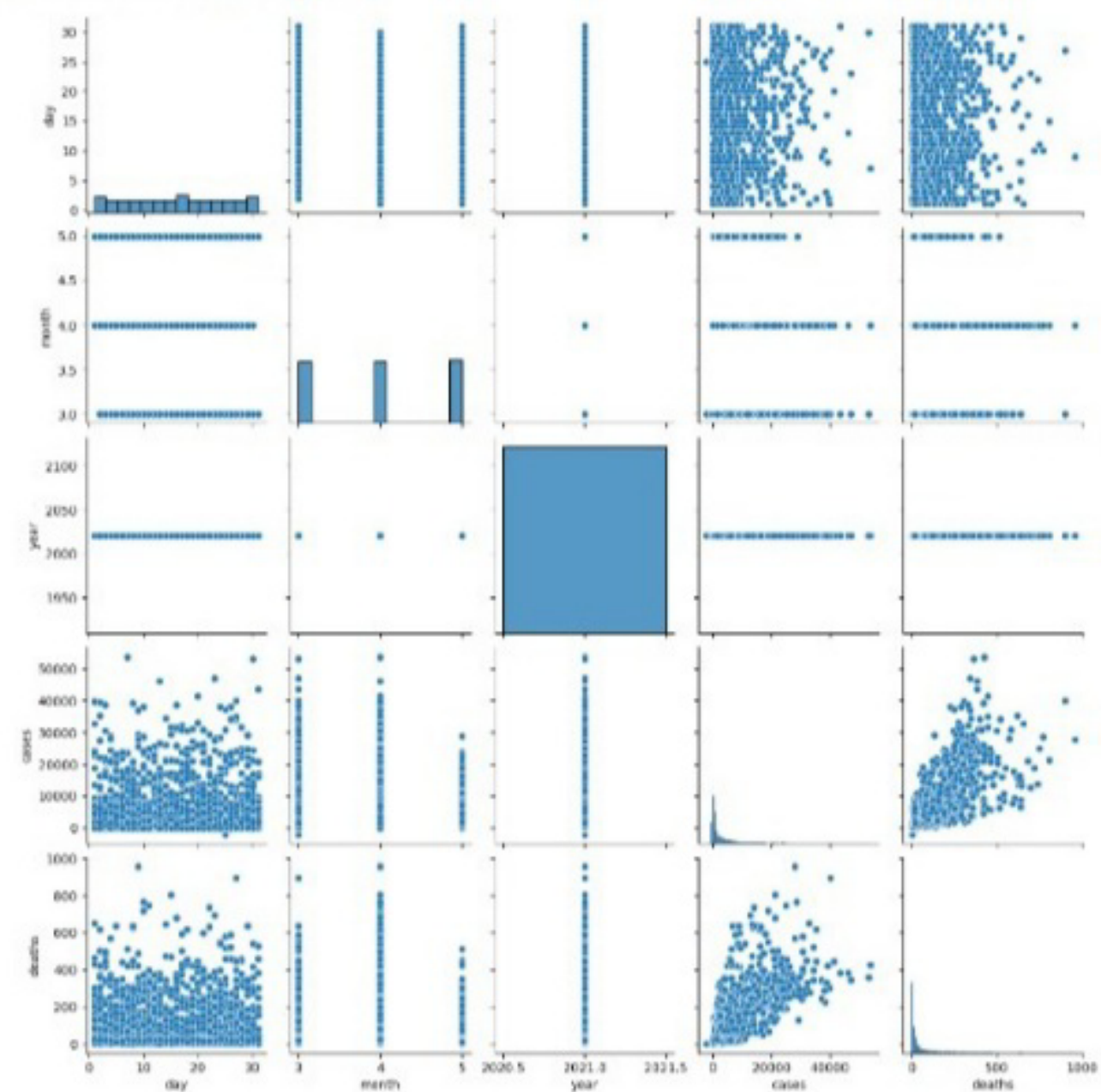


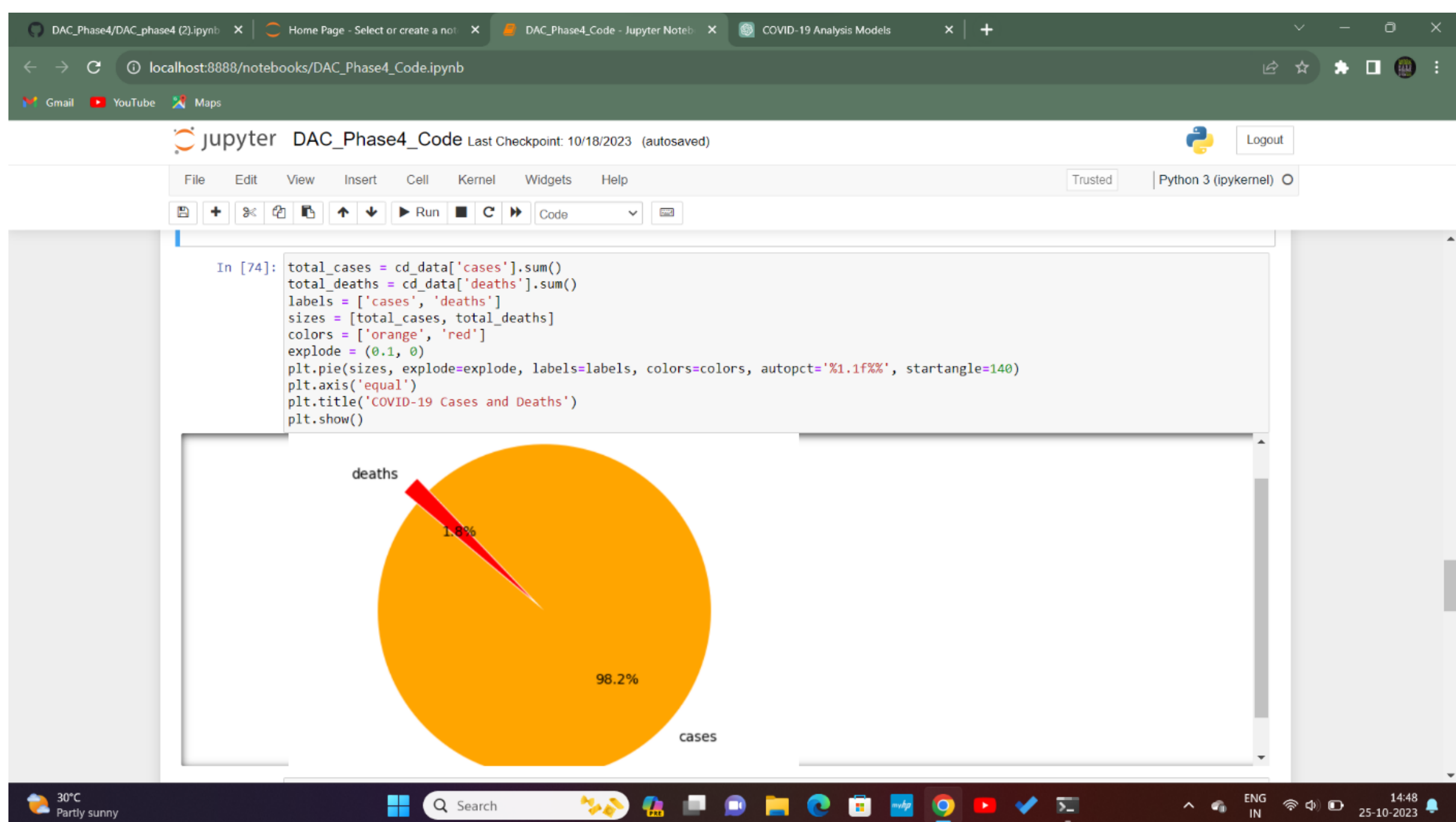
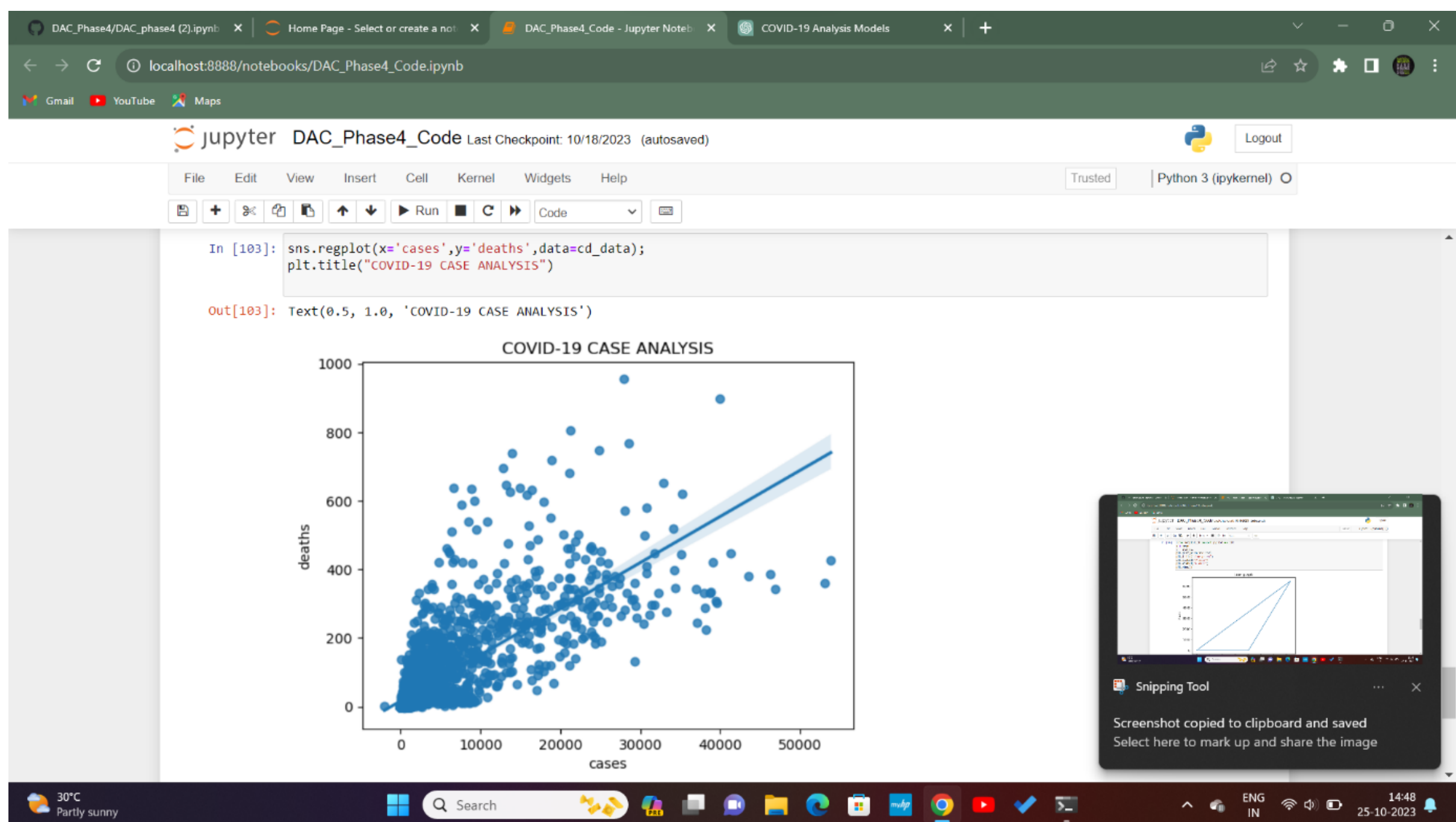
```
In [15]: sns.histplot(ds,x='cases',y='deaths')
```



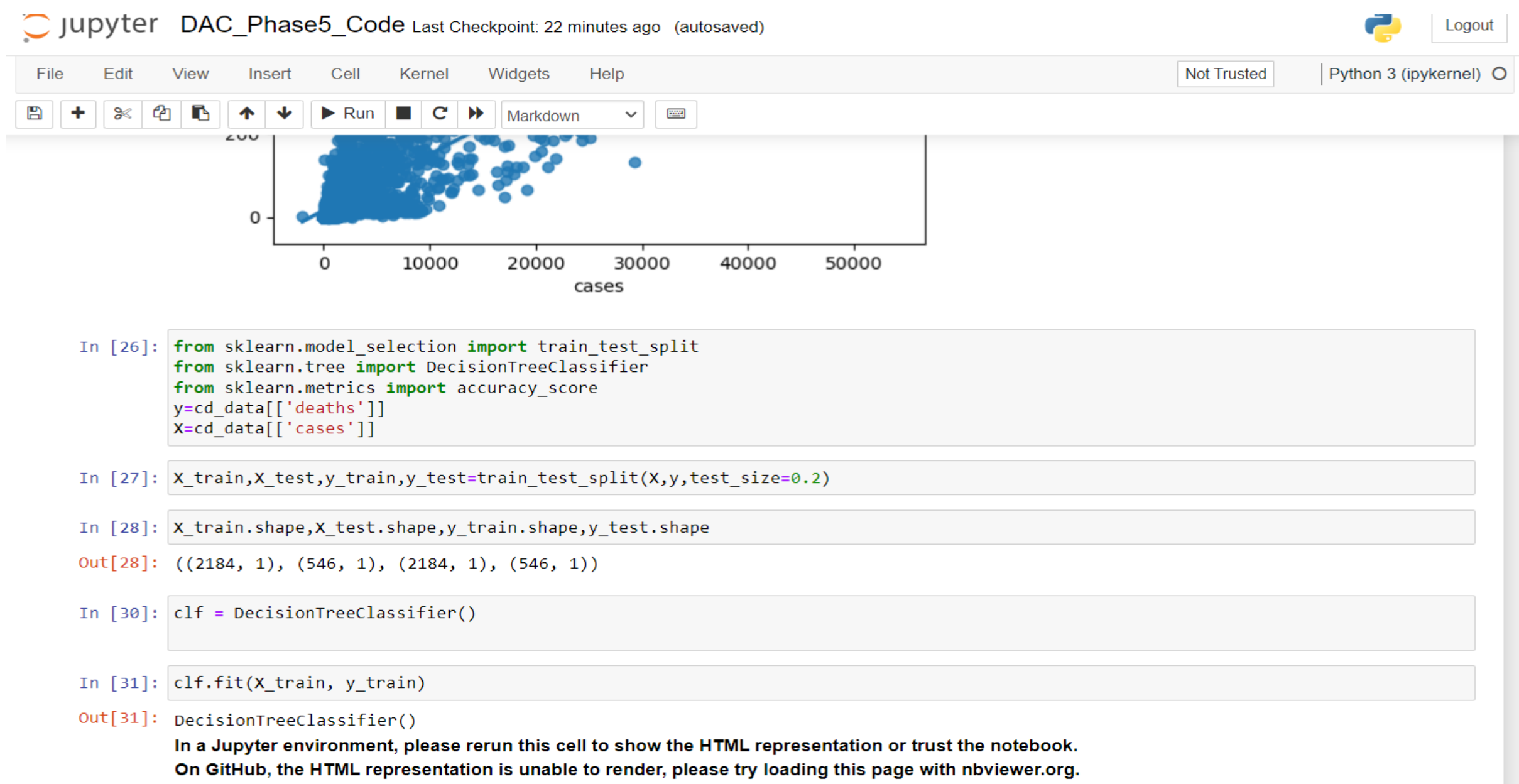
```
sns.pairplot(dataset)
```

```
Out[20]: cseaborn.axisgrid.PairGrid at 0x29522c03790>
```

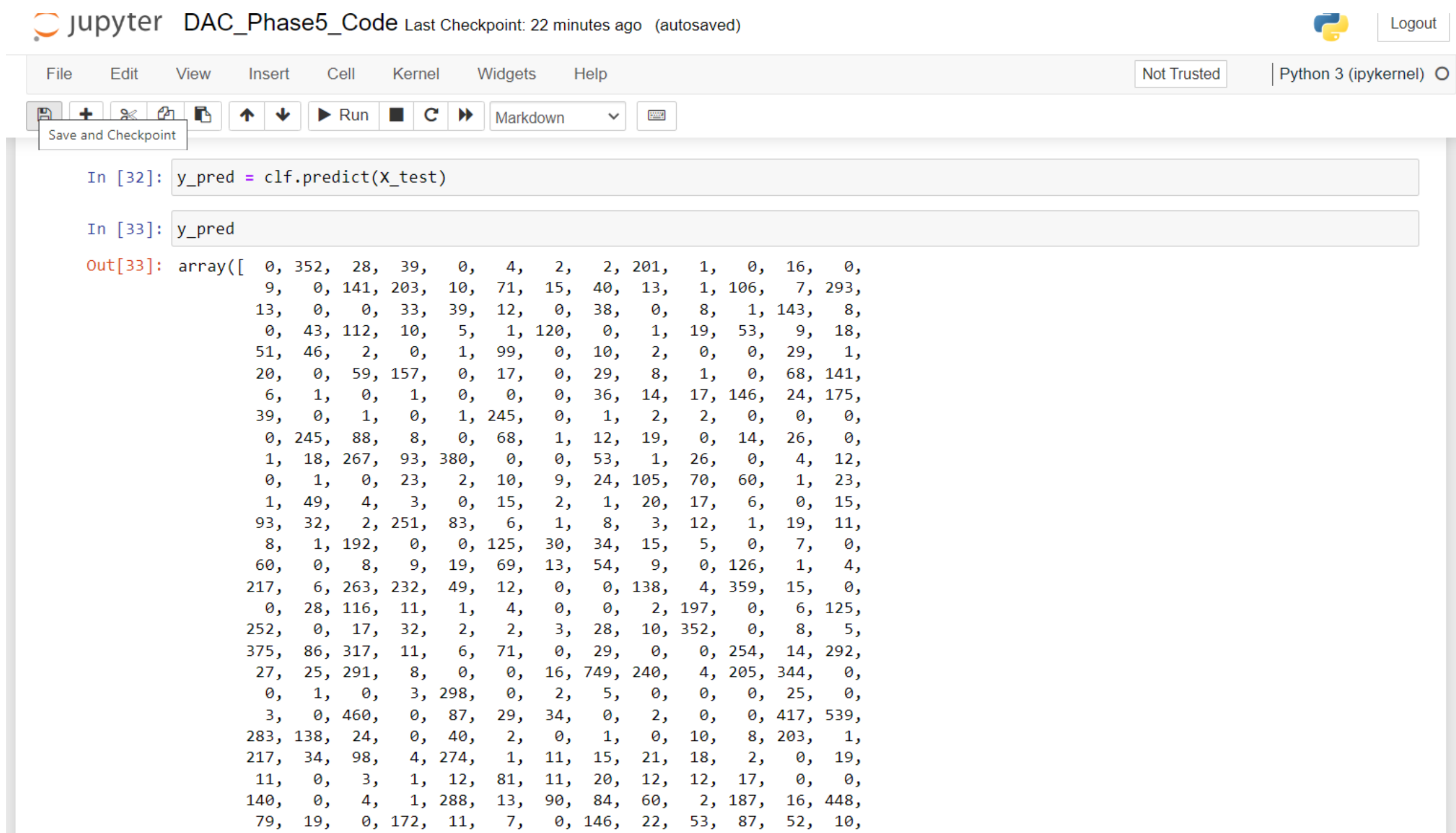




► Model Selection



► Model Evaluation



► *Accuracy*

jupyter DAC_Phase5_Code Last Checkpoint: 23 minutes ago (autosaved)

Python 3 (ipykernel)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted

Python 3 (ipykernel)

Code

79, 19, 0, 172, 11, 7, 0, 146, 22, 53, 87, 52, 10, 1, 2, 278, 0, 17, 203, 1, 1, 22, 71, 65, 187, 8, 105, 1, 0, 11, 7, 185, 51, 1, 1, 33, 166, 9, 36, 13, 34, 0, 13, 36, 0, 189, 330, 172, 359, 4, 278, 6, 0, 4, 201, 2, 0, 27, 188, 49, 0, 19, 32, 288, 300, 77, 4, 302, 15, 125, 5, 161, 38, 39, 128, 217, 1, 48, 0, 15, 460, 1, 0, 60, 4, 0, 100, 0, 18, 315, 373, 24, 161, 1, 285, 0, 1, 1, 221, 11, 342, 251, 23, 0, 25, 4, 19, 25, 0, 1, 1, 2, 0, 0, 0, 539, 20, 0, 359, 0, 0, 1, 0, 24, 1, 44, 2, 100, 1, 162, 0, 2, 112, 18, 0, 0, 79, 2, 8, 66, 32, 10, 422, 0, 41, 66, 3, 0, 502, 27, 2, 253, 221, 36, 63, 106, 201, 5, 9, 34, 2, 234, 0, 201, 43, 0, 34, 43, 13, 11, 502, 2, 300, 138, 3, 285, 8, 1, 189, 24, 0, 86, 112, 43, 653, 718, 0, 274, 0, 14, 0, 2, 22, 4, 18, 1, 44, 0, 0, 50, 0, 12, 4, 1, 0, 36, 102, 0], dtype=int64)

In [35]:

accuracy = accuracy_score(y_test, y_pred)
print(accuracy)
print(f"Accuracy: {accuracy * 100:.2f}%")

0.15018315018315018
Accuracy: 15.02%

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.