Insights into Global Terrorism

Dataset Description:

Global Terrorism is from Kaggle and consists of a vast amount of data. This database is maintained by organizations such as the National Consortium for the Study of Terrorism and Responses to Terrorism (START) at the University of Maryland. The dataset provides a comprehensive record of terrorist incidents worldwide. Each incident is represented by a set of attributes detailing various aspects of the attack. The dataset contains over 180,000 instances of terrorist attacks, spanning from 1970 to 2017.

Each instance includes a multitude of attributes capturing information about the attack, its perpetrators, victims, and outcomes. It serves as a valuable resource for understanding the patterns, trends, and characteristics of terrorism globally. It's used for analysis, research, and informing policy decisions related to counterterrorism efforts. The data is typically provided in a structured format, such as a CSV (comma-separated values) file or a relational database format. Each row represents a single terrorist incident, and columns represent different attributes of the incident. The dataset is suitable for various types of analyses, including temporal analysis of terrorist trends, spatial analysis of attack hotspots, identification of patterns in attack tactics and motives, and assessment of the effectiveness of counterterrorism strategies.

The Global Terrorism Database contains the following features:

Event ID: A unique identifier for each terrorist event.

Date: The date of the terrorist incident.

Year: The year of the terrorist incident.

Month: The month of the terrorist incident.

Day: The day of the terrorist incident.

Country: The country where the terrorist incident occurred.

Region: The region within the country where the incident occurred.

City: The city or locality where the terrorist incident took place.

Latitude: The latitude coordinates of the incident location.

Longitude: The longitude coordinates of the incident location.

Attack Type: The type of attack, such as bombing/explosion, armed assault, assassination, hijacking, etc.

Target Type: The type of target or victims of the attack (e.g., civilians, military, government, police).

Weapon Type: The type of weapon or tactic used in the attack (e.g., firearms, explosives)

Data Preprocessing

- Categorizing Attack Types Based on Casualty Counts:

  Here, we are creating a new categorical variable named 'attack_type'. This variable serves to classify terrorist attacks into three distinct categories based on the number of casualties. The pd.cut() function is utilized to bin the casualty counts into predefined categories: 'minor', 'small', and 'major'. Attacks resulting in 0 to 2 casualties are classified as 'minor', those with 3 to 10 casualties as 'small', and those with more than 10 casualties as 'major'. This categorization facilitates a clear understanding of the severity and impact of terrorist attacks, enabling subsequent analysis to differentiate between various levels of threat and harm.

- Selective Filtering of Irrelevant Columns:

  Based on the variable created (attack_type) we created, here we selectively filter out irrelevant columns from the dataset. The list relevant_columns specify the columns deemed relevant for the analysis, including the year of the attack ('iyear'), the categorized attack type ('attack_type'), and the number of casualties ('nkill'). By retaining only these essential columns, the preprocessing streamlines the dataset, ensuring that subsequent analyses are targeted and efficient.

- Handling Missing Values:

  To maintain data integrity and reliability, the code addresses missing values within the dataset. The. dropna() method is employed to remove any rows containing incomplete or missing data. By eliminating these instances, the preprocessing ensures that the dataset is clean and complete, free from any inconsistencies or inaccuracies that could potentially skew the analysis results.

- Displaying the Cleaned Subset of the Dataset:

  Finally, the cleaned subset of the dataset is displayed using the .head() method, which provides a concise overview of the preprocessed data. This step allows for a quick visual inspection of the cleaned dataset, enabling analysts to verify that the preprocessing steps have been applied correctly.

Analysis

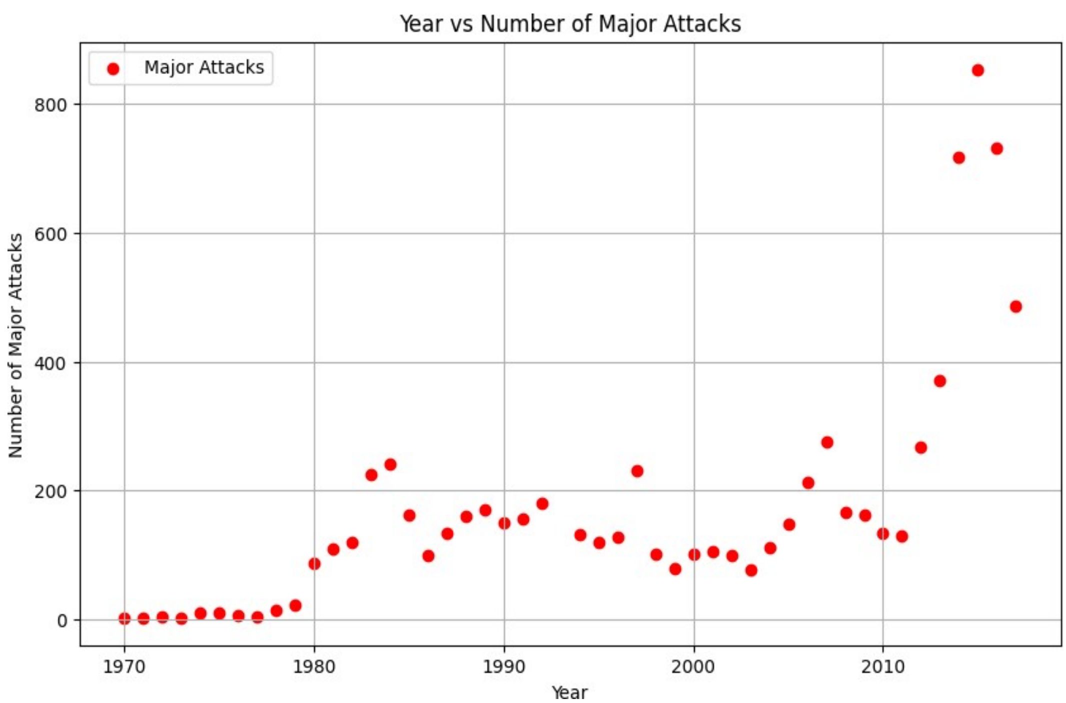
Scatter plot code aims to visualize the trends of terrorist attacks categorized by severity levels: major, small, and minor. It begins by organizing the dataset by year and attack type using pandas' `groupby()` function. This aggregates the data, generating a DataFrame with the count of attacks for each severity level per year. Subsequently, the code creates individual scatter plots for each attack type, depicting the number of attacks over the years. These scatter plots, color-coded for clarity, showcase trends in major (red), small (yellow), and minor (green) attacks separately. Additionally, a combined scatter plot overlays the trends of all attack types for comparison. This visualization approach facilitates the identification of temporal patterns and fluctuations in terrorist activities, aiding analysts in understanding the evolving nature of terrorism over time.

- **Scatter plot of Year vs Number of major attacks**

Observations:

The analysis of major terrorist attacks over time shows some interesting trends. In the 1970s, there weren't many major attacks. But from the 1980s to the early 1990s, there was a big increase, with some years having lots of attacks. Then, from the mid-1990s to the early 2000s, the number of major attacks went down a lot. After 2000, there was another big increase, especially around the mid-2000s and early 2010s. These ups and downs might be linked to big events in the world, changes in laws, or the actions of terrorist groups. The pattern suggests that outside factors have a big impact on how often major attacks happen, showing why it's important to understand and try to prevent terrorism.

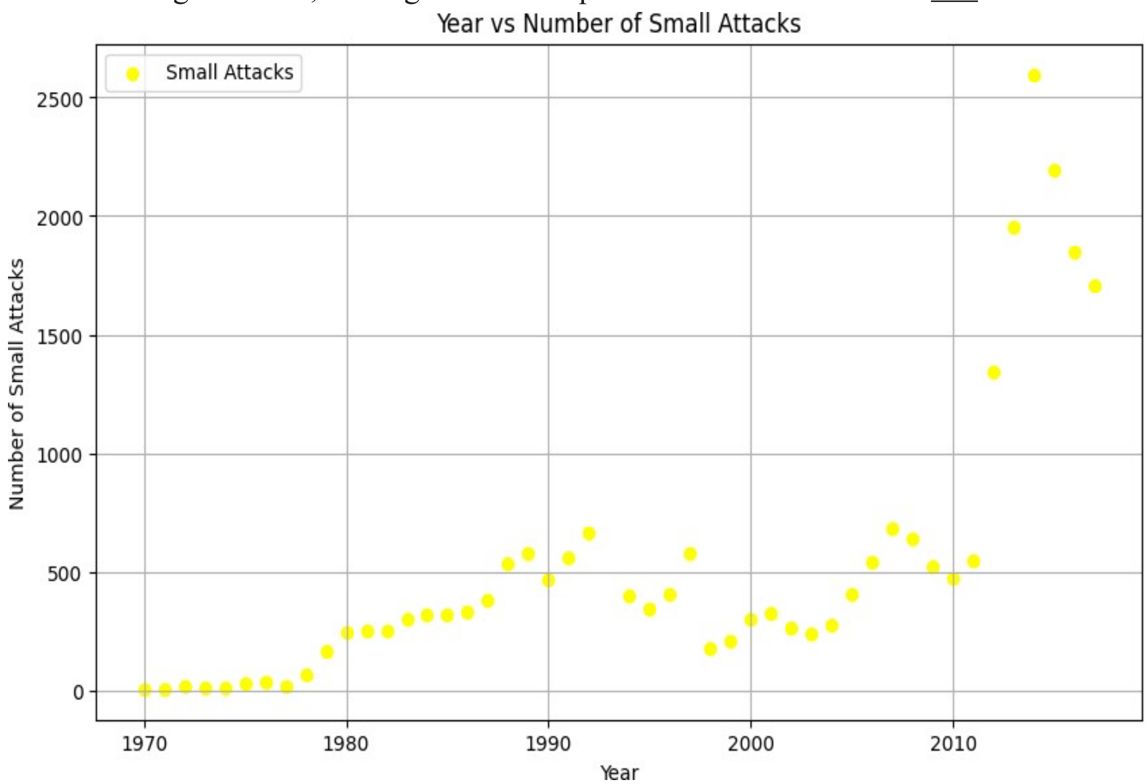
Plot:



- **Scatter plot of Year vs Number of small attacks**

We can see some interesting trends in small-scale terrorist attacks over the years. In the 1970s and 1980s, there's a gradual increase in the number of these attacks, starting from very few in the early 1970s and becoming more frequent by the late 1980s. This trend continues into the

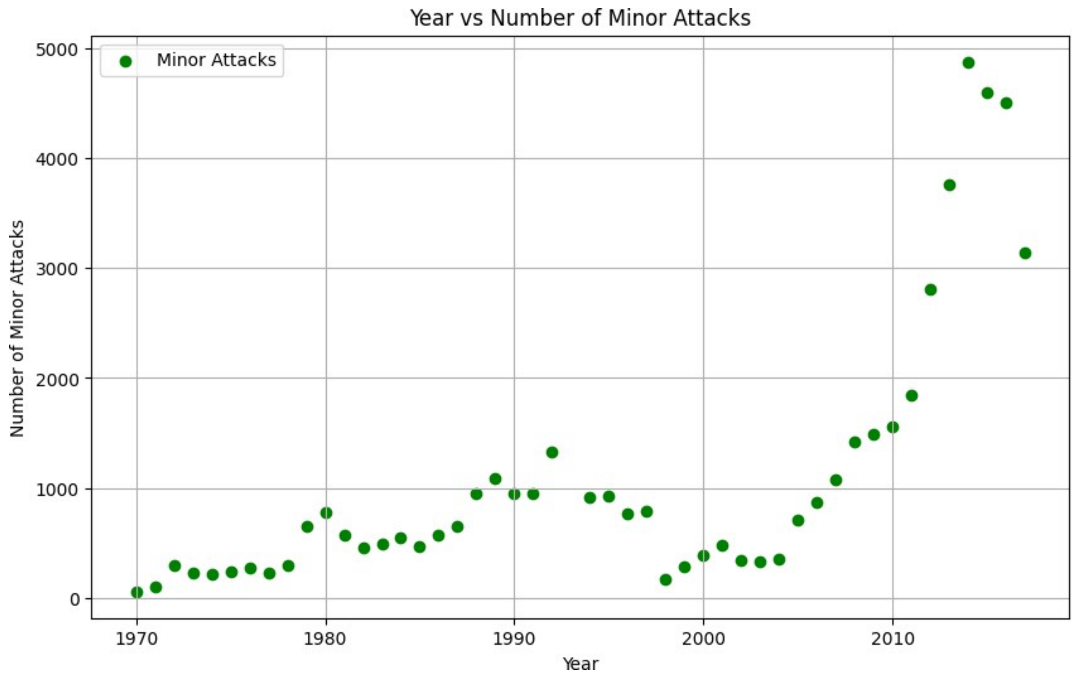
1990s, reaching a peak around the mid-1990s. However, in the 2000s, there's some fluctuation in the number of attacks, with a slight decline towards the late 2000s. Then, in the early 2010s, there's a noticeable increase again, followed by a sharp peak later in the decade. Overall, there's been a general increase in small-scale attacks over the decades, with significant ups and downs in the 2000s and 2010s. The peak in the early 2010s might be linked to global political or social unrest during that time, leading to more frequent attacks of this nature. Plot



- **Scatter plot of Year vs Number of minor attacks**

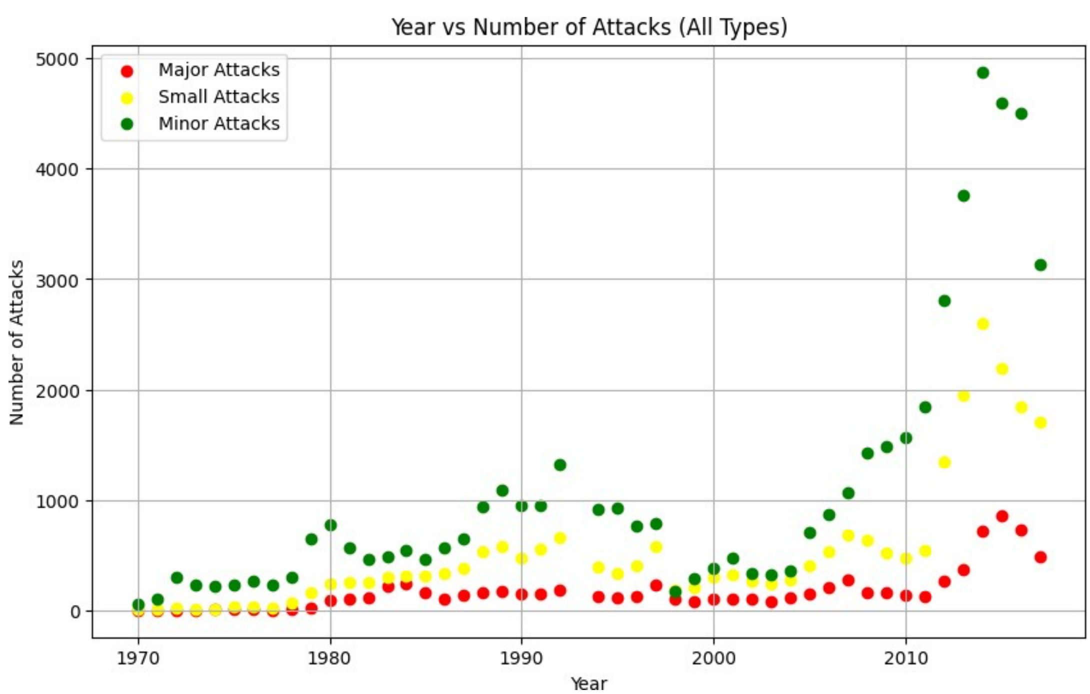
We can see trends in minor terrorist attacks over the years. From the 1970s to the late 1990s, there's a gradual increase in these attacks, though the numbers are still relatively low compared to later years. Around the early 2000s, there's a slight dip, but from the mid-2000s onwards, there's a big increase in minor attacks, with sharp spikes in the late 2000s and early 2010s. This rise might be due to changes in tactics or better reporting of attacks. The sharp spikes could be linked to specific global events or changes in conflicts, which may have led to more minor attacks happening.

Plot



• **Scatter plot of Year vs all attacks**

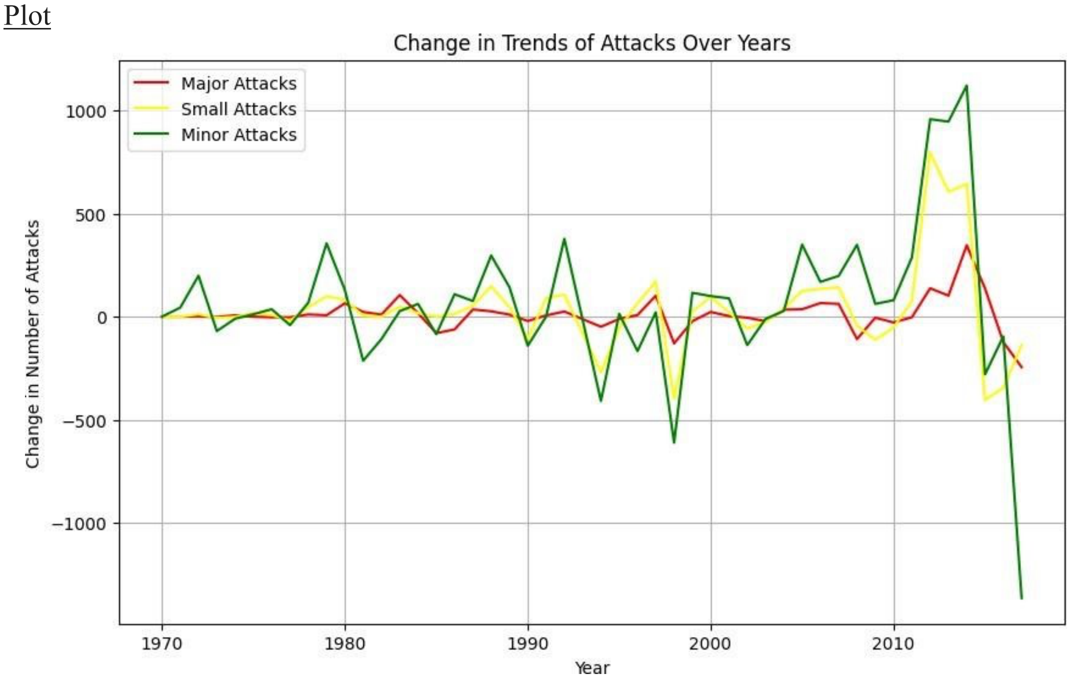
Plot



- **Change in Trends of Attacks over Years**

The graph illustrates the change in terrorist attacks over the years for major, small, and minor attacks, useful for identifying trends and shifts. Major attacks show significant fluctuations with notable spikes in the mid-1980s, early 2000s, and early 2010s, followed by declines. Small attacks exhibit moderate fluctuations, with increases in the early 1990s and mid-2000s, then a sudden rise and drop in the 2010s. Minor attacks show gradual changes until the mid-2000s, followed by a steady increase peaking sharply in the 2010s, then a sudden drop. Notable changes occur in the late 1980s, early 1990s, early 2000s, mid-2000s to early 2010s, and post-2010, indicating shifts in global events, counter-terrorism efforts, or terrorist group strategies. The declines post-2010 suggest major changes in counter-terrorism strategies or terrorist dynamics. These insights are valuable for policymakers, security analysts, and researchers studying global security and terrorism.

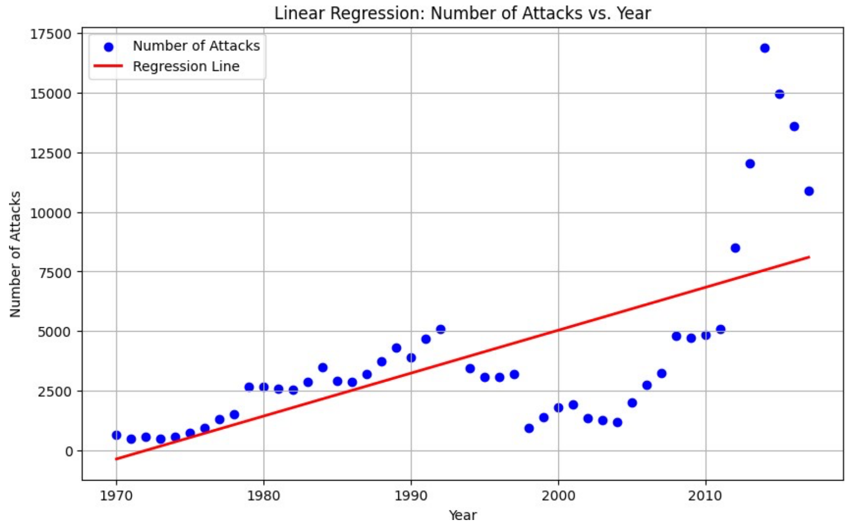
Plot



- **Linear Regression for Number of Total Attacks vs Year**

The chart you provided is a scatter plot with a linear regression line, showing the number of attacks per year from 1970 to around 2010. The x-axis represents the year, and the y-axis represents the number of attacks. Each blue dot represents the number of attacks in a specific year. The data points generally increase over time, particularly showing a steeper rise from the late 1990s onwards. The red line on the graph represents the linear regression line, indicating the overall trend in the data across the years. This line suggests a general increase in the number of attacks over the period displayed. The plot is titled "Linear Regression: Number of Attacks vs. Year," and the y-axis ranges from 0 to around 17,500 attacks.

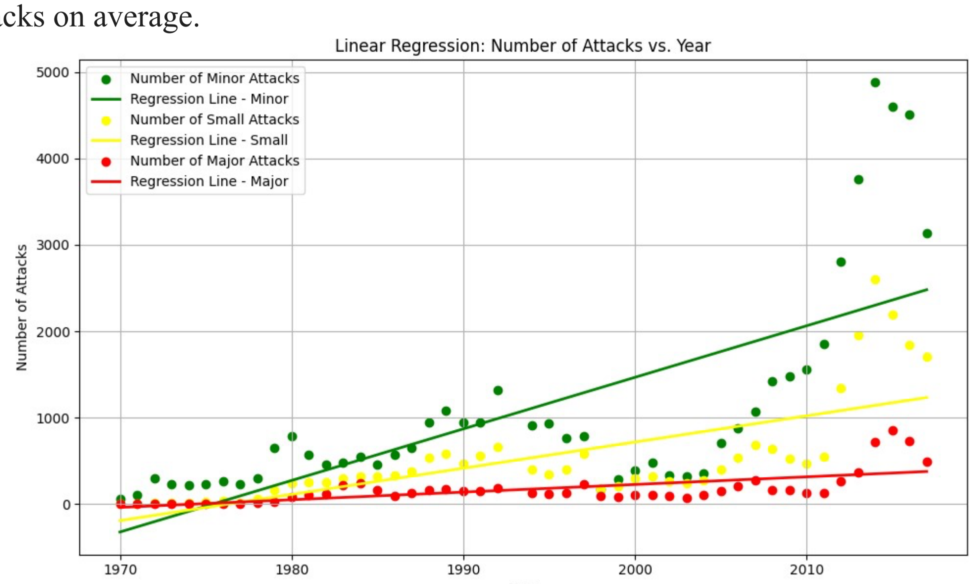
Plot



- **Linear Regression for Number of Attacks (minor, major, small) vs Year**

The number of minor attacks shows a steady increase over the period. Small Attacks eases over time, following a similar trend to the minor attacks but with fewer overall incidents. The number of major attacks remains relatively stable and low compared to the other categories, with the regression line showing a slight upward trend but much less pronounced than the other two categories.

For minor attacks, the coefficient is 59.63 suggests that for each year that passes, we expect to see an increase of approximately 59.63 minor attacks on average. Similarly, for small attacks, the coefficient of 30.28 indicates that for every year that goes by, we anticipate a rise of about 30.28 small attacks on average. For major attacks, we predict an increase of around 8.83 major attacks on average.



Code:

Data Preprocessing

import pandas as pd import

matplotlib.pyplot as plt #

Loading the dataset

df = pd.read_csv('globalterrorismdb_0718dist.csv', encoding='ISO-8859-1', low_memory=False)

print(df.head()) print(df.shape) # Data Preprocessing

df['attack_type'] = pd.cut(df['nkill'], bins=[0, 2, 10, float('inf')], labels=['minor', 'small', 'major'])

relevant_columns = ['iyear', 'attack_type', 'nkill'] df_subset = df[relevant_columns]

Removing rows with missing values by creating a copy of the subset

DataFrame df_subset = df_subset.dropna().copy() # Displaying the cleaned

subset of the dataset print(df_subset.head())

```
import pandas as pd
import matplotlib.pyplot as plt

# Loading the dataset
df = pd.read_csv('globalterrorismdb_0718dist.csv', encoding='ISO-8859-1', low_memory=False)

print(df.head())
print(df.shape)

# Data Preprocessing
df['attack_type'] = pd.cut(df['nkill'], bins=[0, 2, 10, float('inf')], labels=['minor', 'small', 'major'])
relevant_columns = ['iyear', 'attack_type', 'nkill']
df_subset = df[relevant_columns]

# Removing rows with missing values by creating a copy of the subset DataFrame
df_subset = df_subset.dropna().copy()

# Displaying the cleaned subset of the dataset
print(df_subset.head())
```

```
       eventid  iyear  imonth  iday approxdate  extended resolution  country  \
```

Scatter plot of Year vs Number of attack attacks_by_year_type =

df.groupby(['iyear', 'attack_type']).size().unstack(fill_value=0)

Scatter plot for major attacks plt.figure(figsize=(10,

6))

plt.scatter(attacks_by_year_type.index, attacks_by_year_type['major'], color='red', label='Major Attacks')
plt.xlabel('Year')

plt.ylabel('Number of Major Attacks')

plt.title('Year vs Number of Major Attacks')

plt.legend() plt.grid(True) plt.show()

```
# Scatter plot for small attacks
plt.figure(figsize=(10, 6))
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['small'], color='yellow', label='Small Attacks')
plt.xlabel('Year')
plt.ylabel('Number of Small Attacks')
plt.title('Year vs Number of Small Attacks')
plt.legend() plt.grid(True) plt.show()
# Scatter plot for minor attacks plt.figure(figsize=(10,
6))
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['minor'], color='green', label='Minor Attacks')
plt.xlabel('Year')
plt.ylabel('Number of Minor Attacks')
plt.title('Year vs Number of Minor Attacks')
plt.legend() plt.grid(True) plt.show()
# Scatter plot for all attacks together plt.figure(figsize=(10,
6))
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['major'], color='red', label='Major Attacks')
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['small'], color='yellow', label='Small Attacks')
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['minor'], color='green', label='Minor Attacks')
plt.xlabel('Year')
plt.ylabel('Number of Attacks')
plt.title('Year vs Number of Attacks (All Types)')
plt.legend() plt.grid(True) plt.show()
```

```
attacks_by_year_type = df.groupby(['iyear', 'attack_type']).size().unstack(fill_value=0)

# Scatter plot for major attacks
plt.figure(figsize=(10, 6))
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['major'], color='red', label='Major Attacks')
plt.xlabel('Year')
plt.ylabel('Number of Major Attacks')
plt.title('Year vs Number of Major Attacks')
plt.legend()
plt.grid(True)
plt.show()

# Scatter plot for small attacks
plt.figure(figsize=(10, 6))
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['small'], color='yellow', label='Small Attacks')
plt.xlabel('Year')
plt.ylabel('Number of Small Attacks')
plt.title('Year vs Number of Small Attacks')
plt.legend()
plt.grid(True)
plt.show()

# Scatter plot for minor attacks
plt.figure(figsize=(10, 6))
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['minor'], color='green', label='Minor Attacks')
plt.xlabel('Year')
plt.ylabel('Number of Minor Attacks')
plt.title('Year vs Number of Minor Attacks')
plt.legend()
plt.grid(True)
plt.show()

# Scatter plot for all attacks together
plt.figure(figsize=(10, 6))
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['major'], color='red', label='Major Attacks')
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['small'], color='yellow', label='Small Attacks')
plt.scatter(attacks_by_year_type.index, attacks_by_year_type['minor'], color='green', label='Minor Attacks')
plt.xlabel('Year')
plt.ylabel('Number of Attacks')
plt.title('Year vs Number of Attacks (All Types)')
plt.legend()
plt.grid(True)
plt.show()
```

Changes in the trend of attacks Vs Year diff_major = attacks_by_year_type['major'].diff().fillna(0) diff_small = attacks_by_year_type['small'].diff().fillna(0) diff_minor = attacks_by_year_type['minor'].diff().fillna(0)

Plotting trends for each attack type plt.figure(figsize=(10, 6)) plt.plot(attacks_by_year_type.index, diff_major, label='Major Attacks', color='red') plt.plot(attacks_by_year_type.index, diff_small, label='Small Attacks', color='yellow') plt.plot(attacks_by_year_type.index, diff_minor, label='Minor Attacks', color='green') plt.xlabel('Year')

plt.ylabel('Change in Number of Attacks')

plt.title('Change in Trends of Attacks Over Years')

plt.legend() plt.grid(True) plt.show()

c

```
]: diff_major = attacks_by_year_type['major'].diff().fillna(0)
diff_small = attacks_by_year_type['small'].diff().fillna(0)
diff_minor = attacks_by_year_type['minor'].diff().fillna(0)

# Plotting trends for each attack type
plt.figure(figsize=(10, 6))

plt.plot(attacks_by_year_type.index, diff_major, label='Major Attacks', color='red')
plt.plot(attacks_by_year_type.index, diff_small, label='Small Attacks', color='yellow')
plt.plot(attacks_by_year_type.index, diff_minor, label='Minor Attacks', color='green')

plt.xlabel('Year')
plt.ylabel('Change in Number of Attacks')
plt.title('Change in Trends of Attacks Over Years')
plt.legend()
plt.grid(True)
plt.show()
```

Linear regression on attacks(major,minor,small) Vs year

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression

attacks_by_year_type = df.groupby(['iyear', 'attack_type']).size().unstack(fill_value=0)

Performing linear regression for each attack type fig, ax =

plt.subplots(figsize=(10, 6)) colors = {'minor': 'green',

'small': 'yellow', 'major': 'red'} for attack_type, color in

colors.items(): X = attacks_by_year_type.index.to_frame() y

= attacks_by_year_type[attack_type].values.reshape(-1, 1)

regression = LinearRegression().fit(X, y)

Plotting scatter plot and regression line for each attack type ax.scatter(X, y,

color=color, label=f'Number of {attack_type.capitalize()} Attacks') ax.plot(X,

regression.predict(X), color=color, linewidth=2, label=f'Regression Line -

{attack_type.capitalize()}')

print(f"Regression Coefficient ({attack_type.capitalize()} Attacks): {regression.coef_[0][0]}")

ax.set_xlabel('Year')

ax.set_ylabel('Number of Attacks') ax.set_title('Linear

Regression: Number of Attacks vs. Year') ax.legend()

ax.grid(True) plt.tight_layout() plt.show()

```python
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

attacks_by_year_type = df.groupby(['iyear', 'attack_type']).size().unstack(fill_value=0)
# Performing linear regression for each attack type
fig, ax = plt.subplots(figsize=(10, 6))
colors = {'minor': 'green', 'small': 'yellow', 'major': 'red'}
for attack_type, color in colors.items():
    X = attacks_by_year_type.index.to_frame()
    y = attacks_by_year_type[attack_type].values.reshape(-1, 1)
    regression = LinearRegression().fit(X, y)
   # Plotting scatter plot and regression line for each attack type
    ax.scatter(X, y, color=color, label=f'Number of {attack_type.capitalize()} Attacks')
    ax.plot(X, regression.predict(X), color=color, linewidth=2, label=f'Regression Line - {attack_type.capitalize()}')
    print(f"Regression Coefficient ({attack_type.capitalize()} Attacks): {regression.coef_[0][0]}")
ax.set_xlabel('Year')
ax.set_ylabel('Number of Attacks')
ax.set_title('Linear Regression: Number of Attacks vs. Year')
ax.legend()
ax.grid(True)
plt.tight_layout()
plt.show()
```

```
Regression Coefficient (Minor Attacks): 59.62593543857056
Regression Coefficient (Small Attacks): 30.284999260888053
Regression Coefficient (Major Attacks): 8.825853674310316
```

Linear Regression: Number of Attacks vs. Year

Linear regression on total number of attacks Vs year import

matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression import

pandas as pd

attacks_by_year = df.groupby('iyear').size()

X = attacks_by_year.index.to_frame() y =

attacks_by_year.values.reshape(-1, 1)

# Perform linear regression
regression = LinearRegression().fit(X, y) #

Plotting scatter plot and regression line

plt.figure(figsize=(10, 6))

plt.scatter(X, y, color='blue', label='Number of Attacks')

plt.plot(X, regression.predict(X), color='red', linewidth=2, label='Regression Line') plt.xlabel('Year')

plt.ylabel('Number of Attacks')

plt.title('Linear Regression: Number of Attacks vs. Year')

plt.legend() plt.grid(True) plt.show()

print("Regression Coefficients:")

print(f"Coefficient: {regression.coef_[0][0]}")

```python
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
import pandas as pd

attacks_by_year = df.groupby('iyear').size()
X = attacks_by_year.index.to_frame()
y = attacks_by_year.values.reshape(-1, 1)
# Perform linear regression
regression = LinearRegression().fit(X, y)
# Plotting scatter plot and regression line
plt.figure(figsize=(10, 6))
plt.scatter(X, y, color='blue', label='Number of Attacks')
plt.plot(X, regression.predict(X), color='red', linewidth=2, label='Regression Line')
plt.xlabel('Year')
plt.ylabel('Number of Attacks')
plt.title('Linear Regression: Number of Attacks vs. Year')
plt.legend()
plt.grid(True)
plt.show()

print("Regression Coefficients:")
print(f"Coefficient: {regression.coef_[0][0]}")
```

```
Regression Coefficients:
Coefficient: 180.04630767382992
```