Exploratory Data Analysis of COVID-19 Cases

Vishnu G Nath

vishnugnath

vishnugs32@gmail.com

AGENDA

- Introduction
- Objective
- Dataset Overview
- Data Cleaning and Preparation
- Descriptive Statistics
- Data analysis & Visualizations
- Performance Insights
- Key Recommendations

01

Introduction

In the wake of the COVID-19 pandemic, understanding the trends and impacts of the virus across different regions is crucial for effective response and policymaking. This project aims to perform an Exploratory Data Analysis (EDA) on COVID-19 case data to uncover significant patterns, trends, and insights. By analyzing this data, we can gain a deeper understanding of how the virus has spread, the effectiveness of recovery efforts, and the disparities between various regions and countries.

02

Objective

The primary objective of this analysis is to:

- Analyze and visualize COVID-19 case data .
- Identify trends, patterns, and insights across different countries and WHO regions.
- Provide actionable insights to inform public health policies and interventions.

03

Dataset Overview

The dataset is sourced from Kaggle.
'country_wise_latest'

Link : https://www.kaggle.com/datasets/armanmanteghi/covid-19-data-statistics-sql/data

Key Columns:

Country/Region: Name of the country or region.
Confirmed: Total confirmed cases.
Deaths: Total deaths.
Recovered: Total recovered cases.
Active: Active cases.
New_cases: Newly reported cases.
New_deaths: Newly reported deaths.
New_recovered: Newly recovered cases.
Deaths_/_100_Cases: Deaths per 100 confirmed cases.
Recovered_/_100_Cases: Recoveries per 100 confirmed cases.
Deaths_/_100_Recovered: Deaths per 100 recovered cases.
Confirmed_last_week: Confirmed cases from the last week.
1_week_change: Change in confirmed cases over the past week.
1_week_%_increase: Percentage increase in cases over the past week.
WHO_Region: WHO region classification.

04

Data Cleaning and Preparation

Loading Data

The dataset was loaded using pandas, and an initial inspection was performed to understand its structure.

Data Cleaning

Renamed Columns: Columns were renamed to replace spaces with underscores for consistency.Handled Infinite Values: Replaced infinite values with NaN.Handled Missing Values: Replaced missing values with zeros.

```
# Loading the dataset
df = pd.read_csv('covid_19_data.csv')

# Renaming columns
df.columns = [col.replace(' ', '_') for col in df.columns]

# Replacing infinite values with NaN
df.replace([np.inf, -np.inf], np.nan, inplace=True)

# Replacing missing values with zeros
df.fillna(0, inplace=True)
```

05

Descriptive Statistics

Summary Statistics

Confirmed Cases:

- Mean: 88,130.94
- Median: 5,059
- Min: 10
- Max: 4,290,259

Deaths

- Mean: 3,497.52
- Median: 108
- Min: 0
- Max: 148,011

Recovered Cases

- Mean: 50,631.48
- Median: 2,815
- Min: 0
- Max: 1,846,641

New Cases

- Mean: 1,222.96
- Median: 49
- Min: 0
- Max: 56,336

Data Types

# Summary statistics
df.describe()

# Verifying data types
df.dtypes

```
  Country/Region  Confirmed  Deaths  Recovered  Active  New_cases  New_deaths  \
0    Afghanistan      36263    1269      25198    9796        106          10
1        Albania       4880     144       2745    1991        117           6
2        Algeria      27973    1163      18837    7973        616           8
3        Andorra        907      52        803      52         10           0
4         Angola        950      41        242     667         18           1

   New_recovered  Deaths_/_100_Cases  Recovered_/_100_Cases  \
0             18                3.50                  69.49
1             63                2.95                  56.25
2            749                4.16                  67.34
3              0                5.73                  88.53
4              0                4.32                  25.47

   Deaths_/_100_Recovered  Confirmed_last_week  1_week_change  \
0                    5.04                35526            737
1                    5.25                 4171            709
2                    6.17                23691           4282
3                    6.48                  884             23
4                   16.94                  749            201

   1_week_%_increase             WHO_Region
0               2.07  Eastern Mediterranean
1              17.00                 Europe
2              18.07                 Africa
3               2.60                 Europe
4              26.84                 Africa
```

```
Data columns (total 15 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Country/Region          187 non-null    object
 1   Confirmed               187 non-null    int64
 2   Deaths                  187 non-null    int64
 3   Recovered               187 non-null    int64
 4   Active                  187 non-null    int64
 5   New_cases               187 non-null    int64
 6   New_deaths              187 non-null    int64
 7   New_recovered           187 non-null    int64
 8   Deaths_/_100_Cases      187 non-null    float64
 9   Recovered_/_100_Cases   187 non-null    float64
 10  Deaths_/_100_Recovered  187 non-null    float64
 11  Confirmed_last_week     187 non-null    int64
 12  1_week_change           187 non-null    int64
 13  1_week_%_increase       187 non-null    float64
 14  WHO_Region              187 non-null    object
dtypes: float64(4), int64(9), object(2)
memory usage: 22.0+ KB
None
```

06

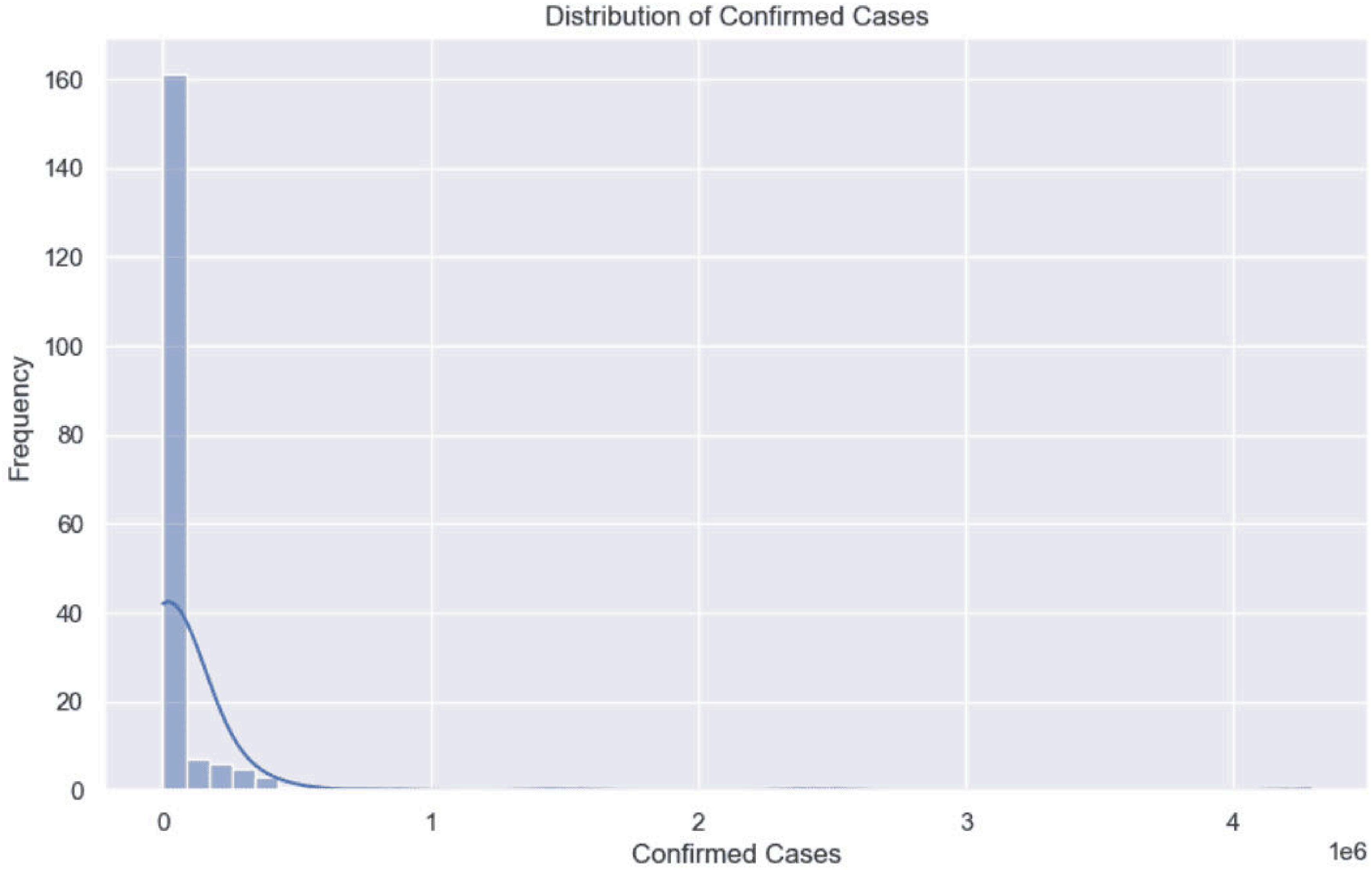
Data analysis & Visualizations

Univariate Analysis

Distribution of Confirmed Cases

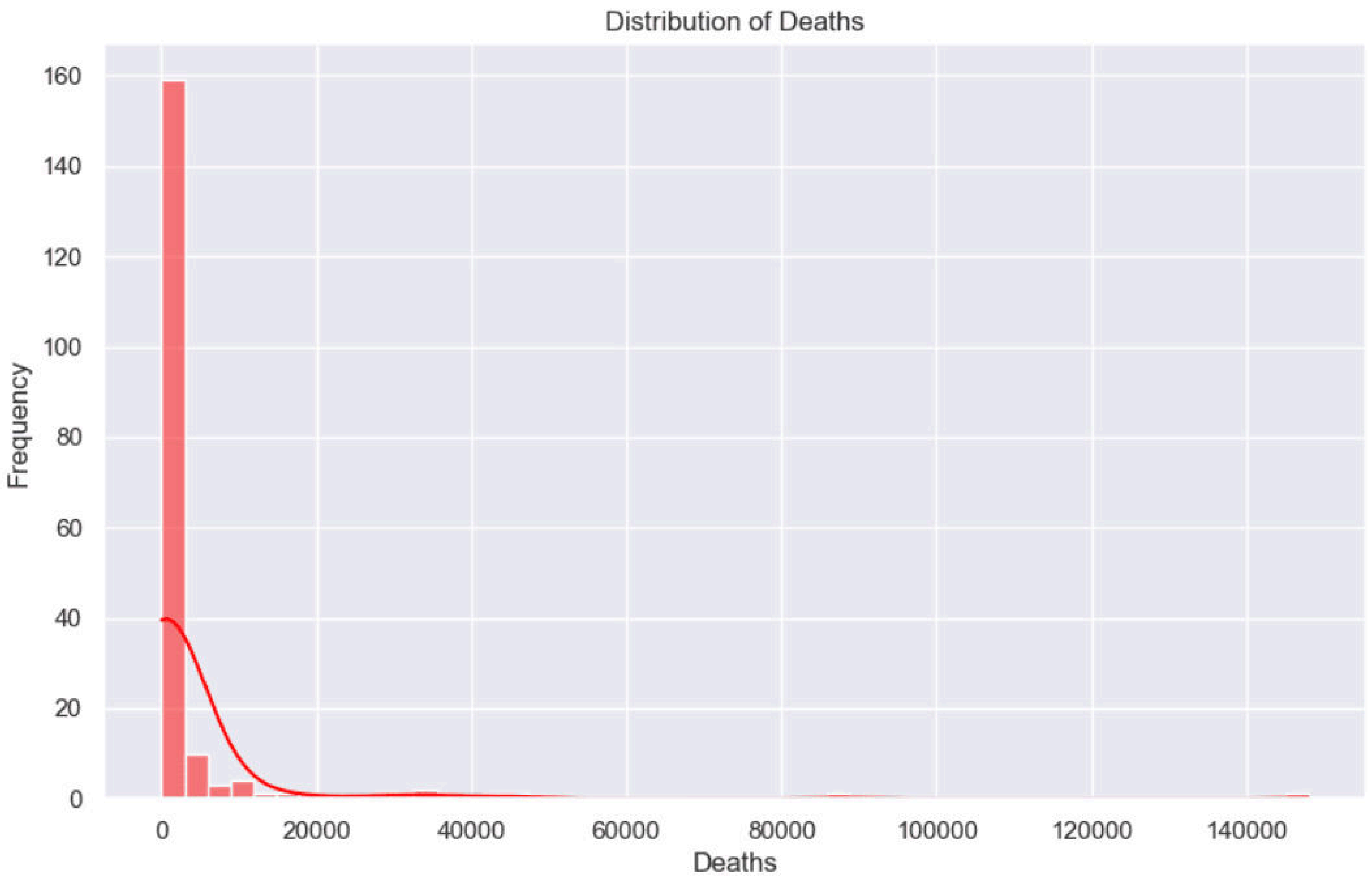
- Histogram: Created to visualize the distribution of confirmed cases across countries.
- Key Findings: Most countries have a moderate number of confirmed cases, with a few countries having extremely high counts.

```
plt.figure(figsize=(10, 6))
sns.histplot(df['Confirmed'], bins=50,
kde=True, color='blue')
plt.title('Distribution of Confirmed
Cases')
plt.xlabel('Confirmed Cases')
plt.ylabel('Frequency')
plt.show()
```



Distribution of Deaths

- Histogram: Showing the distribution of deaths..
- Key Findings: Similar to confirmed cases, there are a few countries with very high death counts.



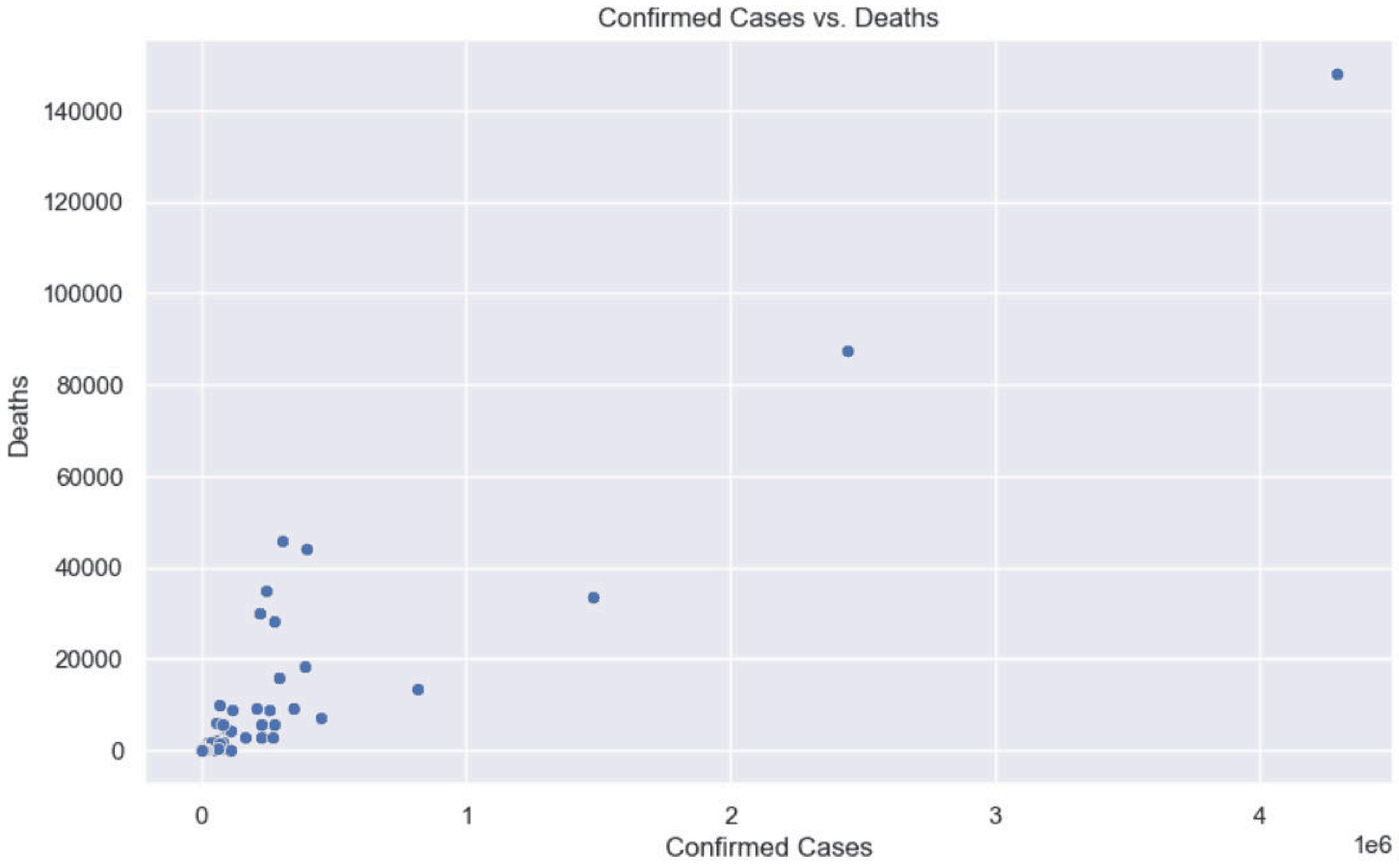
```
plt.figure(figsize=(10, 6))
sns.histplot(df['Deaths'], bins=50,
kde=True, color='red')
plt.title('Distribution of Deaths')
plt.xlabel('Deaths')
plt.ylabel('Frequency')
plt.show()
```

Bivariate Analysis

Confirmed Cases vs. Deaths

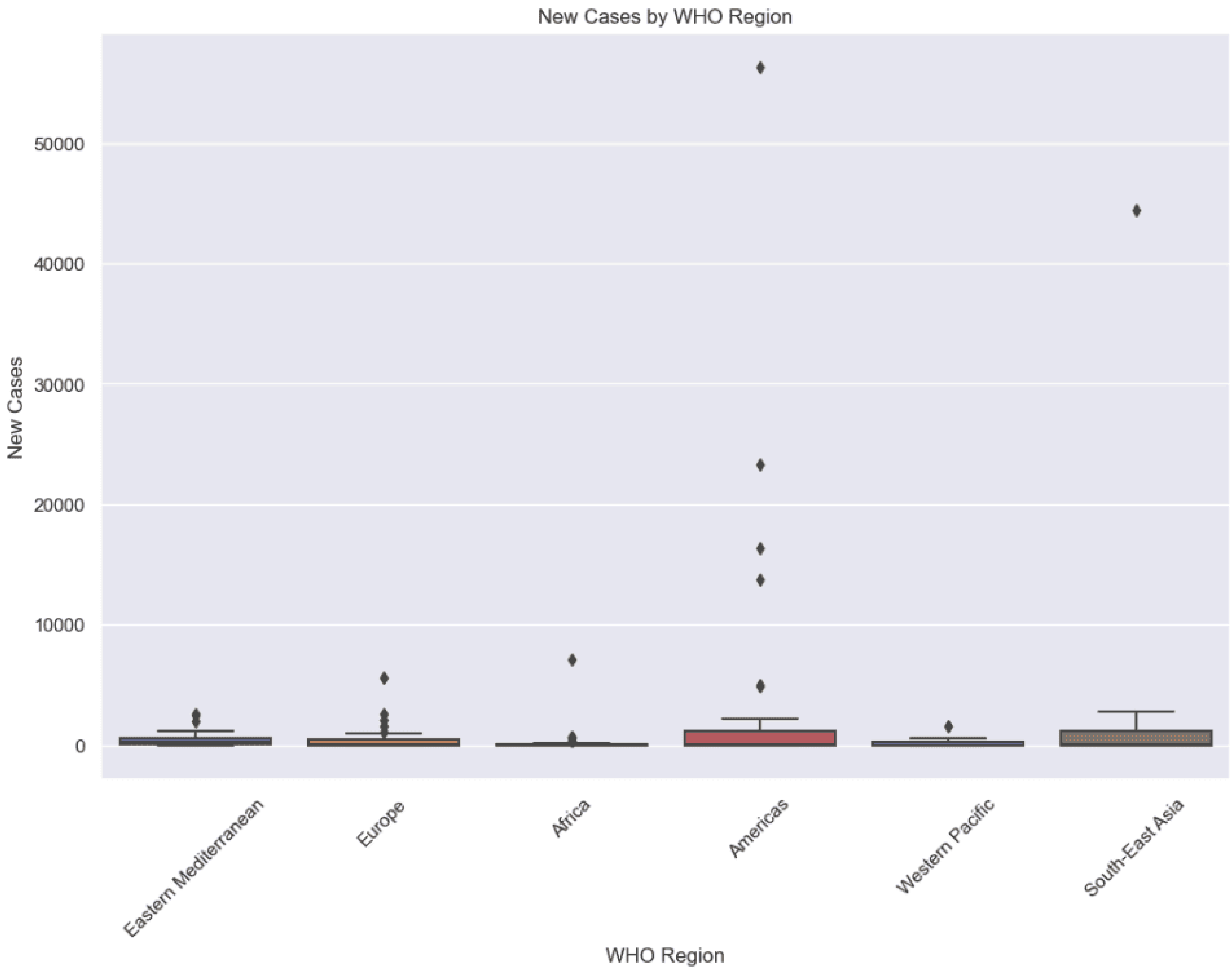
- Scatter Plot: To visualize the relationship between confirmed cases and deaths.
- Key Findings: Positive correlation observed; countries with more confirmed cases tend to have higher death counts.

```
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Confirmed',
y='Deaths', data=df)
plt.title('Confirmed Cases vs. Deaths')
plt.xlabel('Confirmed Cases')
plt.ylabel('Deaths')
plt.show()
```



New Cases by WHO Region

- Box Plot: Shows the distribution of new cases by WHO region.
- Key Findings: Significant variation in new cases across regions; some regions have higher median values.

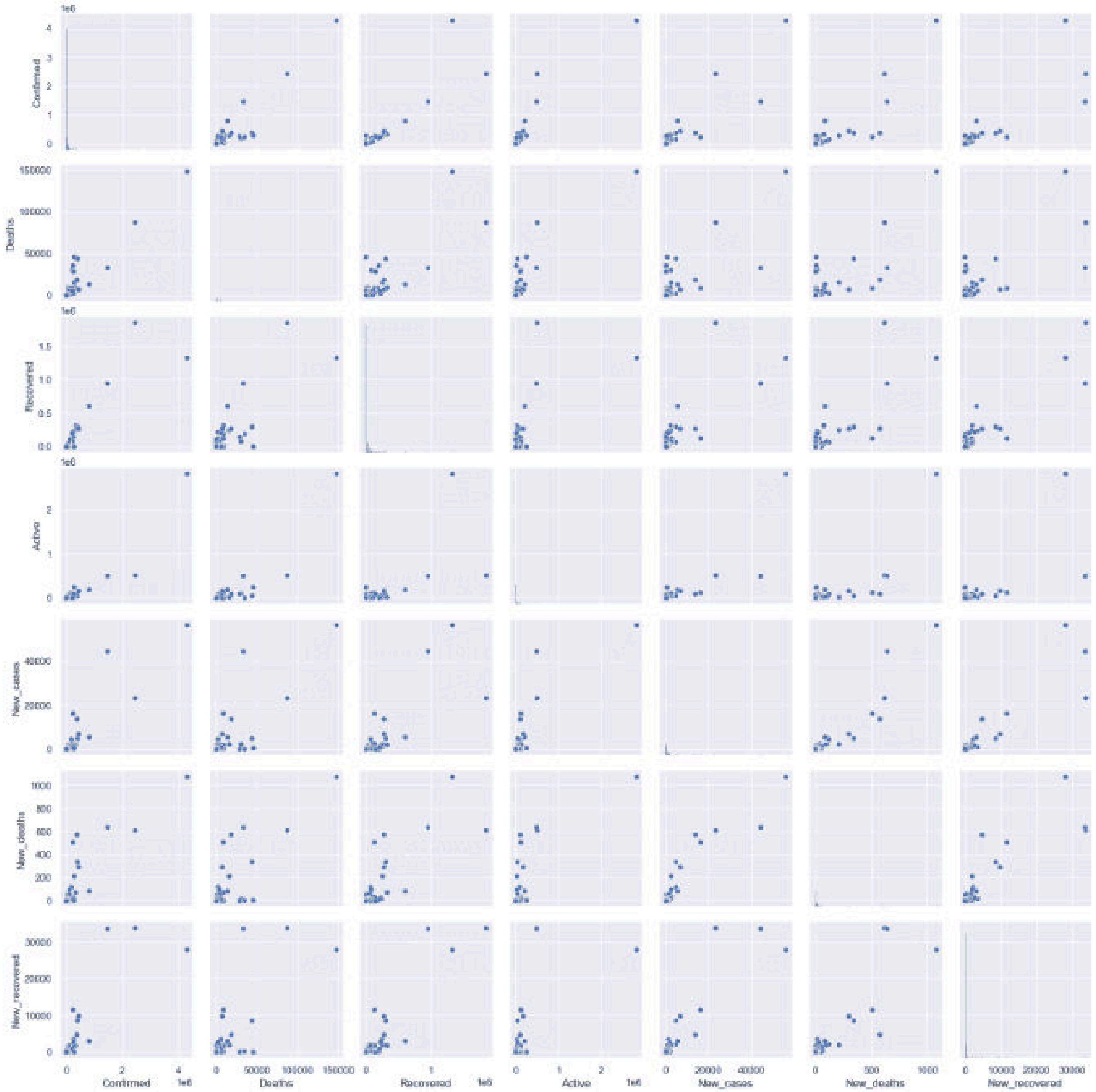


```
plt.figure(figsize=(12, 8))
sns.boxplot(x='WHO_R
egion',    y='New_cases',
data=df)    plt.title('New
Cases by WHO Region')
plt.xlabel('WHO
Region') plt.ylabel('New
Cases')
plt.xticks(rotation=45)
plt.show()
```

Pairwise Relationships

Pairplot

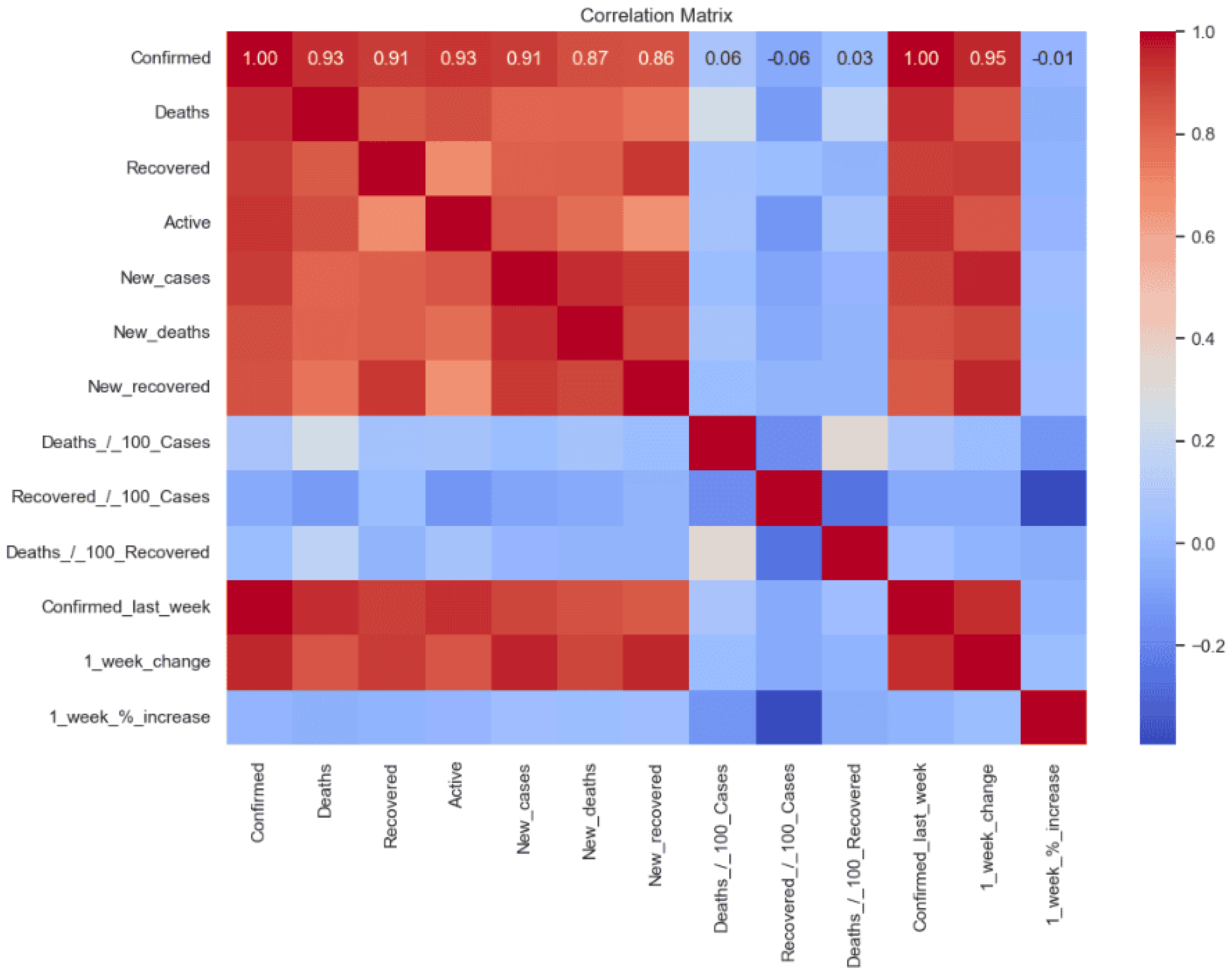
sns.pairplot(df[['Confirmed', 'Deaths', 'Recovered', 'Active', 'New_cases', 'New_deaths', 'New_recovered']]) plt.show()



- Visualizes relationships between multiple numerical features: Including confirmed cases, deaths, recovered cases, etc.
- Key Findings: Helps to identify relationships and potential correlations between different metrics.

Correlation Analysis

Correlation Matrix



- Heatmap: Displays correlations between numerical features.
- Key Findings: High correlation between confirmed cases and active cases; moderate correlation between deaths and confirmed cases.

```
plt.figure(figsize=(12, 8))
sns.heatmap(df.select_dtypes(i
nclude=[np.number]).corr(),
annot=True, cmap='coolwarm',
fmt='.2f') plt.title('Correlation
Matrix') plt.show()
```

Aggregated Data Analysis

Total Cases and Rates

- Total Confirmed Cases: 16,480,485
- Total Deaths: 654,036Total
- Recovered: 9,468,087
- Death Rate: 3.97%
- Recovery Rate: 57.45%

```
total_confirmed = df['Confirmed'].sum()
total_deaths = df['Deaths'].sum()
total_recovered = df['Recovered'].sum()

death_rate = (total_deaths / total_confirmed) *
100
recovery_rate = (total_recovered /
total_confirmed) * 100

print(f'Total Confirmed Cases:
{total_confirmed}')
print(f'Total Deaths: {total_deaths}')
print(f'Total Recovered: {total_recovered}')
print(f'Death Rate: {death_rate:.2f}%')
print(f'Recovery Rate: {recovery_rate:.2f}%')
```

Top 10 Countries by Confirmed Cases

- US: 4,290,259
- Brazil: 2,442,375
- India: 1,480,073
- Russia: 816,680
- South Africa: 452,529
- Mexico: 395,489
- Peru: 389,717
- Chile: 347,923
- United Kingdom: 301,708
- Iran: 293,606

```
top_10_countries = df.nlargest(10, 'Confirmed')
[['Country/Region', 'Confirmed']] print('Top 10
Countries with Highest Confirmed Cases:')
print(top_10_countries)
```

Aggregate New Cases

- Total New Cases: 228693
- Total New Deaths: 5415
- Total New Recovered: 174623

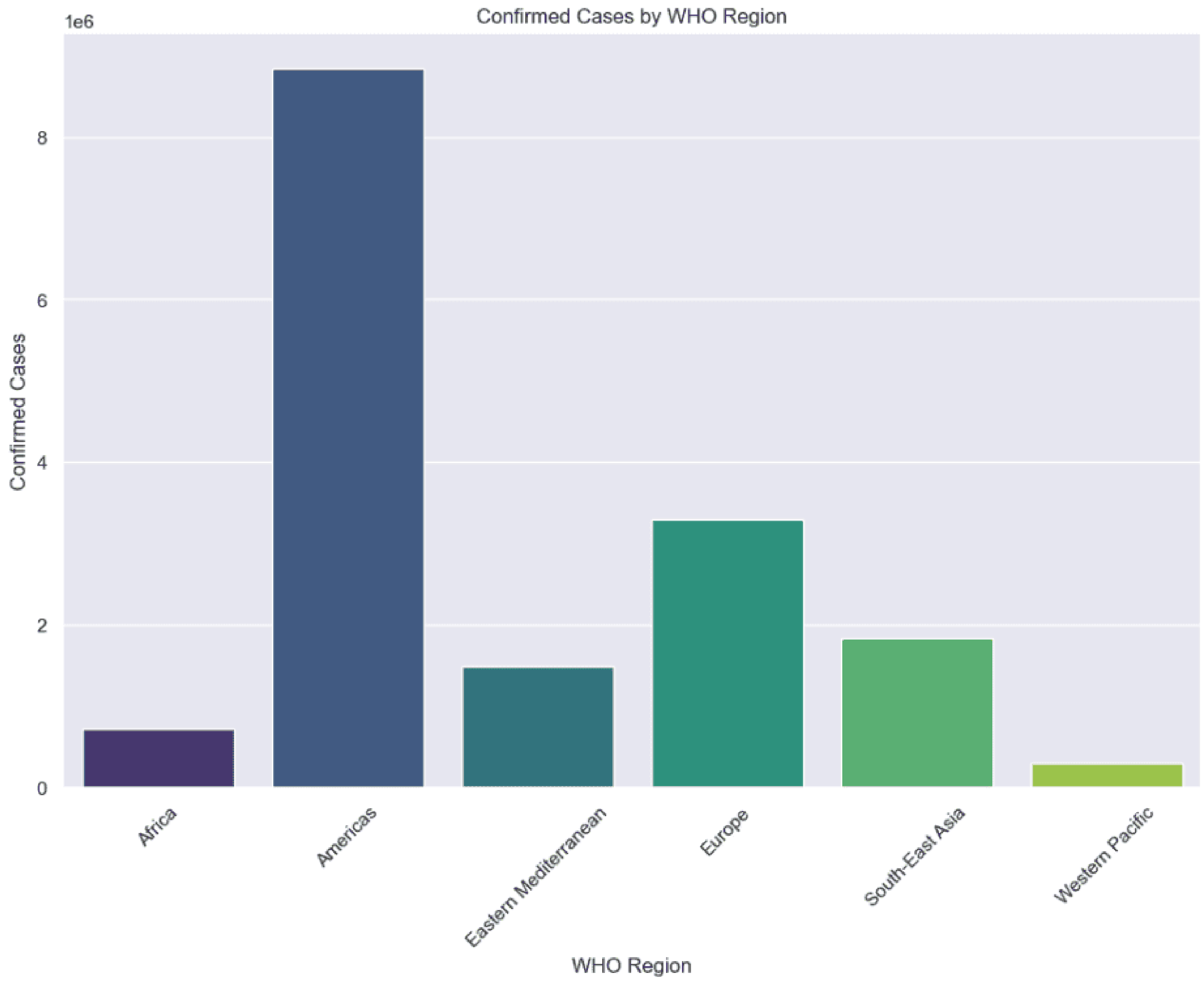
```
Data total_new_cases = df['New_cases'].sum()
total_new_deaths = df['New_deaths'].sum()
total_new_recovered =
df['New_recovered'].sum() print(f'Total New
Cases: {total_new_cases}') print(f'Total New
Deaths: {total_new_deaths}') print(f'Total New
Recovered: {total_new_recovered}')
```

Regional Analysis

Confirmed Cases by WHO Region

- Bar Plot: Visualizes confirmed cases by WHO region.
- Key Findings: The Americas region has the highest number of confirmed cases.

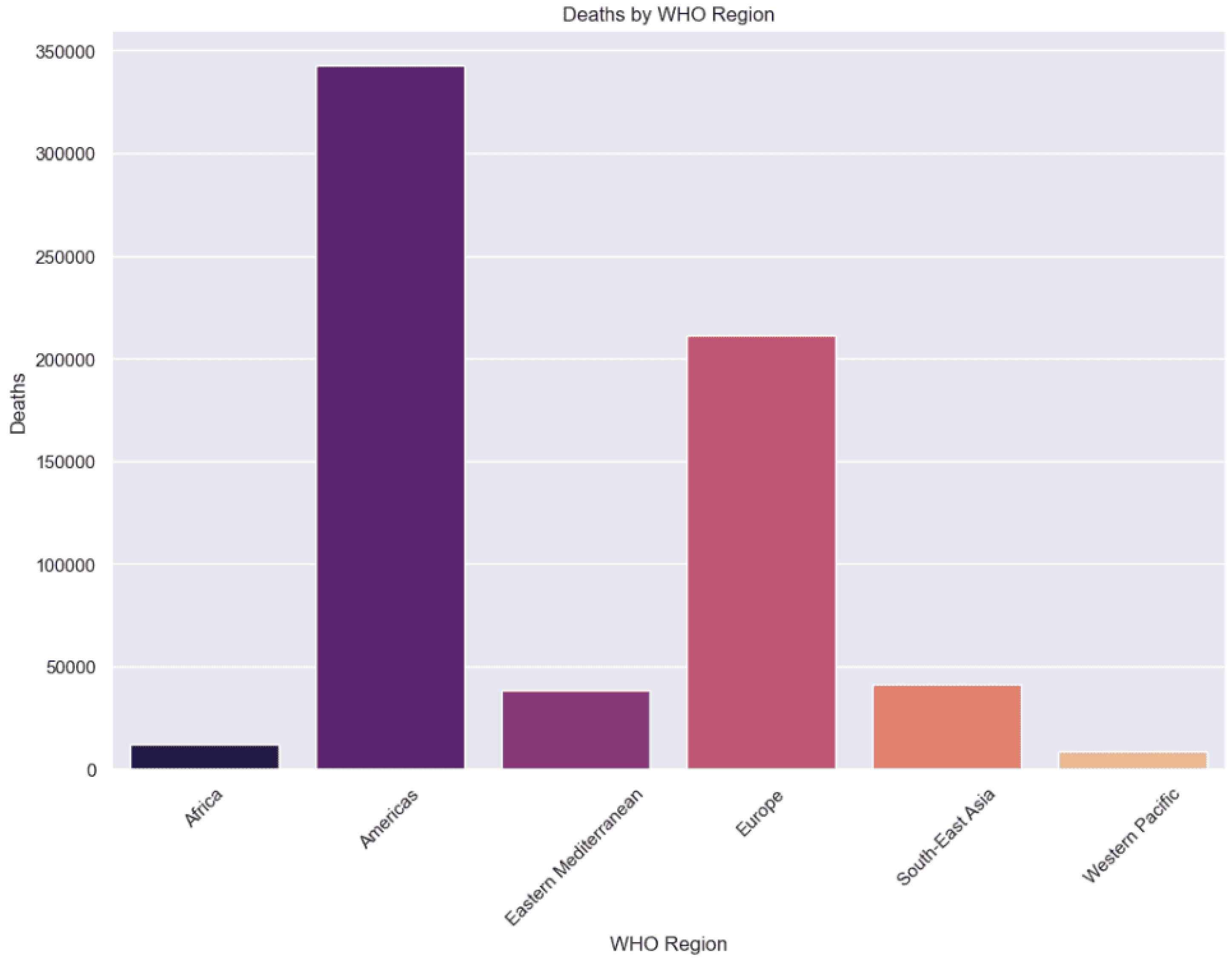
```
region_confirmed = df.groupby('WHO_Region')
['Confirmed'].sum().reset_index()
plt.figure(figsize=(12, 8))
sns.barplot(x='WHO_Region', y='Confirmed',
data=region_confirmed, palette='viridis')
plt.title('Confirmed Cases by WHO Region')
plt.xlabel('WHO Region') plt.ylabel('Confirmed
Cases') plt.xticks(rotation=45) plt.show()
```



Deaths by WHO Region

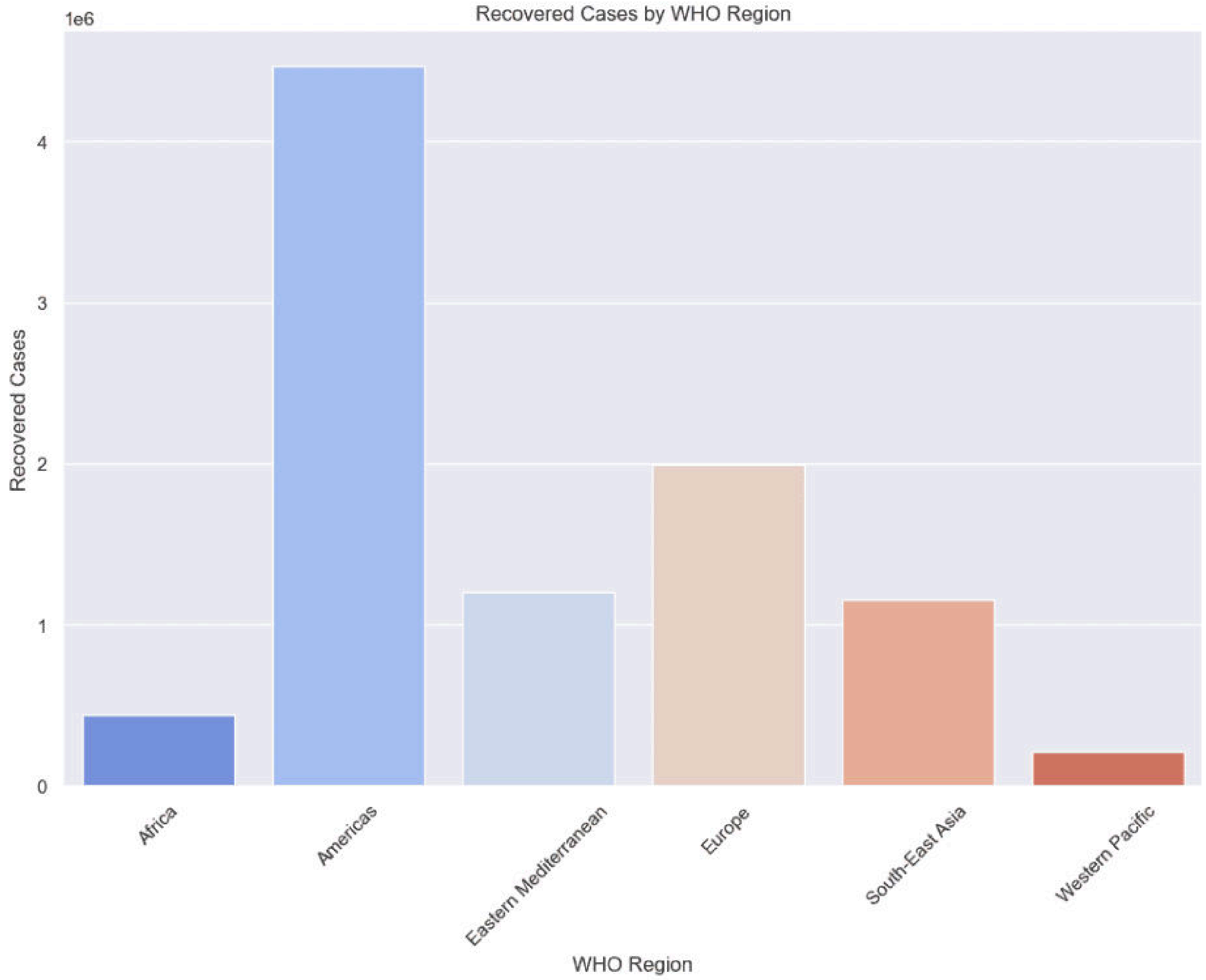
- Bar Plot: Visualizes deaths by WHO region.
- Key Findings: Europe and the Americas have the highest death counts.

```
region_deaths = df.groupby('WHO_Region')
['Deaths'].sum().reset_index() plt.figure(figsize=
(12, 8)) sns.barplot(x='WHO_Region', y='Deaths',
data=region_deaths, palette='magma')
plt.title('Deaths by WHO Region')
plt.xlabel('WHO Region') plt.ylabel('Deaths')
plt.xticks(rotation=45) plt.show()
```



Recovered Cases by WHO Region

- Bar Plot: Shows recovered cases by WHO region.
- Key Findings: The Americas region also leads in recovered cases.



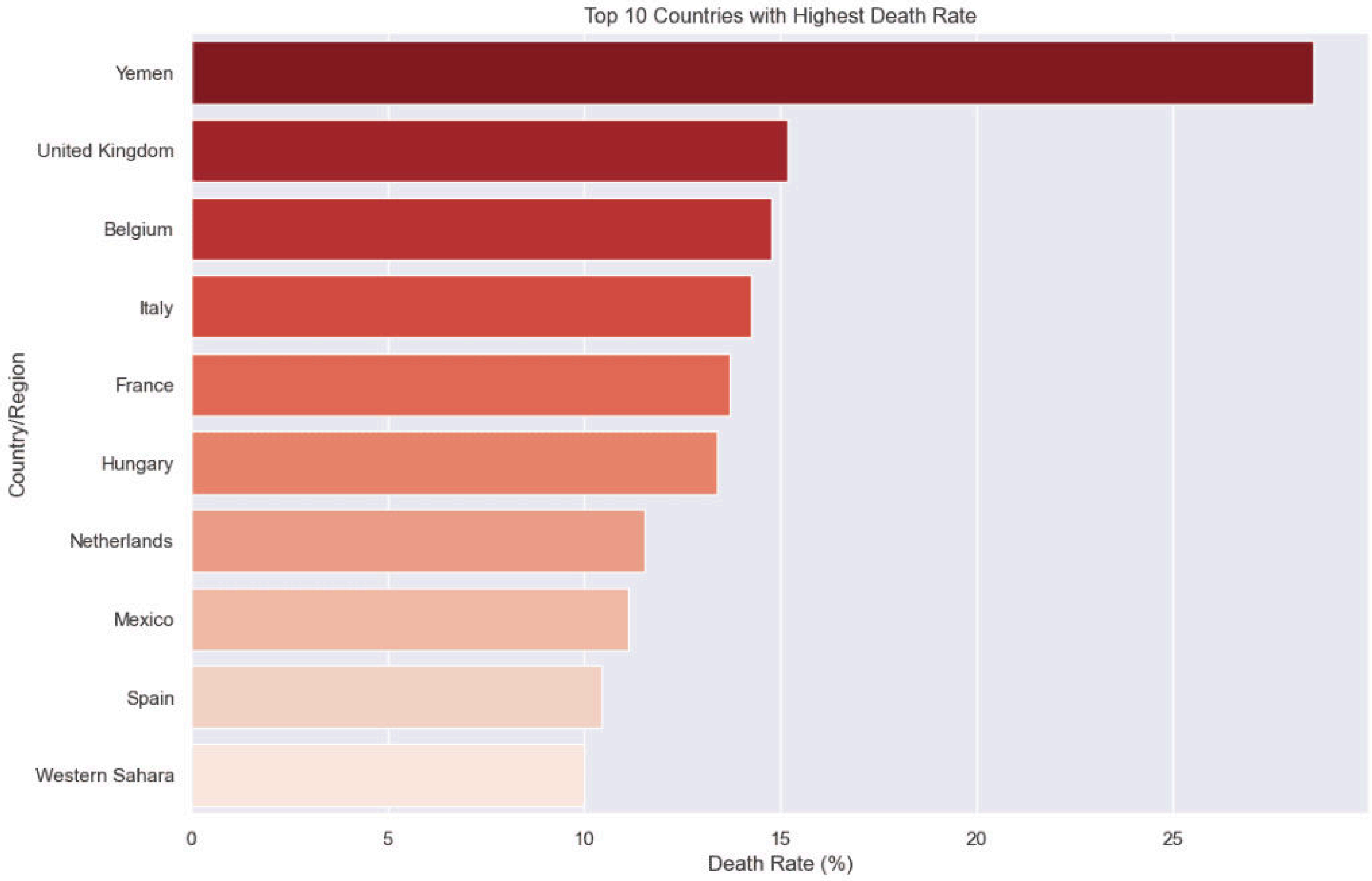
```
region_recovered = df.groupby('WHO_Region')
['Recovered'].sum().reset_index()
plt.figure(figsize=(12, 8))
sns.barplot(x='WHO_Region', y='Recovered',
data=region_recovered, palette='coolwarm')
plt.title('Recovered Cases by WHO Region')
plt.xlabel('WHO Region') plt.ylabel('Recovered
Cases') plt.xticks(rotation=45) plt.show()
```

Advanced Metrics

Death Rate by Country

- Calculation: (Deaths / Confirmed) * 100
- Top 10 Countries with Highest Death Rate: Detailed list of countries with the highest death rates.

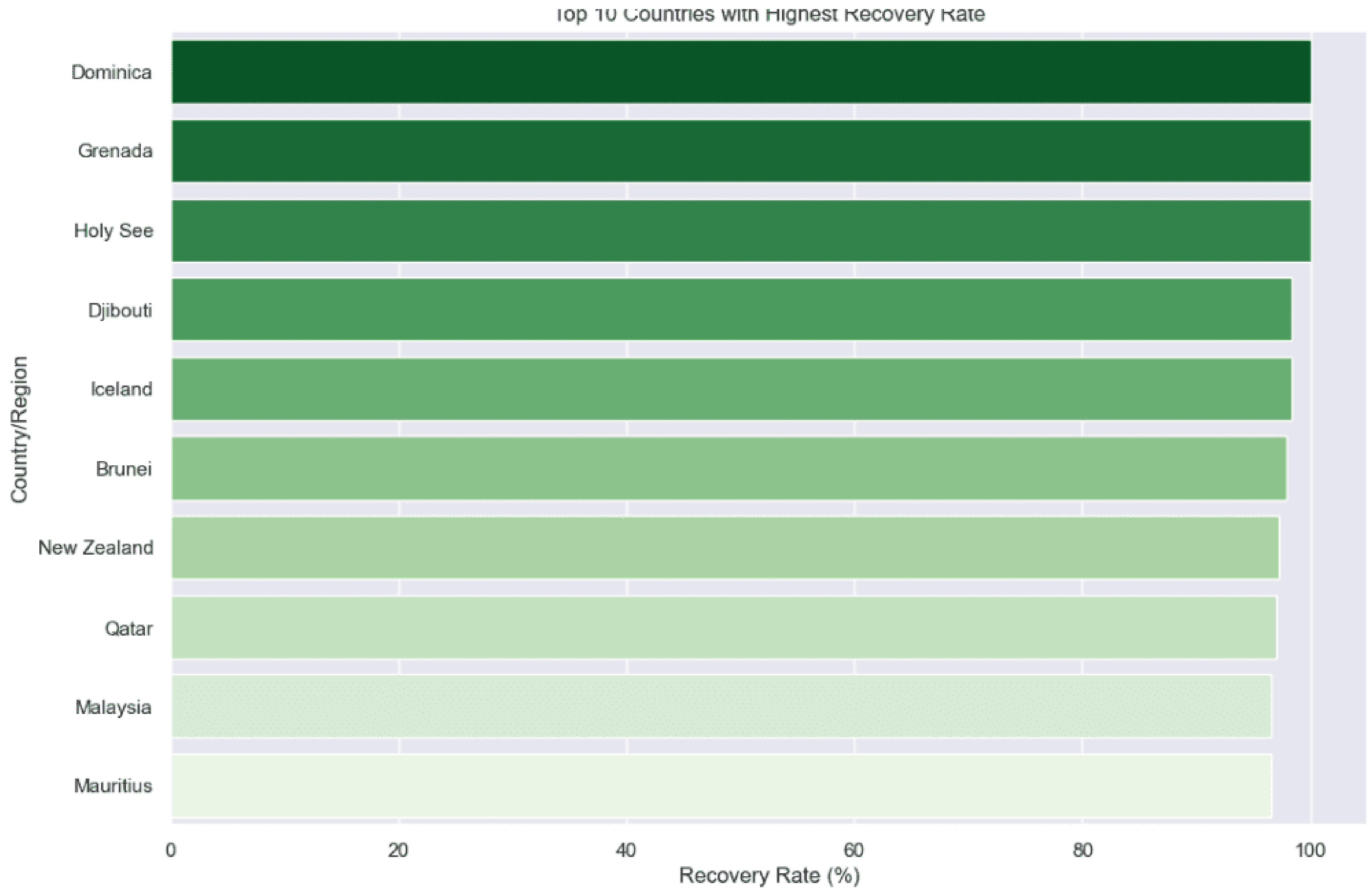
```
df['Death_Rate'] = (df['Deaths'] /
df['Confirmed']) * 100 top_10_death_rate =
df.nlargest(10, 'Death_Rate')[['Country/Region',
'Death_Rate']] print('Top 10 Countries with
Highest Death Rate:') print(top_10_death_rate)
```



Recovery Rate by Country

- Calculation: (Recovered / Confirmed) * 100
- Top 10 Countries with Highest Recovery Rate: Detailed list of countries with the highest recovery rates.

```
df['Recovery_Rate'] = (df['Recovered'] /
df['Confirmed']) * 100 top_10_recovery_rate =
df.nlargest(10, 'Recovery_Rate')
[['Country/Region', 'Recovery_Rate']] print('Top
10 Countries with Highest Recovery Rate:')
print(top_10_recovery_rate)
```



07

Performance Insights

Overall Data Quality and Processing

- Data Integrity: The dataset demonstrated good overall integrity with minimal missing values and appropriate data types for analysis. The data cleaning process, which included renaming columns for consistency and handling missing and infinite values, ensured that the analysis was robust and accurate.
- Processing Efficiency: The use of Python's pandas library allowed for efficient data manipulation and aggregation. The dataset's size was manageable, and all processing tasks, including calculations and visualizations, were performed without significant computational delays.

Statistical Observations

- High Variability:There was high variability in COVID-19 impacts across countries and regions, as seen in the descriptive statistics. Some countries had extremely high counts of confirmed cases, deaths, and recoveries, while others had much lower figures.
- Correlations:Significant correlations were observed between confirmed cases and deaths, and between confirmed cases and active cases. These correlations were critical in understanding the spread and severity of the pandemic in different regions.

Visualization Insights

- Data Distribution:Histograms and scatter plots provided clear visualizations of the distribution and relationships between key metrics. For instance, the scatter plot of confirmed cases vs. deaths highlighted the positive correlation and identified outliers where death rates were particularly high.
- Regional Comparisons:Bar plots and box plots enabled clear comparisons across WHO regions, revealing significant differences in the number of new cases, deaths, and recoveries across regions.

Advanced Metrics Analysis

- High Variability:Death and Recovery Rates:The analysis of death and recovery rates by country provided deeper insights into the effectiveness of different countries' responses to the pandemic. Countries with extremely high death rates, despite having high numbers of recovered cases, were flagged as areas of concern.

08

Key Recommendations

- Target High-Risk Regions:Prioritize interventions in countries with high death rates through improved healthcare resources and stricter containment measures.
- Strengthen Healthcare Systems:Allocate additional resources to regions with high active cases and develop long-term strategies for healthcare system resilience.
- Promote Regional Collaboration:Encourage knowledge sharing and support between countries and WHO regions to manage the pandemic more effectively.
- Continuous Monitoring:Implement real-time data analysis and enhanced reporting to track trends and provide early warnings of potential outbreaks.
- Enhance Public Health Communication:Strengthen public health education campaigns and ensure transparent communication with the public to maintain adherence to guidelines.

THANK YOU

Vishnu G Nath

vishnugnath

vishnugs32@gmail.com