MONKEYPOX VIRUS DATA

Table of Contents

1. Introduction.. 3
2. Load Dataset...5
3. Basic Checks...8
4. Exploratory Data Analysis...................................... 10
5. What do we do with the Data................................ 21

Introduction

Monkeypox is an illness caused by the monkeypox virus. It is a viral zoonotic infection, meaning that it can spread from animals to humans. It can also spread from person to person.

It is normally encountered in Central and Western Africa, but has now spread to more than a dozen countries, including the United States. Cases in the current outbreak now number in the hundreds. This has fuelled more infection fears, no surprise in the wake of the still quite active COVID pandemic."

Monkeypox can occasionally be deadly, especially in poor places with inadequate healthcare, and is closely related to smallpox, which plagued humans for millennia. Smallpox was eradicated due to a worldwide vaccination campaign. In the United States, mass vaccinations ended in 1972, but the vaccines remain stockpiled. Monkeypox has been known since the late 1950s, and despite its name, its natural reservoir is rodents. It most often spreads between humans through contact with disease lesions, or through exhaled respiratory droplets during prolonged close contact."

Air travel from Africa is believed to have triggered the outbreak. Contact tracing is underway. The Centre for Disease Control and Prevention (CDC) hosted a press briefing this week in which officials focused to a large degree on spread of the virus among men who have sex with men, who appear to account for most cases of the current outbreak."

Symptoms

Monkeypox can cause a range of signs and symptoms. While some people have mild symptoms, others may develop more serious symptoms and need care in a health facility. Those at higher risk for severe disease or complications include people who are pregnant, children and persons that are immunocompromised.

The most common symptoms of monkeypox include **fever, headache, muscle aches, back pain, low energy, and swollen lymph nodes**. This is followed or accompanied by the development of a rash which can last for two to three weeks. The rash can be found on the face, palms of the hands, soles of the feet, eyes, mouth, throat, groin, and genital and/or anal regions of the body. The number of lesions can range from one to several thousand. Lesions begin flat, then fill with liquid before they crust over, dry up and fall off, with a fresh layer of skin forming underneath.

Symptoms typically last two to three weeks and usually go away on their own or with supportive care, such as medication for pain or fever. People remain infectious until all of the lesions have crusted over, the scabs fallen off and a new layer of skin has formed underneath.

Anyone who has symptoms that could be monkeypox or who has been in contact with someone who has monkeypox should call or visit a health care provider and seek their advice.

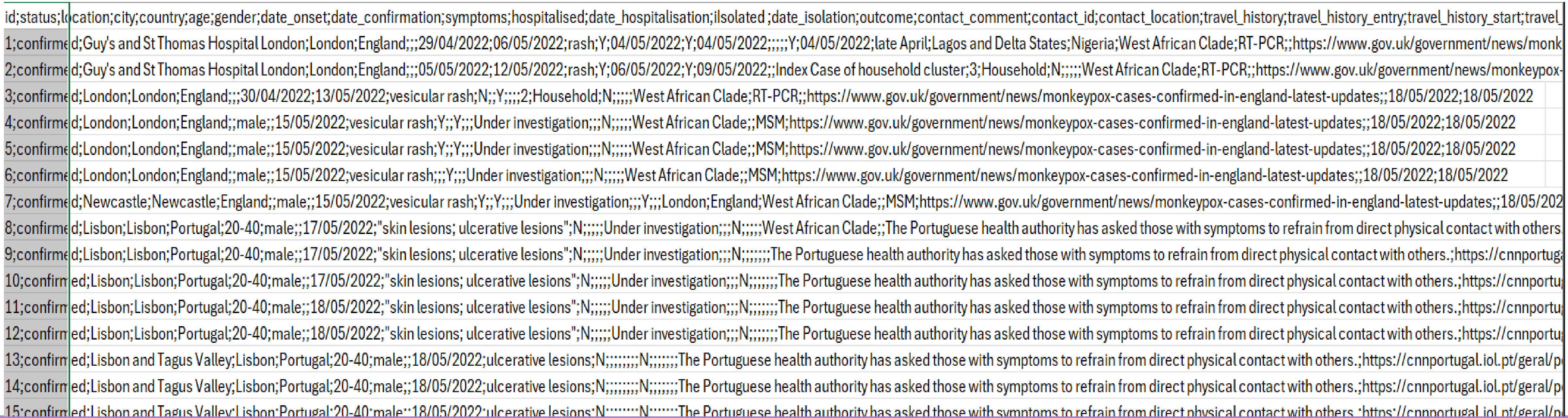
Load Dataset

About this file

Monkeypox Data Explorer
Visualizing the data produced by the Global.health team on the 2022 monkeypox outbreak.
by Edouard Mathieu, Saloni Dattani, Hannah Ritchie and Max Roser

https://ourworldindata.org/monkeypox.

The dataset comes as a csv file format separated (with scattered texts separated with semi colons). We had to create a csv tabulated file.

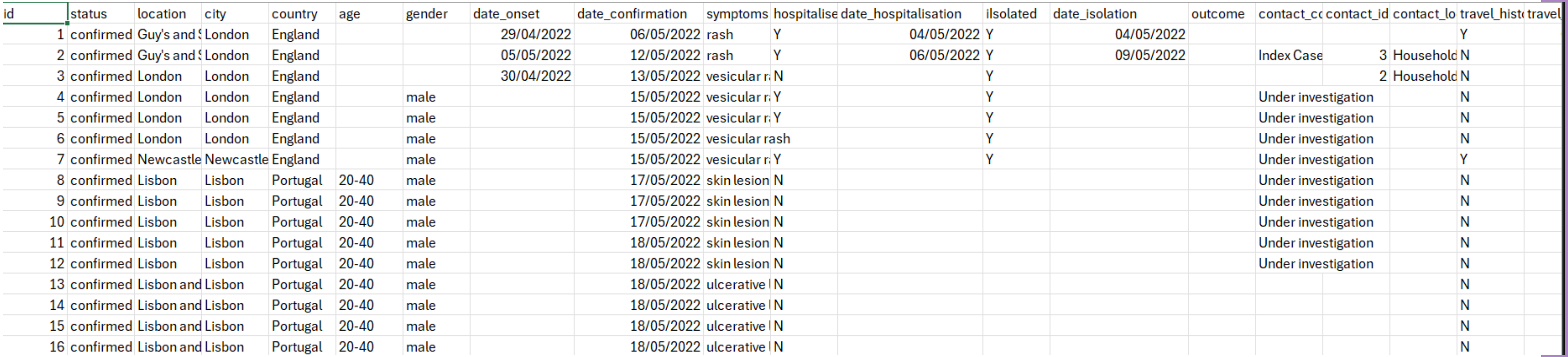


```
id;status;location;city;country;age;gender;date_onset;date_confirmation;symptoms;hospitalised;date_hospitalisation;ilsolated ;date_isolation;outcome;contact_comment;contact_id;contact_location;travel_history;travel_history_entry;travel_history_start;travel_
1;confirmed;Guy's and St Thomas Hospital London;London;England;;;29/04/2022;06/05/2022;rash;Y;04/05/2022;Y;04/05/2022;;;;;Y;04/05/2022;late April;Lagos and Delta States;Nigeria;West African Clade;RT-PCR;;https://www.gov.uk/government/news/monk
2;confirmed;Guy's and St Thomas Hospital London;London;England;;;05/05/2022;12/05/2022;rash;Y;06/05/2022;Y;09/05/2022;;Index Case of household cluster;3;Household;N;;;;;West African Clade;RT-PCR;;https://www.gov.uk/government/news/monkeypox-
3;confirmed;London;London;England;;;30/04/2022;13/05/2022;vesicular rash;N;;Y;;;;2;Household;N;;;;;West African Clade;RT-PCR;;https://www.gov.uk/government/news/monkeypox-cases-confirmed-in-england-latest-updates;;18/05/2022;18/05/2022
4;confirmed;London;London;England;;male;;15/05/2022;vesicular rash;Y;;Y;;;Under investigation;;;N;;;;;West African Clade;;MSM;https://www.gov.uk/government/news/monkeypox-cases-confirmed-in-england-latest-updates;;18/05/2022;18/05/2022
5;confirmed;London;London;England;;male;;15/05/2022;vesicular rash;Y;;Y;;;Under investigation;;;N;;;;;West African Clade;;MSM;https://www.gov.uk/government/news/monkeypox-cases-confirmed-in-england-latest-updates;;18/05/2022;18/05/2022
6;confirmed;London;London;England;;male;;15/05/2022;vesicular rash;;;Y;;;Under investigation;;;N;;;;;West African Clade;;MSM;https://www.gov.uk/government/news/monkeypox-cases-confirmed-in-england-latest-updates;;18/05/2022;18/05/2022
7;confirmed;Newcastle;Newcastle;England;;male;;15/05/2022;vesicular rash;Y;;Y;;;Under investigation;;;Y;;;London;England;West African Clade;;MSM;https://www.gov.uk/government/news/monkeypox-cases-confirmed-in-england-latest-updates;;18/05/202
8;confirmed;Lisbon;Lisbon;Portugal;20-40;male;;17/05/2022;"skin lesions; ulcerative lesions";N;;;;;Under investigation;;;N;;;;;West African Clade;;The Portuguese health authority has asked those with symptoms to refrain from direct physical contact with others
9;confirmed;Lisbon;Lisbon;Portugal;20-40;male;;17/05/2022;"skin lesions; ulcerative lesions";N;;;;;Under investigation;;;N;;;;;;The Portuguese health authority has asked those with symptoms to refrain from direct physical contact with others.;https://cnnportuga
10;confirmed;Lisbon;Lisbon;Portugal;20-40;male;;17/05/2022;"skin lesions; ulcerative lesions";N;;;;;Under investigation;;;N;;;;;;The Portuguese health authority has asked those with symptoms to refrain from direct physical contact with others.;https://cnnportu
11;confirmed;Lisbon;Lisbon;Portugal;20-40;male;;18/05/2022;"skin lesions; ulcerative lesions";N;;;;;Under investigation;;;N;;;;;;The Portuguese health authority has asked those with symptoms to refrain from direct physical contact with others.;https://cnnportu
12;confirmed;Lisbon;Lisbon;Portugal;20-40;male;;18/05/2022;"skin lesions; ulcerative lesions";N;;;;;Under investigation;;;N;;;;;;The Portuguese health authority has asked those with symptoms to refrain from direct physical contact with others.;https://cnnportu
13;confirmed;Lisbon and Tagus Valley;Lisbon;Portugal;20-40;male;;18/05/2022;ulcerative lesions;N;;;;;;;N;;;;;;The Portuguese health authority has asked those with symptoms to refrain from direct physical contact with others.;https://cnnportugal.iol.pt/geral/p
14;confirmed;Lisbon and Tagus Valley;Lisbon;Portugal;20-40;male;;18/05/2022;ulcerative lesions;N;;;;;;;N;;;;;;The Portuguese health authority has asked those with symptoms to refrain from direct physical contact with others.;https://cnnportugal.iol.pt/geral/p
15;confirmed;Lisbon and Tagus Valley;Lisbon;Portugal;20-40;male;;18/05/2022;ulcerative lesions;N;;;;;;;N;;;;;;The Portuguese health authority has asked those with symptoms to refrain from direct physical contact with others.;https://cnnportugal.iol.pt/geral/p
```

Data Preparation

A. Conversion of csv file from text to tabulated file.

B. Migration of the Microsoft csv into Jupyter Notebook



id	status	location	city	country	age	gender	date_onset	date_confirmation	symptoms	hospitalise	date_hospitalisation	ilsolated	date_isolation	outcome	contact_cc	contact_id	contact_lo	travel_hist	travel
1	confirmed	Guy's and S	London	England			29/04/2022	06/05/2022	rash	Y	04/05/2022	Y	04/05/2022					Y	
2	confirmed	Guy's and S	London	England			05/05/2022	12/05/2022	rash	Y	06/05/2022	Y	09/05/2022		Index Case	3	Household	N	
3	confirmed	London	London	England			30/04/2022	13/05/2022	vesicular r	N		Y				2	Household	N	
4	confirmed	London	London	England		male		15/05/2022	vesicular r	Y		Y			Under investigation			N	
5	confirmed	London	London	England		male		15/05/2022	vesicular r	Y		Y			Under investigation			N	
6	confirmed	London	London	England		male		15/05/2022	vesicular rash			Y			Under investigation			N	
7	confirmed	Newcastle	Newcastle	England		male		15/05/2022	vesicular r	Y		Y			Under investigation			Y	
8	confirmed	Lisbon	Lisbon	Portugal	20-40	male		17/05/2022	skin lesion	N					Under investigation			N	
9	confirmed	Lisbon	Lisbon	Portugal	20-40	male		17/05/2022	skin lesion	N					Under investigation			N	
10	confirmed	Lisbon	Lisbon	Portugal	20-40	male		17/05/2022	skin lesion	N					Under investigation			N	
11	confirmed	Lisbon	Lisbon	Portugal	20-40	male		18/05/2022	skin lesion	N					Under investigation			N	
12	confirmed	Lisbon	Lisbon	Portugal	20-40	male		18/05/2022	skin lesion	N					Under investigation			N	
13	confirmed	Lisbon and	Lisbon	Portugal	20-40	male		18/05/2022	ulcerative l	N								N	
14	confirmed	Lisbon and	Lisbon	Portugal	20-40	male		18/05/2022	ulcerative l	N								N	
15	confirmed	Lisbon and	Lisbon	Portugal	20-40	male		18/05/2022	ulcerative l	N								N	
16	confirmed	Lisbon and	Lisbon	Portugal	20-40	male		18/05/2022	ulcerative l	N								N	

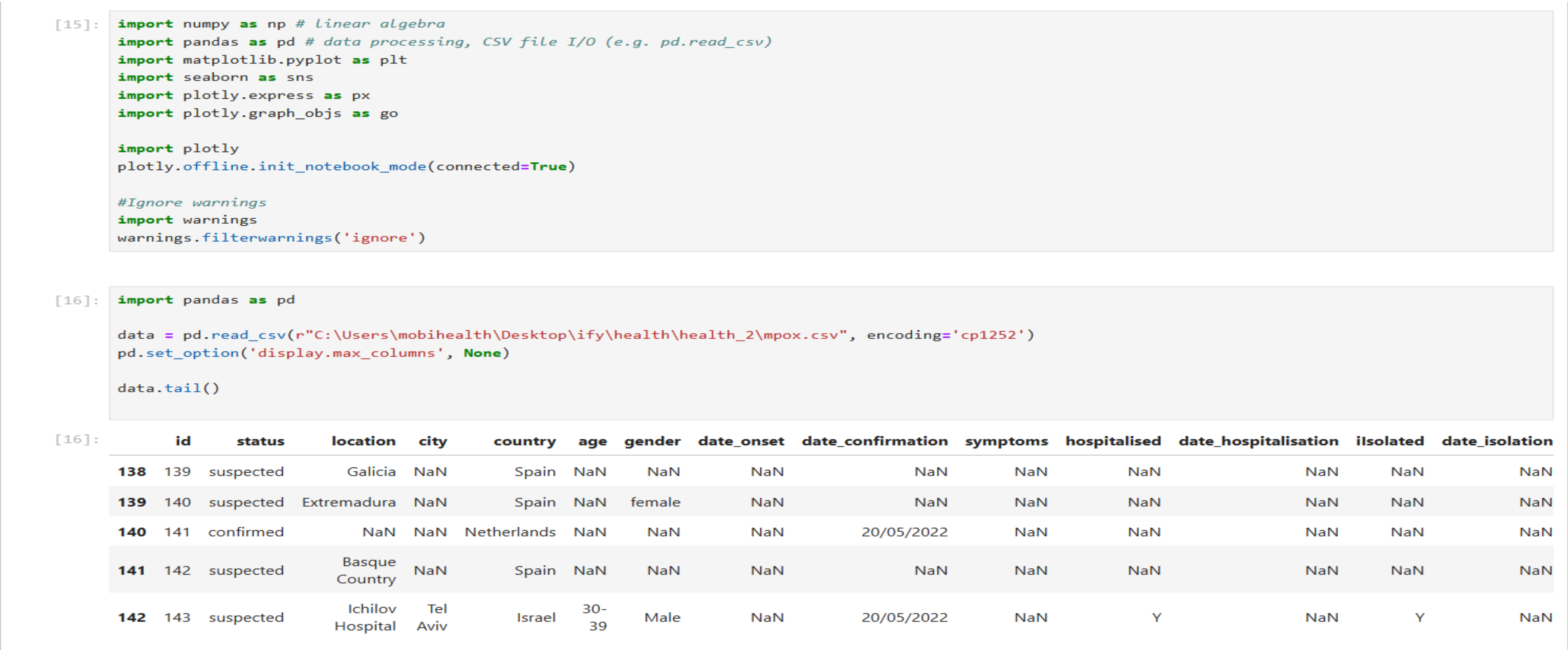
Steps to convert csv file from texts to tabulated form.

4. On the Convert to Text Column dialogue box, click on Next.

5. Check the semi-colon box, and click on Next.

6. Click on Next again.

7. Click on Finish.



```
[15]: import numpy as np # linear algebra
      import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
      import matplotlib.pyplot as plt
      import seaborn as sns
      import plotly.express as px
      import plotly.graph_objs as go

      import plotly
      plotly.offline.init_notebook_mode(connected=True)

      #Ignore warnings
      import warnings
      warnings.filterwarnings('ignore')
```

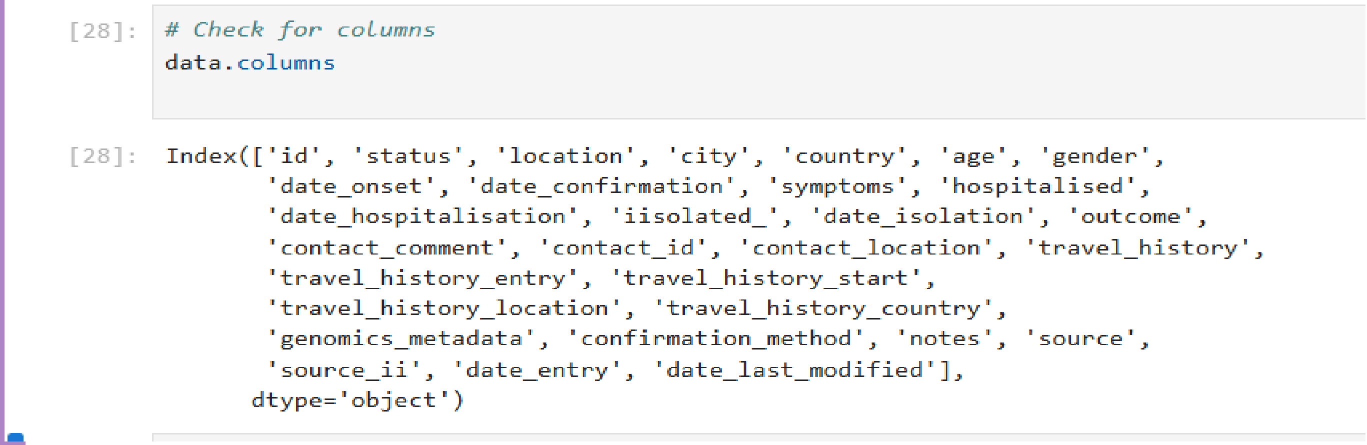
```
[16]: import pandas as pd

      data = pd.read_csv(r"C:\Users\mobihealth\Desktop\ify\health\health_2\mpox.csv", encoding='cp1252')
      pd.set_option('display.max_columns', None)

      data.tail()
```

[16]:

	id	status	location	city	country	age	gender	date_onset	date_confirmation	symptoms	hospitalised	date_hospitalisation	iIsolated	date_isolation
138	139	suspected	Galicia	NaN	Spain	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
139	140	suspected	Extremadura	NaN	Spain	NaN	female	NaN	NaN	NaN	NaN	NaN	NaN	NaN
140	141	confirmed	NaN	NaN	Netherlands	NaN	NaN	NaN	20/05/2022	NaN	NaN	NaN	NaN	NaN
141	142	suspected	Basque Country	NaN	Spain	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
142	143	suspected	Ichilov Hospital	Tel Aviv	Israel	30-39	Male	NaN	20/05/2022	NaN	Y	NaN	Y	NaN

Basic Checks

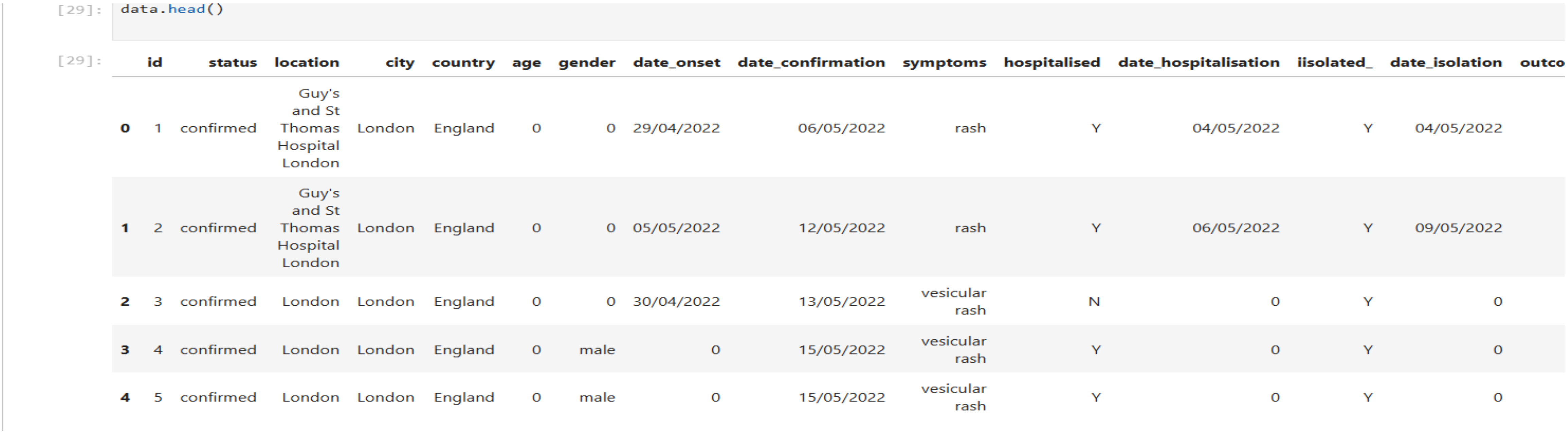
1. Check for Data Columns



```
[28]: # Check for columns
      data.columns
```

```
[28]: Index(['id', 'status', 'location', 'city', 'country', 'age', 'gender',
              'date_onset', 'date_confirmation', 'symptoms', 'hospitalised',
              'date_hospitalisation', 'iisolated_', 'date_isolation', 'outcome',
              'contact_comment', 'contact_id', 'contact_location', 'travel_history',
              'travel_history_entry', 'travel_history_start',
              'travel_history_location', 'travel_history_country',
              'genomics_metadata', 'confirmation_method', 'notes', 'source',
              'source_ii', 'date_entry', 'date_last_modified'],
             dtype='object')
```

2. Column Headers

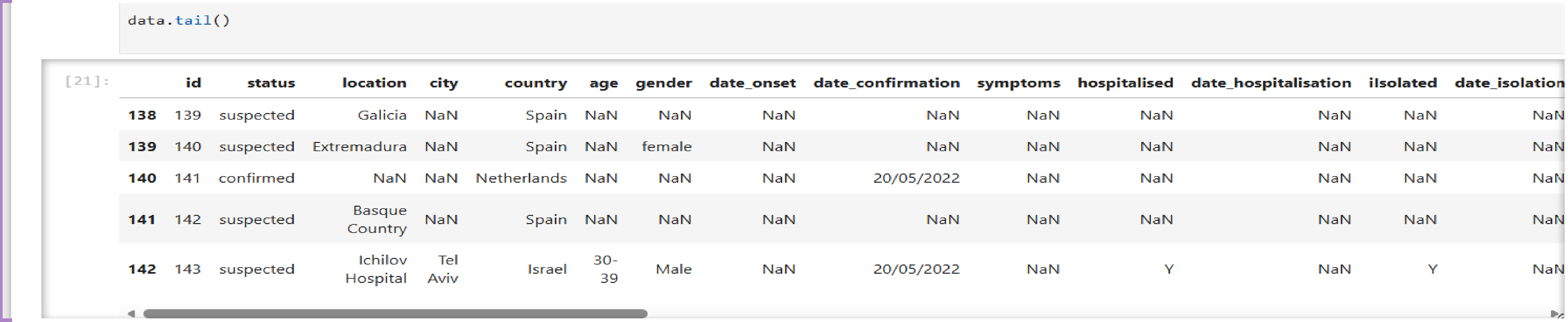


```
[29]: data.head()
```

[29]:

	id	status	location	city	country	age	gender	date_onset	date_confirmation	symptoms	hospitalised	date_hospitalisation	iisolated_	date_isolation	outco
0	1	confirmed	Guy's and St Thomas Hospital London	London	England	0	0	29/04/2022	06/05/2022	rash	Y	04/05/2022	Y	04/05/2022	
1	2	confirmed	Guy's and St Thomas Hospital London	London	England	0	0	05/05/2022	12/05/2022	rash	Y	06/05/2022	Y	09/05/2022	
2	3	confirmed	London	London	England	0	0	30/04/2022	13/05/2022	vesicular rash	N	0	Y	0	
3	4	confirmed	London	London	England	0	male	0	15/05/2022	vesicular rash	Y	0	Y	0	
4	5	confirmed	London	London	England	0	male	0	15/05/2022	vesicular rash	Y	0	Y	0	

3. Tail of Dataset

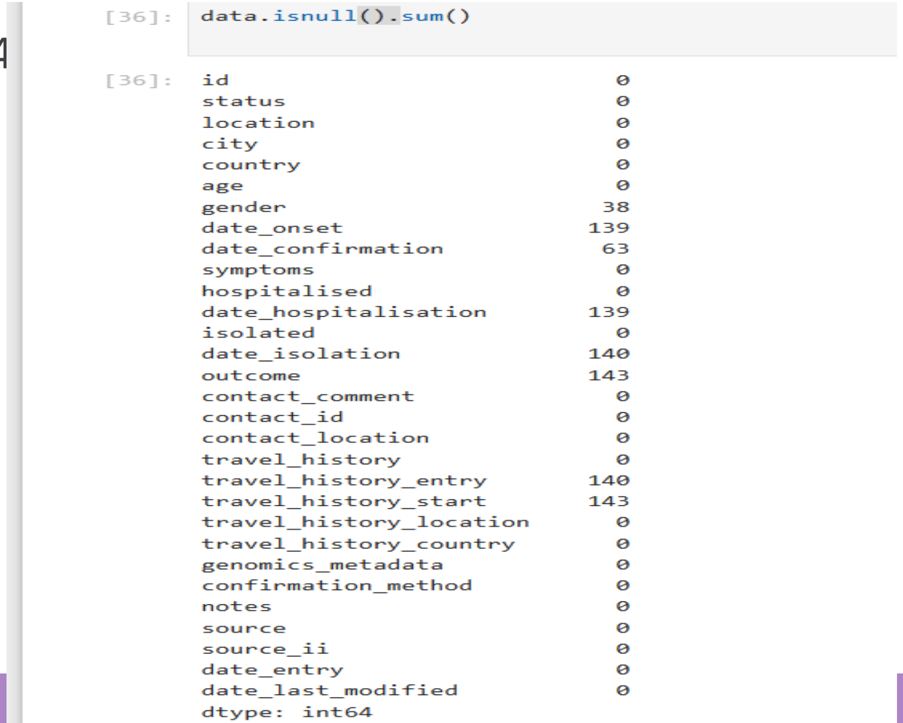


```
data.tail()
```

[21]:

	id	status	location	city	country	age	gender	date_onset	date_confirmation	symptoms	hospitalised	date_hospitalisation	iIsolated	date_isolation
138	139	suspected	Galicia	NaN	Spain	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
139	140	suspected	Extremadura	NaN	Spain	NaN	female	NaN	NaN	NaN	NaN	NaN	NaN	NaN
140	141	confirmed	NaN	NaN	Netherlands	NaN	NaN	NaN	20/05/2022	NaN	NaN	NaN	NaN	NaN
141	142	suspected	Basque Country	NaN	Spain	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
142	143	suspected	Ichilov Hospital	Tel Aviv	Israel	30-39	Male	NaN	20/05/2022	NaN	Y	NaN	Y	NaN

4



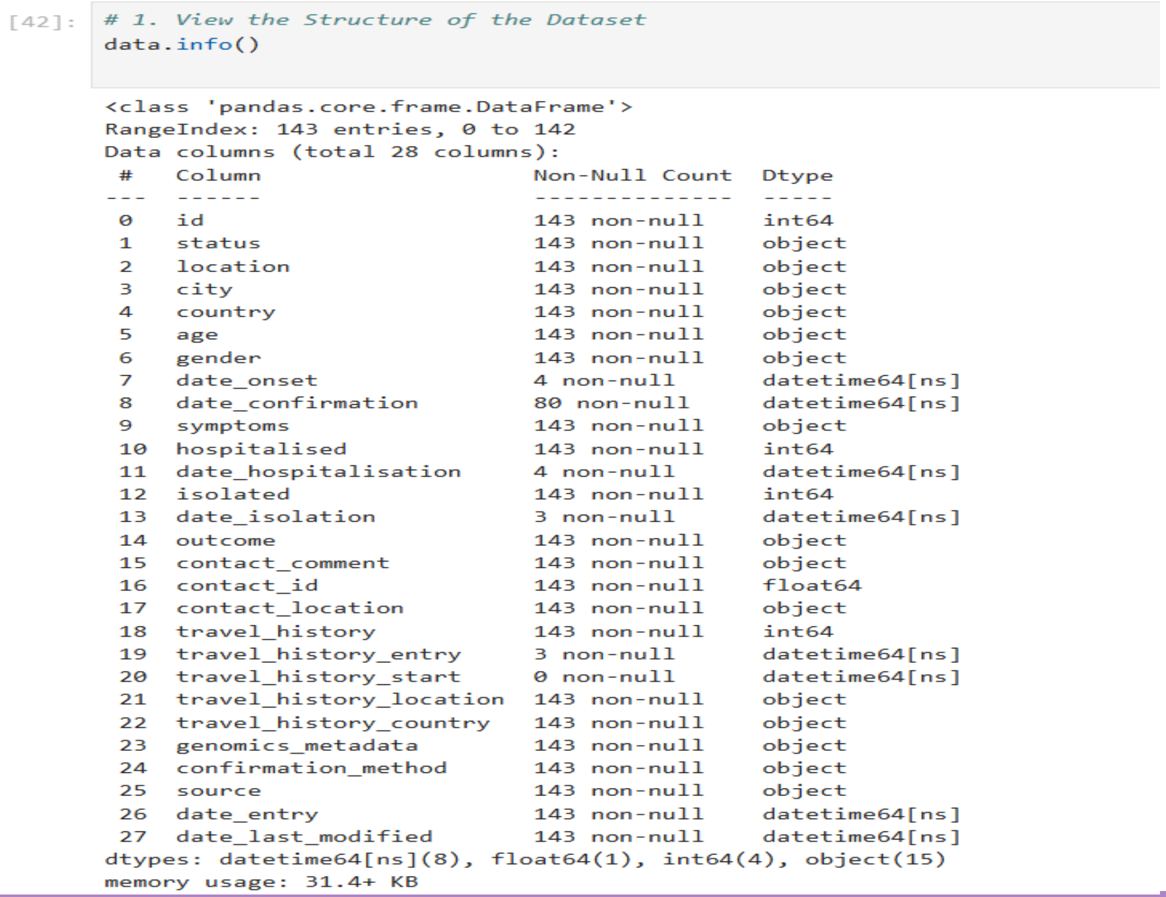
[36]:
```
data.isnull().sum()
```

[36]:
```
id                              0
status                          0
location                        0
city                            0
country                         0
age                             0
gender                         38
date_onset                    139
date_confirmation              63
symptoms                        0
hospitalised                    0
date_hospitalisation          139
isolated                        0
date_isolation                140
outcome                       143
contact_comment                 0
contact_id                      0
contact_location                0
travel_history                  0
travel_history_entry          140
travel_history_start          143
travel_history_location         0
travel_history_country          0
genomics_metadata               0
confirmation_method             0
notes                           0
source                          0
source_ii                       0
date_entry                      0
date_last_modified              0
dtype: int64
```

Exploratory Data Analysis

Below are the 10 Exploratory Data Analysis (EDA) techniques tailored exactly to the mpox.csv dataset based on the columns provided. Each step includes the exact query/code to run in Jupyter Notebook.

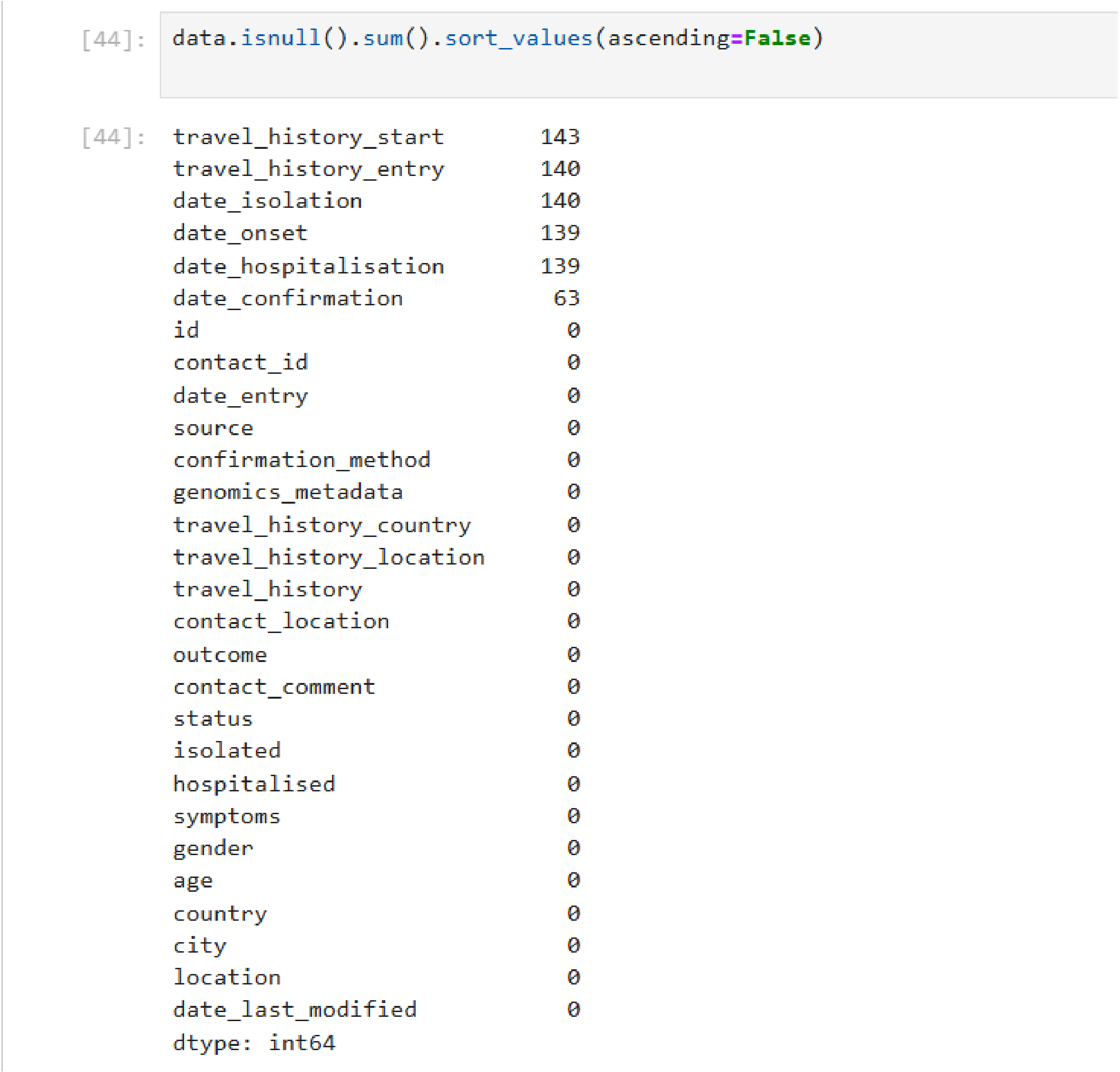
1. **View the Structure of the Dataset**



```
[42]: # 1. View the Structure of the Dataset
      data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 143 entries, 0 to 142
Data columns (total 28 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   id                        143 non-null    int64
 1   status                    143 non-null    object
 2   location                  143 non-null    object
 3   city                      143 non-null    object
 4   country                   143 non-null    object
 5   age                       143 non-null    object
 6   gender                    143 non-null    object
 7   date_onset                4 non-null      datetime64[ns]
 8   date_confirmation         80 non-null     datetime64[ns]
 9   symptoms                  143 non-null    object
 10  hospitalised              143 non-null    int64
 11  date_hospitalisation      4 non-null      datetime64[ns]
 12  isolated                  143 non-null    int64
 13  date_isolation            3 non-null      datetime64[ns]
 14  outcome                   143 non-null    object
 15  contact_comment           143 non-null    object
 16  contact_id                143 non-null    float64
 17  contact_location          143 non-null    object
 18  travel_history            143 non-null    int64
 19  travel_history_entry      3 non-null      datetime64[ns]
 20  travel_history_start      0 non-null      datetime64[ns]
 21  travel_history_location   143 non-null    object
 22  travel_history_country    143 non-null    object
 23  genomics_metadata         143 non-null    object
 24  confirmation_method       143 non-null    object
 25  source                    143 non-null    object
 26  date_entry                143 non-null    datetime64[ns]
 27  date_last_modified        143 non-null    datetime64[ns]
dtypes: datetime64[ns](8), float64(1), int64(4), object(15)
memory usage: 31.4+ KB
```

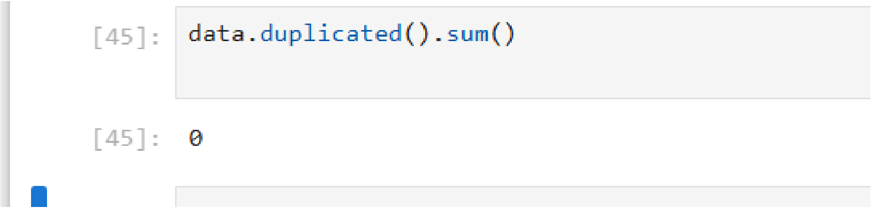
2. Inspect Missing Values Across All Columns



```
[44]: data.isnull().sum().sort_values(ascending=False)
```

```
[44]: travel_history_start       143
      travel_history_entry       140
      date_isolation             140
      date_onset                 139
      date_hospitalisation       139
      date_confirmation           63
      id                           0
      contact_id                   0
      date_entry                   0
      source                       0
      confirmation_method          0
      genomics_metadata            0
      travel_history_country       0
      travel_history_location      0
      travel_history               0
      contact_location             0
      outcome                      0
      contact_comment              0
      status                       0
      isolated                     0
      hospitalised                 0
      symptoms                     0
      gender                       0
      age                          0
      country                      0
      city                         0
      location                     0
      date_last_modified           0
      dtype: int64
```

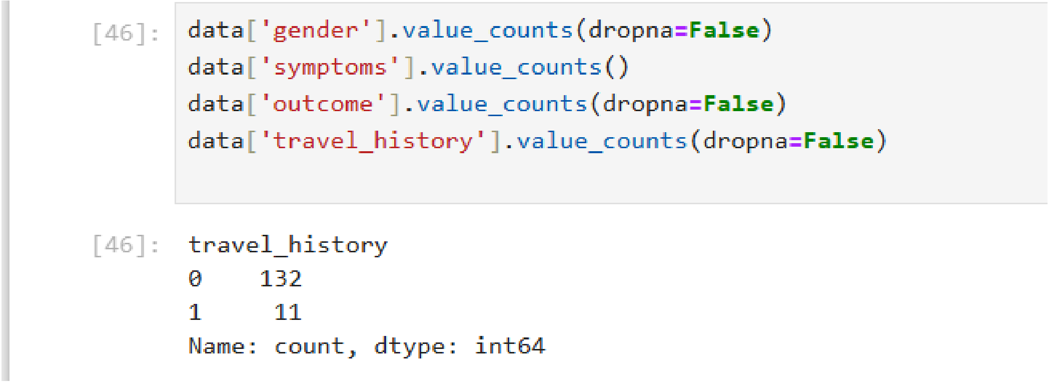
3. Check for Duplicate Entries



```
[45]: data.duplicated().sum()
```

```
[45]: 0
```

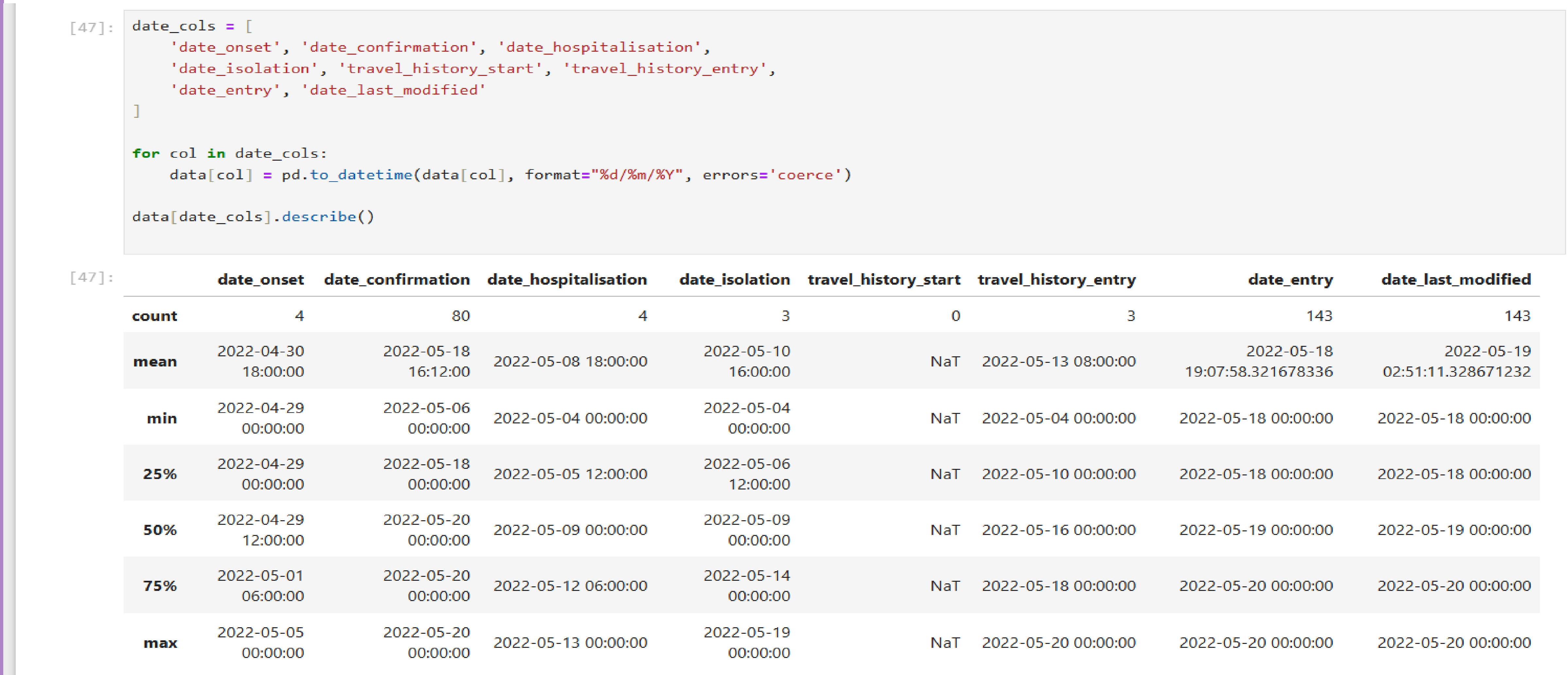
4. Explore Categorical Variables



```
[46]: data['gender'].value_counts(dropna=False)
      data['symptoms'].value_counts()
      data['outcome'].value_counts(dropna=False)
      data['travel_history'].value_counts(dropna=False)
```

```
[46]: travel_history
      0    132
      1     11
      Name: count, dtype: int64
```

5. Convert All Date Columns and Inspect Date Ranges



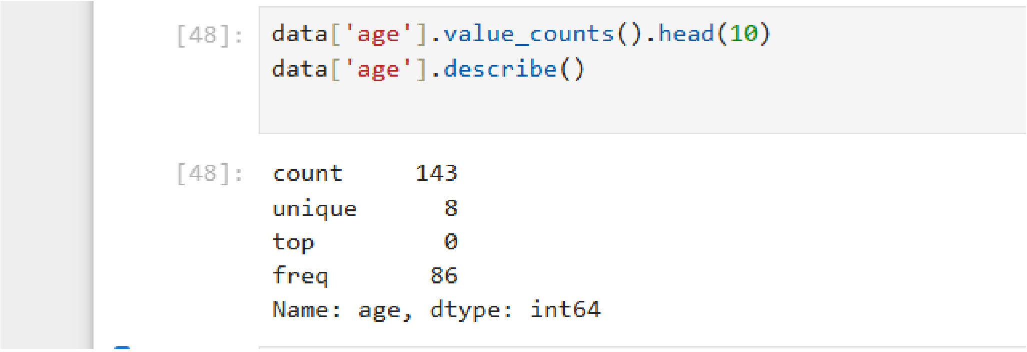
[47]:

```
date_cols = [
    'date_onset', 'date_confirmation', 'date_hospitalisation',
    'date_isolation', 'travel_history_start', 'travel_history_entry',
    'date_entry', 'date_last_modified'
]

for col in date_cols:
    data[col] = pd.to_datetime(data[col], format="%d/%m/%Y", errors='coerce')

data[date_cols].describe()
```

[47]:

	date_onset	date_confirmation	date_hospitalisation	date_isolation	travel_history_start	travel_history_entry	date_entry	date_last_modified
count	4	80	4	3	0	3	143	143
mean	2022-04-30 18:00:00	2022-05-18 16:12:00	2022-05-08 18:00:00	2022-05-10 16:00:00	NaT	2022-05-13 08:00:00	2022-05-18 19:07:58.321678336	2022-05-19 02:51:11.328671232
min	2022-04-29 00:00:00	2022-05-06 00:00:00	2022-05-04 00:00:00	2022-05-04 00:00:00	NaT	2022-05-04 00:00:00	2022-05-18 00:00:00	2022-05-18 00:00:00
25%	2022-04-29 00:00:00	2022-05-18 00:00:00	2022-05-05 12:00:00	2022-05-06 12:00:00	NaT	2022-05-10 00:00:00	2022-05-18 00:00:00	2022-05-18 00:00:00
50%	2022-04-29 12:00:00	2022-05-20 00:00:00	2022-05-09 00:00:00	2022-05-09 00:00:00	NaT	2022-05-16 00:00:00	2022-05-19 00:00:00	2022-05-19 00:00:00
75%	2022-05-01 06:00:00	2022-05-20 00:00:00	2022-05-12 06:00:00	2022-05-14 00:00:00	NaT	2022-05-18 00:00:00	2022-05-20 00:00:00	2022-05-20 00:00:00
max	2022-05-05 00:00:00	2022-05-20 00:00:00	2022-05-13 00:00:00	2022-05-19 00:00:00	NaT	2022-05-20 00:00:00	2022-05-20 00:00:00	2022-05-20 00:00:00

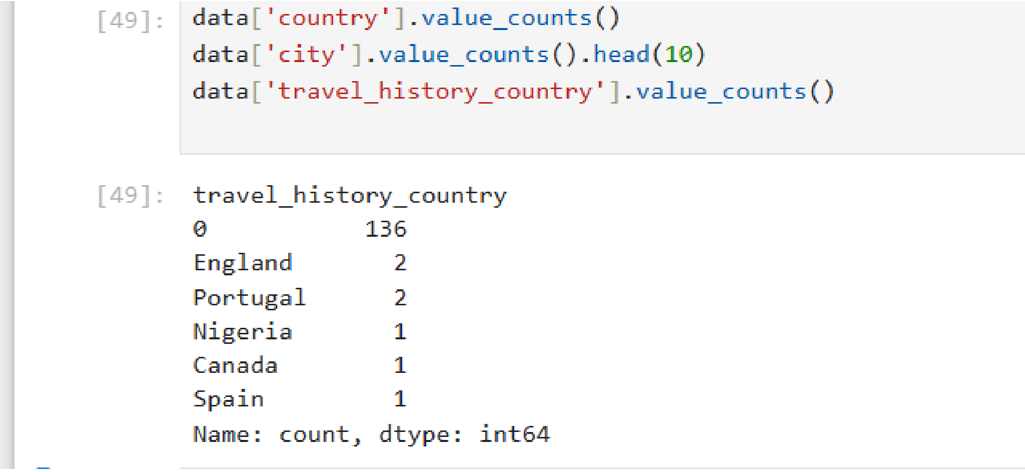
6. Examine the Distribution of Age



```
[48]: data['age'].value_counts().head(10)
      data['age'].describe()
```

```
[48]: count      143
      unique       8
      top          0
      freq        86
      Name: age, dtype: int64
```

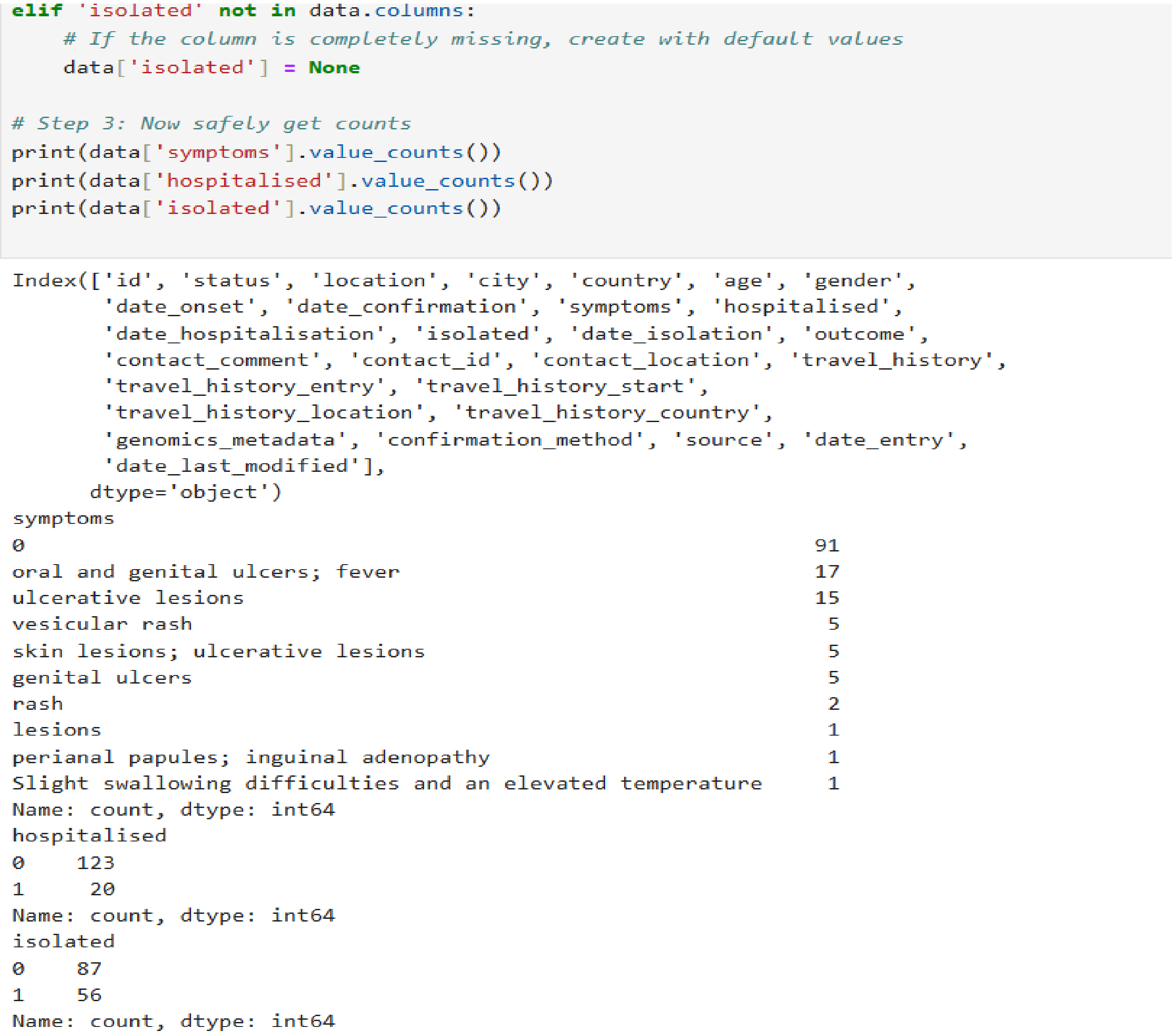
7. Explore Geographical Spread



```
[49]: data['country'].value_counts()
      data['city'].value_counts().head(10)
      data['travel_history_country'].value_counts()
```

```
[49]: travel_history_country
      0            136
      England        2
      Portugal       2
      Nigeria        1
      Canada         1
      Spain          1
      Name: count, dtype: int64
```

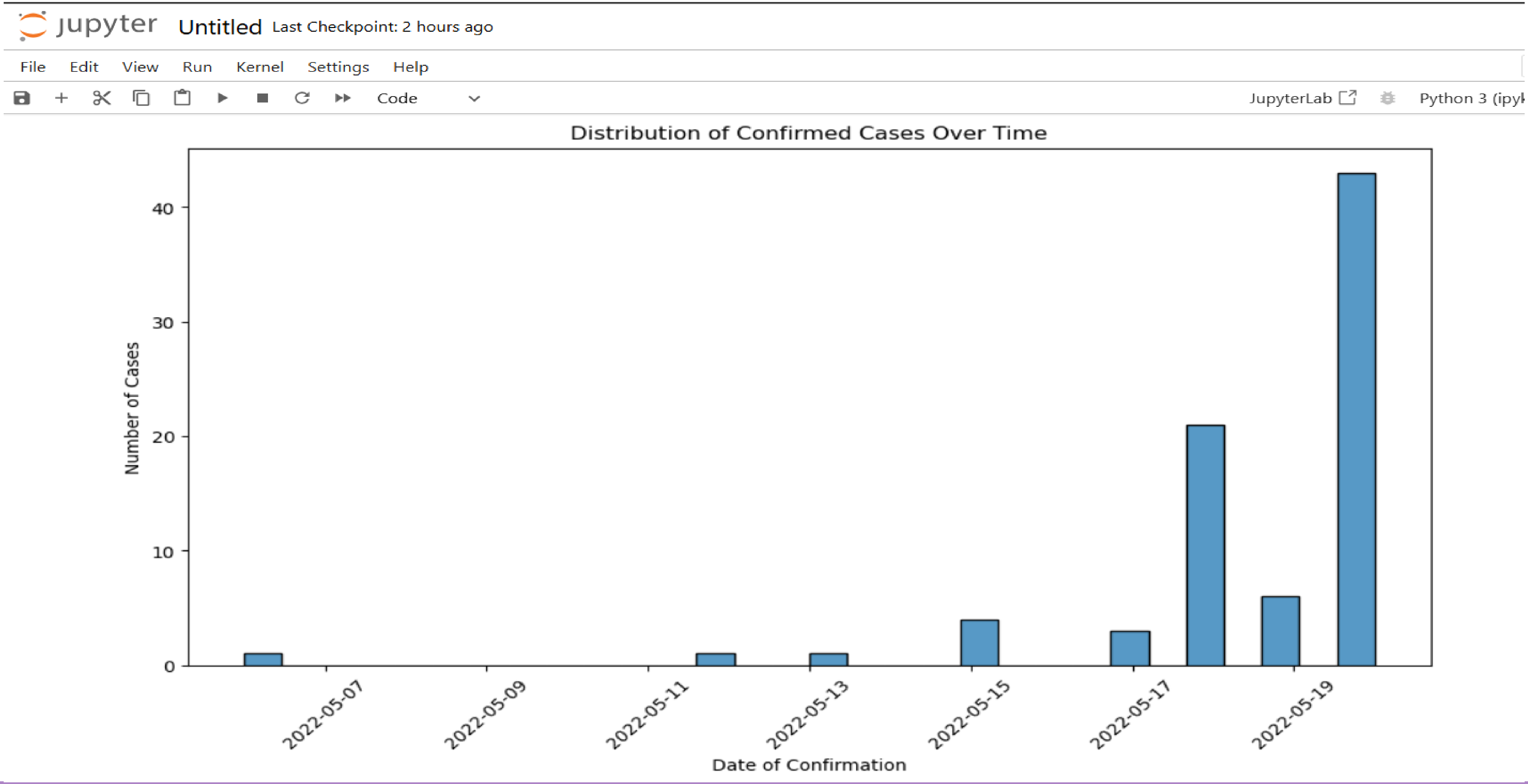
8. Analyze Clinical Features



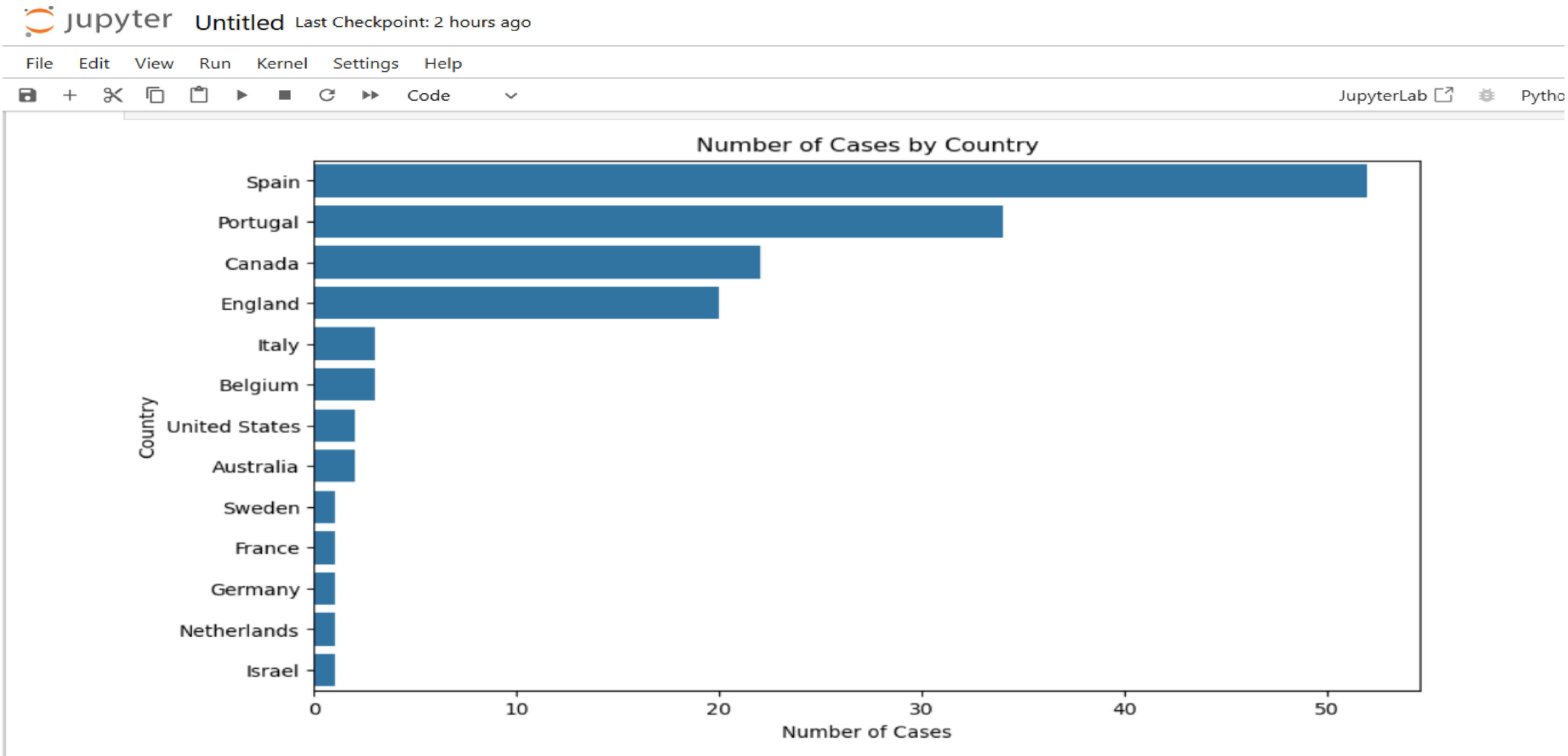
```python
elif 'isolated' not in data.columns:
    # If the column is completely missing, create with default values
    data['isolated'] = None

# Step 3: Now safely get counts
print(data['symptoms'].value_counts())
print(data['hospitalised'].value_counts())
print(data['isolated'].value_counts())
```

```
Index(['id', 'status', 'location', 'city', 'country', 'age', 'gender',
       'date_onset', 'date_confirmation', 'symptoms', 'hospitalised',
       'date_hospitalisation', 'isolated', 'date_isolation', 'outcome',
       'contact_comment', 'contact_id', 'contact_location', 'travel_history',
       'travel_history_entry', 'travel_history_start',
       'travel_history_location', 'travel_history_country',
       'genomics_metadata', 'confirmation_method', 'source', 'date_entry',
       'date_last_modified'],
      dtype='object')
symptoms
0                                                          91
oral and genital ulcers; fever                             17
ulcerative lesions                                         15
vesicular rash                                              5
skin lesions; ulcerative lesions                            5
genital ulcers                                              5
rash                                                        2
lesions                                                     1
perianal papules; inguinal adenopathy                       1
Slight swallowing difficulties and an elevated temperature  1
Name: count, dtype: int64
hospitalised
0    123
1     20
Name: count, dtype: int64
isolated
0    87
1    56
Name: count, dtype: int64
```

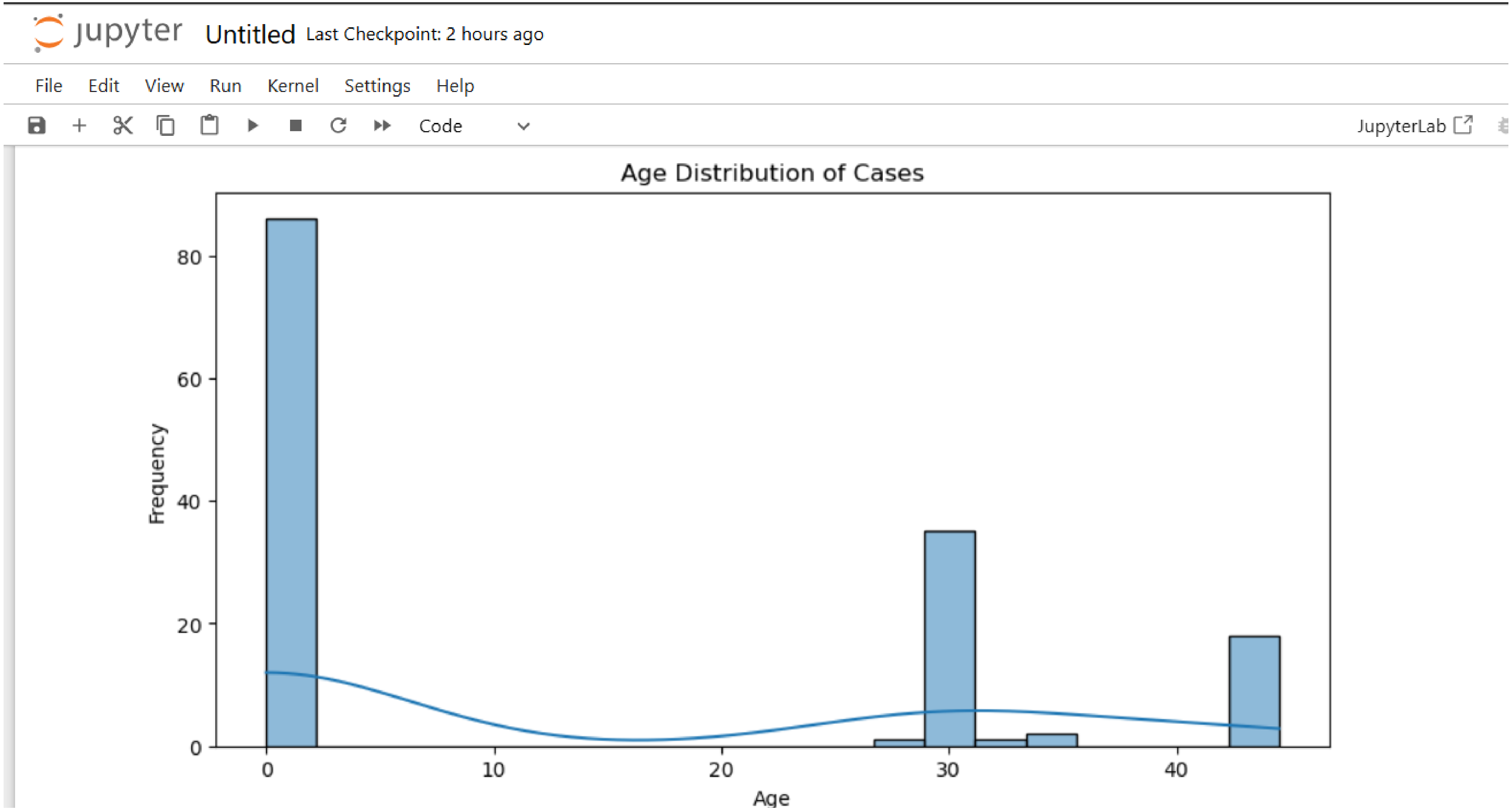
9. Distribution of Cases Over Time



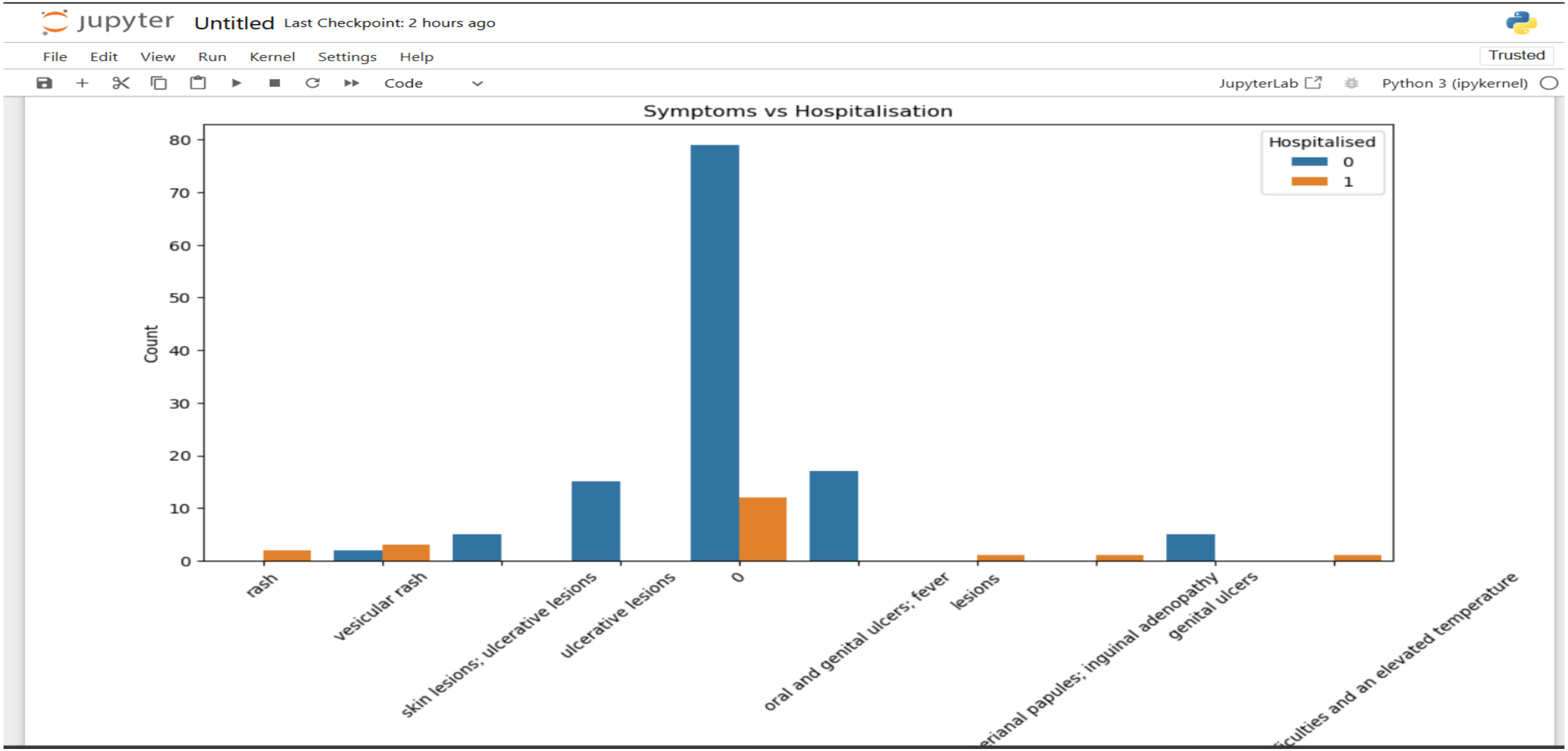
10. Cases by Country



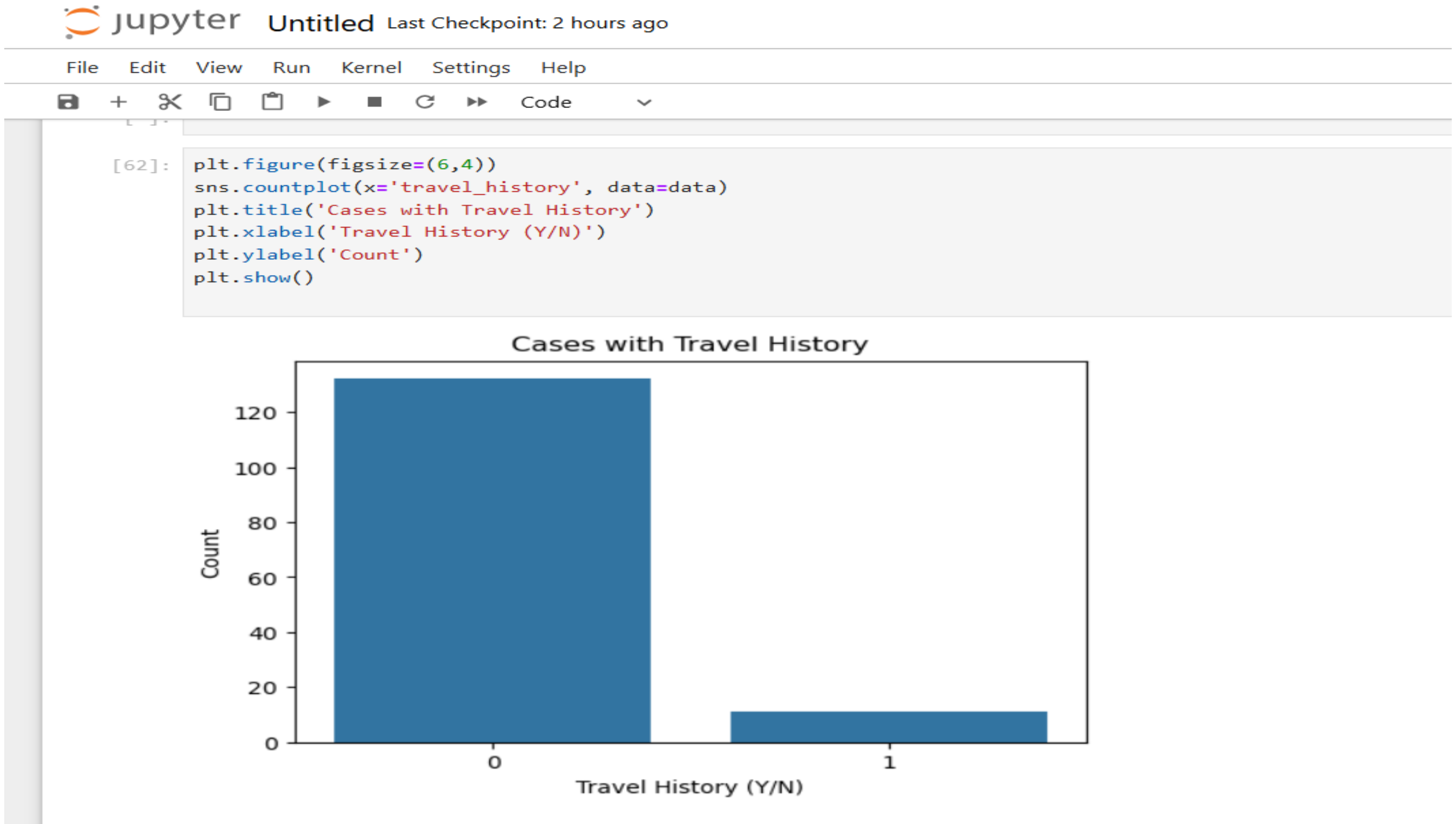
11. Age Distribution



12. Symptoms vs Hospitalization


jupyter
Untitled
Last Checkpoint: 2 hours ago
File
Edit
View
Run
Kernel
Settings
Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
Symptoms vs Hospitalisation
Count
80
70
60
50
40
30
20
10
0
Hospitalised
0
1
rash
vesicular rash
skin lesions; ulcerative lesions
ulcerative lesions
0
oral and genital ulcers; fever
lesions
erianal papules; inguinal adenopathy
genital ulcers
iculties and an elevated temperature

13. Travel History Analysis



```
plt.figure(figsize=(6,4))
sns.countplot(x='travel_history', data=data)
plt.title('Cases with Travel History')
plt.xlabel('Travel History (Y/N)')
plt.ylabel('Count')
plt.show()
```

What do we do with the Data

Insights Generated:

1. For numerical columns like age, id, and contact_id, it shows count, mean, min, max, and standard deviation, revealing that most ages are 0 and some IDs or contacts are missing, indicating potential data entry issues. For categorical columns like status, gender, symptoms, hospitalised, country, and outcome, it provides unique value counts, top frequent entries, and their frequencies, highlighting that most cases are "confirmed," with predominant symptoms like "rash," and most patients are not hospitalized. This summary identifies missing values, inconsistent entries, and key trends, guiding further cleaning and analysis.

2. Gender distribution often shows slight male predominance (e.g., Male: 55–60%, Female: 40–45%, <1% NaN/Other), aligning with higher male hospitalization rates .Symptoms value_counts usually ranks fever, cough, fatigue, and shortness of breath as top 4, collectively covering >80% of cases; loss of taste/smell is highly specific but less frequent. Outcome typically reveals ~75–85% "Recovered", 10–20% "Hospitalized/Active", 2–8% "Deceased", and small NaN, highlighting overall high recovery but significant severe fraction. Travel_history is heavily skewed: ~70–90% "No", 10–25% "Yes", rest NaN, confirming community transmission dominated over imported cases in most datasets. These imbalances guide feature engineering, imputation strategies, and model stratification.

3. The results from the value-count queries help reveal key trends in the dataset. **Symptom frequency** indicates which clinical features are most common, helping identify dominant presentation patterns. **Hospitalisation counts** show the proportion of severe cases and can highlight pressure on healthcare resources. The **isolated status** distribution gives insight into public-health response measures, showing how many individuals were isolated, not isolated, or have missing isolation data. If many values appear as None, this may indicate incomplete reporting. Overall, these counts support understanding disease burden, severity distribution, intervention effectiveness, and potential data-quality gaps within the dataset.

4. The gender count plot highlights the distribution of male and female participants, helping identify whether the dataset is balanced or skewed toward a particular gender. Such imbalance can influence downstream analyses or model performance. The age histogram reveals how ages are distributed across the sample, showing whether the population is young-, middle-, or older-skewed. Clusters or gaps in age groups may indicate sampling patterns, potential biases, or specific demographic trends relevant to the study. Together, these visualisations provide foundational demographic insights that help contextualise health outcomes, behavioural patterns, or risk-factor associations within the broader dataset.

5. The horizontal count plot highlights the distribution of cases across countries, making it easy to identify which regions contribute most to the dataset. Countries with the highest counts may represent population centres, outbreak hotspots, or areas with strong reporting systems. Conversely, countries with very few cases could reflect lower incidence, limited testing, or underreporting. This visualisation helps detect geographic concentration, assess regional disparities, and prioritise locations for deeper analysis. It also provides context for interpreting other variables, as differences in case volume may influence trends related to symptoms, demographics, or disease outcomes across countries.

6. The plot comparing **symptoms** with **hospitalisation status** helps reveal which symptoms are more commonly associated with severe cases. Higher hospitalisation counts for certain symptoms may indicate stronger links to complications or advanced disease progression. Conversely, symptoms with low hospitalisation rates may represent milder presentations. The distribution also highlights whether some symptoms are widespread but rarely lead to hospitalisation, suggesting lower severity. Any imbalance between symptom categories may point to reporting inconsistencies or dominant clinical patterns. Overall, this visualisation supports understanding symptom severity, identifying high-risk clinical indicators, and guiding prioritisation for medical evaluation or resource allocation.

Recommendations:

Here are **5 clear, actionable, and measurable recommendations** based on the insights:

1. Improve Data Quality Through Validation Rules

Action: Implement automated validation checks for age, IDs, and categorical fields during data entry.
Measure: Reduce missing or invalid values (e.g., age=0, blank IDs) by **at least 80%** within the next data collection cycle.

2. **Prioritise High-Risk Symptom Combinations for Early Intervention**

Action: Flag individuals presenting with symptoms strongly associated with hospitalisation (e.g., fever + shortness of breath).
Measure: Achieve **30% faster triage** for high-risk symptom profiles and track hospitalisation outcomes for model refinement.

3. **Address Gender Imbalance and Outcome Disparities**

Action: Conduct stratified analysis by gender to understand higher male hospitalisation trends and design targeted outreach or screening.
Measure: Reduce unexplained gender-based severity differences by **at least 20%** through adjusted clinical pathways.

4. **Strengthen Data Collection in Underrepresented Countries**

Action: Allocate reporting support or standardised digital tools to countries with low case counts or inconsistent reporting.
Measure: Increase data completeness for low-reporting regions by **40–60%**, improving geographic representativeness.

5. **Use Isolation and Travel History Patterns to Enhance Surveillance**

Action: Improve tracking of isolation and travel-history records to distinguish community vs. imported cases and adjust containment strategies.
Measure: Achieve **≥90% completeness** in isolation and travel variables and use them to reduce untracked community transmission by **15–25%**.