covid_analysis_starter

May 11, 2025

1 COVID-19 Data Analysis Starter Notebook

This notebook uses the Our World in Data (OWID) COVID-19 dataset.

```python
# Step 1: Import Libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Optional for interactive visualizations
# import plotly.express as px
```

```python
# Step 2: Load Dataset
df = pd.read_csv('data/owid-covid-data.csv')
```

```python
# Step 3: Explore Data
print("Columns:\n", df.columns)
print("\nData types:\n", df.dtypes)
print("\nMissing values:\n", df.isnull().sum())

# Preview dataset
df.head()
```

```python
# Step 4: Clean Data
# Keep selected countries
countries = ['Kenya', 'United States', 'India']
df = df[df['location'].isin(countries)]

# Convert date to datetime
df['date'] = pd.to_datetime(df['date'])

# Drop rows with missing critical values
df = df.dropna(subset=['total_cases', 'total_deaths'])

# Fill or interpolate other values
df['total_vaccinations'] = df['total_vaccinations'].interpolate()
```

```python
# Step 5: EDA - Cases Over Time
plt.figure(figsize=(12, 6))
for country in countries:
    subset = df[df['location'] == country]
    plt.plot(subset['date'], subset['total_cases'], label=country)

plt.title('Total COVID-19 Cases Over Time')
plt.xlabel('Date')
plt.ylabel('Total Cases')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

```python
# Step 6: Daily New Cases Comparison
plt.figure(figsize=(12, 6))
for country in countries:
    subset = df[df['location'] == country]
    plt.plot(subset['date'], subset['new_cases'], label=country)

plt.title('Daily New COVID-19 Cases')
plt.xlabel('Date')
plt.ylabel('New Cases')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

```python
# Step 7: Vaccination Progress
plt.figure(figsize=(12, 6))
for country in countries:
    subset = df[df['location'] == country]
    plt.plot(subset['date'], subset['total_vaccinations'], label=country)

plt.title('Total Vaccinations Over Time')
plt.xlabel('Date')
plt.ylabel('Total Vaccinations')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

```python
# Step 8: Calculate Death Rate
df['death_rate'] = df['total_deaths'] / df['total_cases']

# Plot death rate (rolling average for smoothness)
plt.figure(figsize=(12, 6))
```

```python
for country in countries:
    subset = df[df['location'] == country].copy()
    subset['death_rate_smoothed'] = subset['death_rate'].rolling(window=7).
 ↪mean()
    plt.plot(subset['date'], subset['death_rate_smoothed'], label=country)

plt.title('Death Rate (7-Day Rolling Avg)')
plt.xlabel('Date')
plt.ylabel('Death Rate')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```