Global Terrorism Database (GTD) - Exploratory Data Analysis

This notebook explores the Global Terrorism Database, which records worldwide terrorist incidents from 1970 to 2017.
The goal is to clean and analyze the dataset to uncover key trends in time, geography, attack types, and impact.

1. Introduction

Terrorism has shaped the global landscape for decades, affecting societies, economies, and international relations. To better understand these events, researchers rely on the Global Terrorism Database (GTD), an open-source project maintained by the University of Maryland. Covering incidents from 1970 to 2017, the GTD is the most comprehensive dataset of its kind, with more than 180,000 recorded attacks worldwide.

This notebook aims to uncover stories hidden within the data. How has terrorism changed over time? Which regions have been most affected? What tactics and targets are most common, and what has been the human cost? By answering these questions, we can gain a clearer picture of terrorism's global impact and how it has evolved across decades.

The analysis will follow four steps:

1. Data cleaning and preparation.
2. Exploratory analysis of temporal and geographical patterns.
3. Investigation of attack types, targets, and perpetrators.
4. Assessment of impact in terms of casualties and injuries.

By the end, we aim to not just present statistics, but also highlight meaningful insights that help us understand global terrorism trends—and recognize the limitations of the dataset, which ends in 2017.

2.Load and intial inspection

Before diving into analysis, we begin by importing the necessary libraries, loading the dataset, and performing a quick inspection.
This helps us understand the dataset's size, structure, and quality, and identify potential issues such as missing values or duplicates.

Load Dataset

We load the Global Terrorism Database (GTD) CSV file.
Since the dataset is large and has mixed data types, we use `low_memory=False` to avoid warnings.

```
The dataset contains 181,691 rows and 135 columns.
```

First Look

We inspect the first few rows, along with column data types and non-null counts,
to understand the dataset's overall structure.

```
Dataset shape (rows, columns): (181691, 135)
```

	eventid	iyear	imonth	iday	approxdate	extended	resolution	country	country_txt	r
0	197000000001	1970	7	2	NaN	0	NaN	58	Dominican Republic	
1	197000000002	1970	0	0	NaN	0	NaN	130	Mexico	
2	197001000001	1970	1	0	NaN	0	NaN	160	Philippines	
3	197001000002	1970	1	0	NaN	0	NaN	78	Greece	
4	197001000003	1970	1	0	NaN	0	NaN	101	Japan	

5 rows × 135 columns

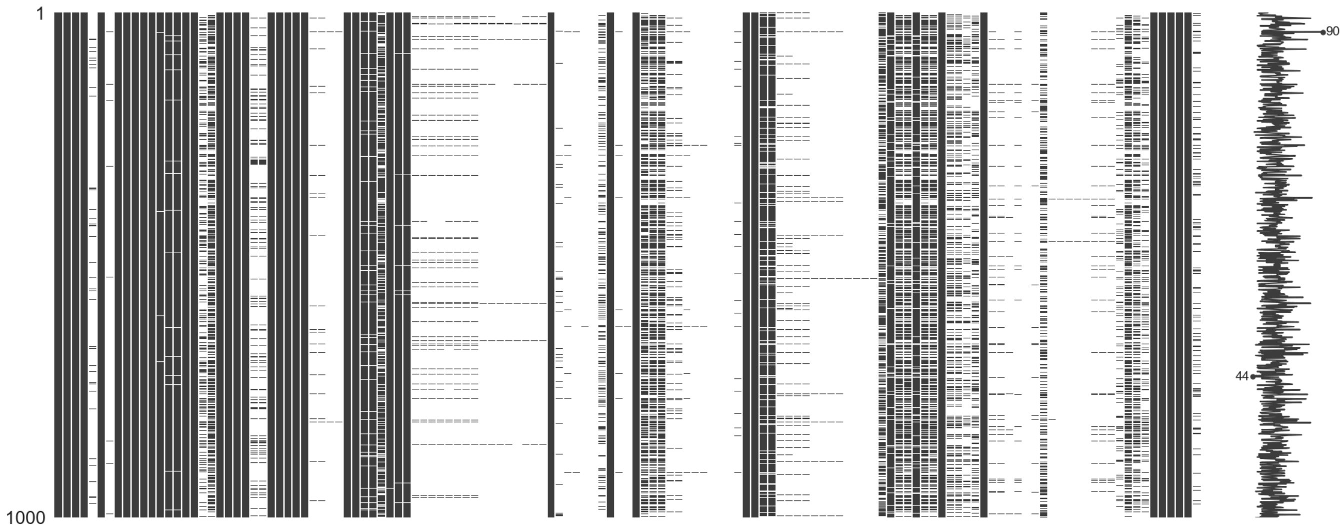
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 181691 entries, 0 to 181690
Columns: 135 entries, eventid to related
dtypes: float64(55), int64(22), object(58)
memory usage: 187.1+ MB
```

Missing Values and Duplicates

Next, we check for missing values and duplicate records.

```
Top 15 columns with most missing values:
gsubname3           181671
weapsubtype4_txt    181621
weapsubtype4        181621
weaptype4           181618
weaptype4_txt       181618
claimmode3          181558
claimmode3_txt      181558
gsubname2           181531
claim3              181373
guncertain3         181371
gname3              181367
divert              181367
attacktype3         181263
attacktype3_txt     181263
ransomnote          181179
dtype: int64

Number of duplicate rows: 0
```



Quick Summary

- The dataset has ~181,000 rows and 135 columns.
- Several columns (e.g., `gsubname3`, `weapsubtype4`) are almost entirely missing.
- No duplicate rows were found.

For clarity in later analysis, we will rename key columns such as `iyear`, `imonth`, `iday`, `country_txt`, and `region_txt`.

3.Data cleaning

```
Shape after column reduction: (181691, 20)
```

```
nkill      non-integer count: 0
nwound     non-integer count: 2
suicide    non-integer count: 0
success    non-integer count: 0
multiple   non-integer count: 0
```

```
== QA report ==
Rows, Cols: (181691, 28)
Exact dates: 180800
No day only: 871
Year only : 20
Killed known % : 94.32
Wounded known %: 91.02
Unique countries: 205
Unique attack types: 9

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 181691 entries, 0 to 181690
Data columns (total 28 columns):
 #   Column           Non-Null Count   Dtype
---  ------           --------------   -----
 0   iyear            181691 non-null  int64
 1   imonth           181691 non-null  int64
 2   iday             181691 non-null  int64
 3   country_txt      181691 non-null  category
 4   region_txt       181691 non-null  category
 5   provstate        181691 non-null  category
 6   city             181691 non-null  category
 7   attacktype1_txt  181691 non-null  category
 8   targtype1_txt    181691 non-null  category
 9   gname            181691 non-null  category
 10  nkill            171378 non-null  Int64
 11  nwound           165380 non-null  Int64
 12  property         181691 non-null  int64
 13  propextent_txt   181691 non-null  category
 14  weaptype1_txt    181691 non-null  category
 15  suicide          181691 non-null  Int64
 16  success          181691 non-null  Int64
 17  multiple         181690 non-null  Int64
 18  latitude         177135 non-null  float64
 19  longitude        177134 non-null  float64
 20  month_unknown    181691 non-null  bool
 21  day_unknown      181691 non-null  bool
 22  date             181691 non-null  datetime64[ns]
 23  date_precision   181691 non-null  category
 24  killed_known     181691 non-null  bool
 25  wounded_known    181691 non-null  bool
 26  nkill_filled     181691 non-null  Int64
 27  nwound_filled    181691 non-null  Int64
dtypes: Int64(7), bool(4), category(10), datetime64[ns](1), float64(2), int64(4)
memory usage: 25.6 MB
```

	iyear	imonth	iday	country_txt	region_txt	provstate	city	attacktype1_txt	targtype1_t
0	1970	7	2	Dominican Republic	Central America & Caribbean	Unknown	Santo Domingo	Assassination	Priva Citizens Proper
1	1970	0	0	Mexico	North America	Federal	Mexico City	Hostage Taking (Kidnapping)	Governme (Diplomati
2	1970	1	0	Philippines	Southeast Asia	Tarlac	Unknown	Assassination	Journalists Med

3 rows × 28 columns

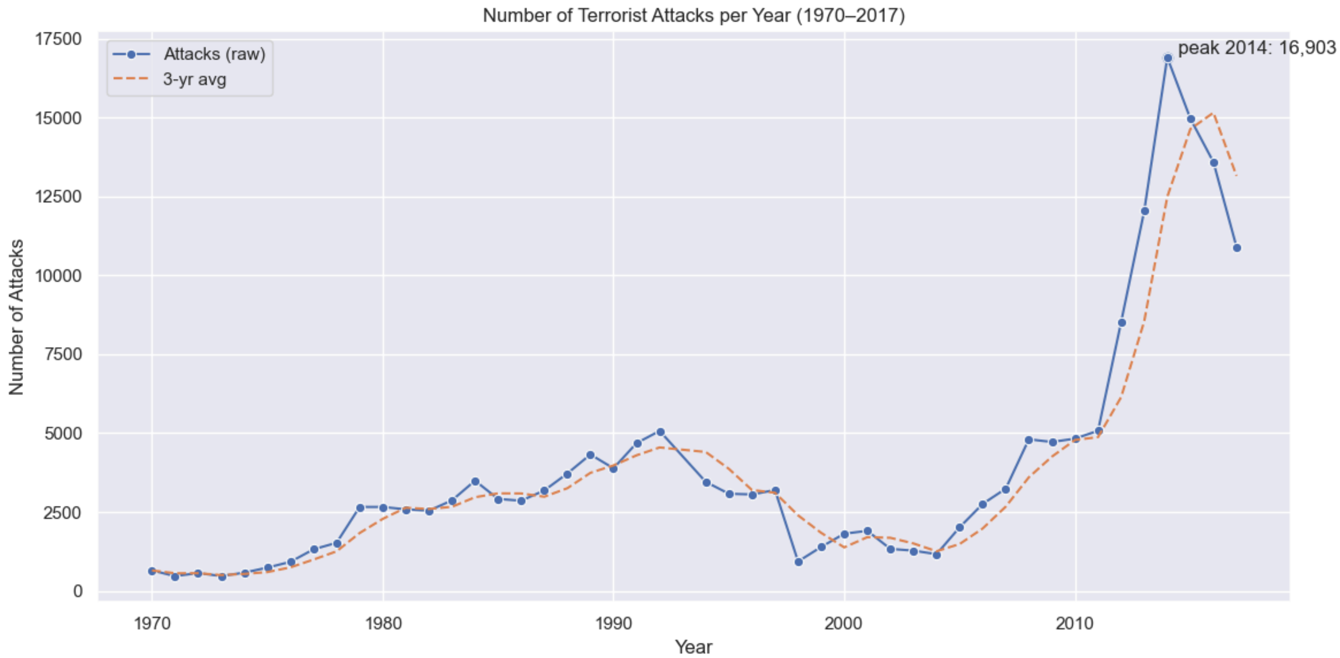
4. EDA

4.1 Over time

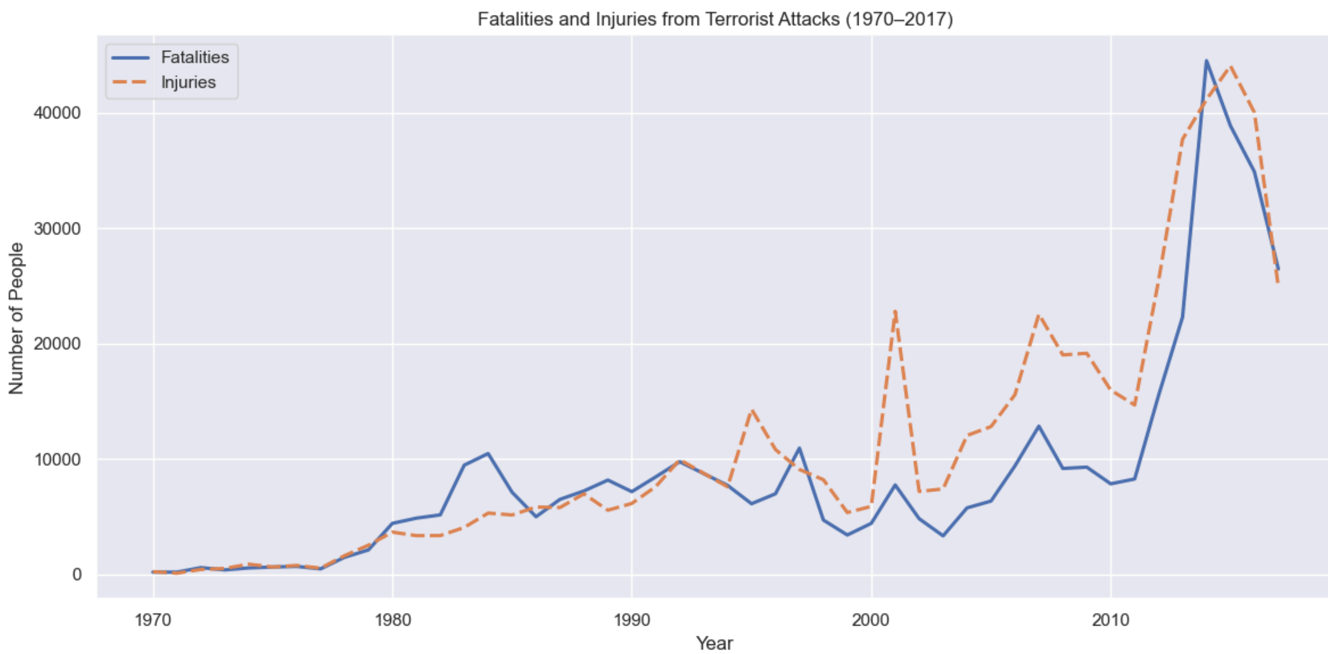
Why this matters. Time trends show how terrorism evolved, where peaks occurred, and whether the average severity of incidents is changing. I analyse:

1. total attacks per year,
2. total casualties (fatalities & injuries) per year, and
3. average casualties per attack (severity) per year.

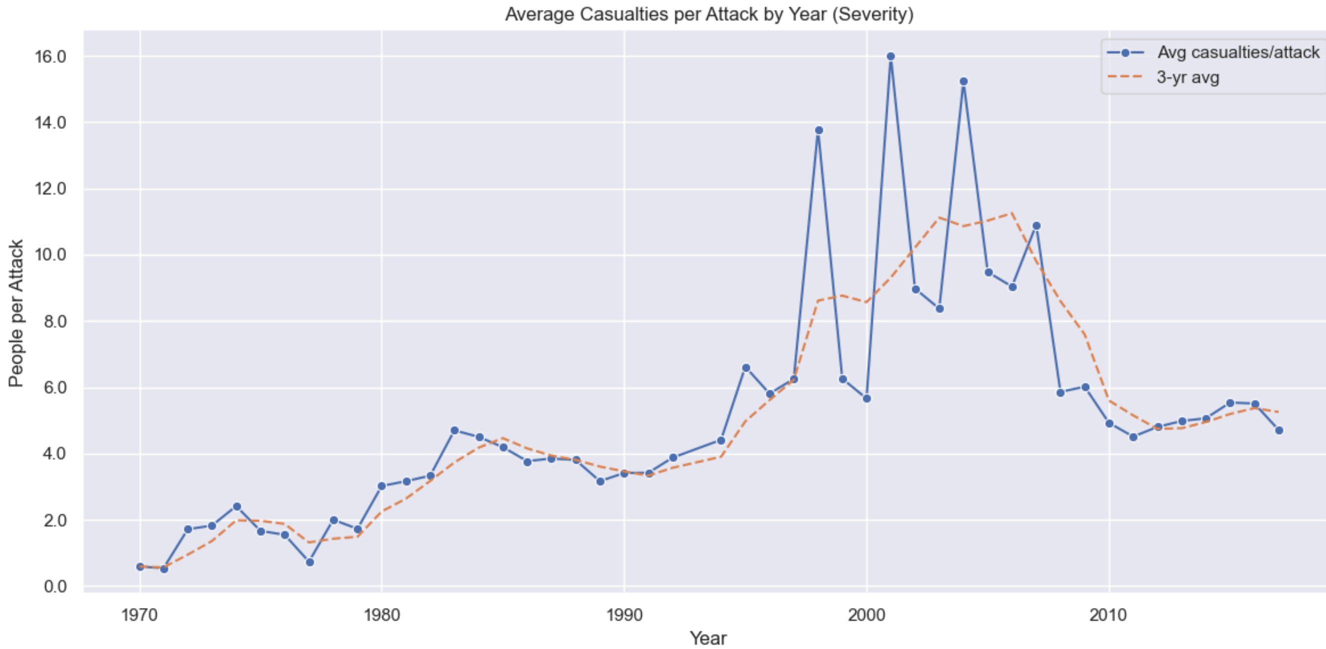
Notes on data quality. Some incidents have unknown month/day; yearly counts use `iyear` to avoid date-precision bias. Casualties use zeros for missing values only for aggregation (the original `nkill` / `nwound` remain untouched for integrity).



Number of Terrorist Attacks per Year (1970–2017)
Attacks (raw)
3-yr avg
peak 2014: 16,903
Number of Attacks
Year



Fatalities and Injuries from Terrorist Attacks (1970–2017)
Fatalities
Injuries
Number of People
Year




```
Attacks peak: 2014 (16,903 attacks)
Worst casualty years (by totals): 2014, 2015, 2016
Highest severity years: 2001, 2004, 1998
```

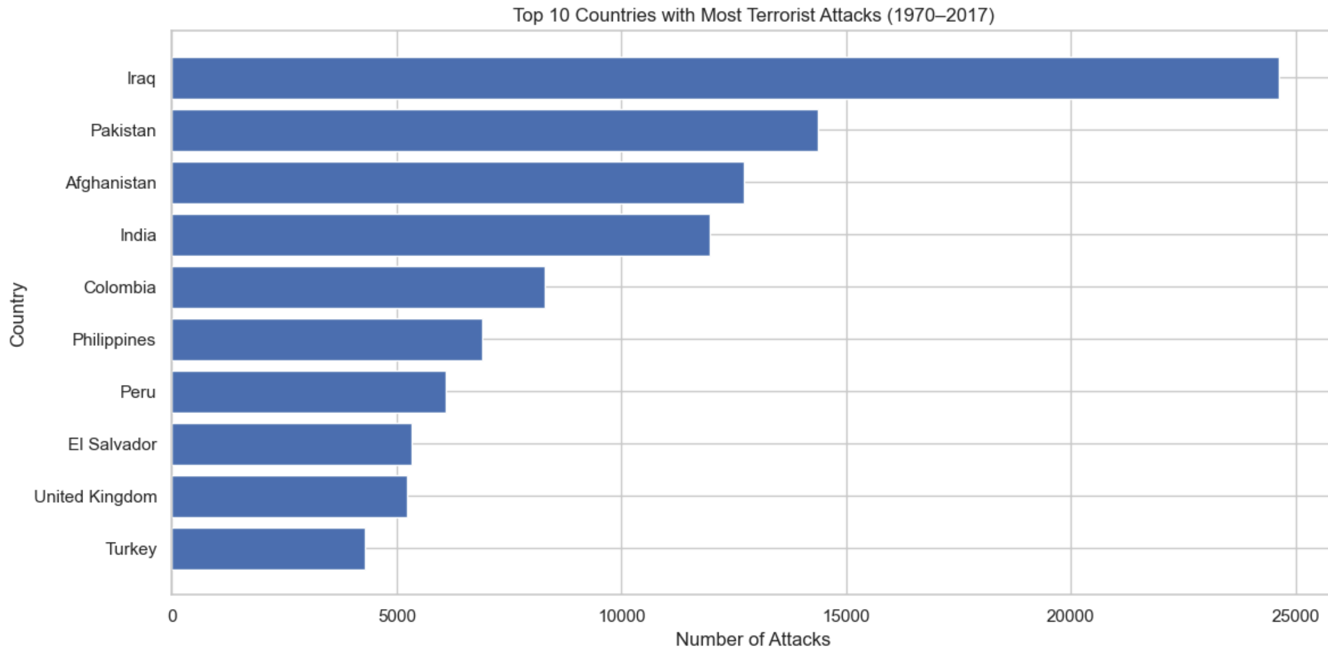
Observations (Over Time):

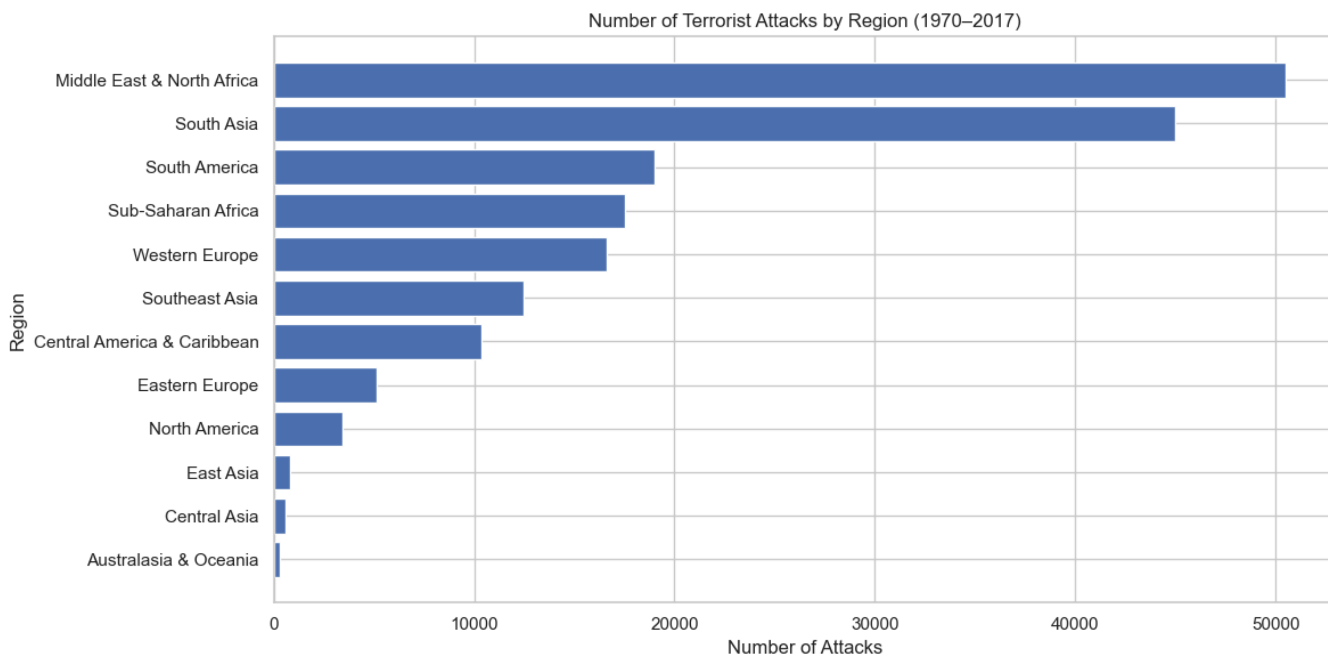
- Terrorist incidents increased steadily from the 1970s, with **notable peaks in the early 1990s** and a sharp surge after **2010**, reaching a record high in **2014 with 16,903 attacks**.
- Fatalities and injuries follow a similar upward trend, with the **worst casualty years** being **2014–2016** (over 40,000 victims annually).
- While the number of incidents declines after 2014, the **average casualties per attack (severity)** peaked earlier in **1998, 2001, and 2004**, reflecting rare but catastrophic events (e.g., 9/11).
- Overall, terrorism became **more frequent after 2010**, but **most deadly on a per-attack basis around the late 1990s–early 2000s**.

4.2 By geography

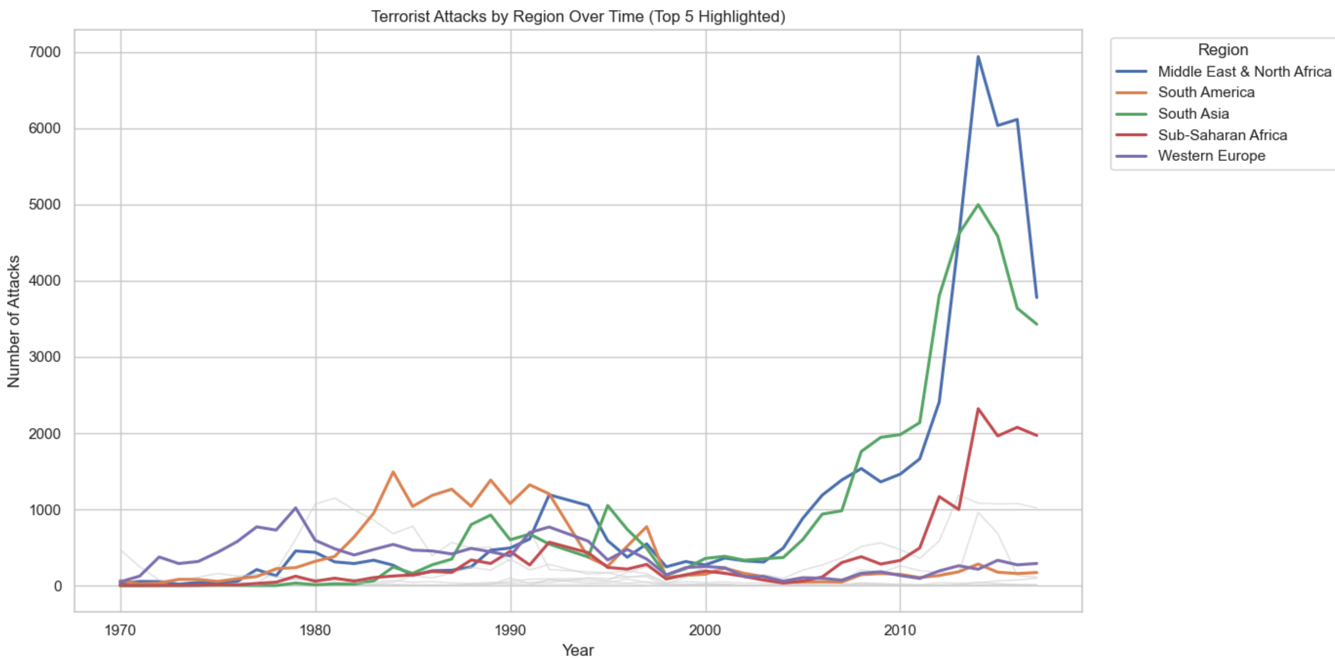
Why this matters. Geography reveals where terrorist activity concentrates and how regional dynamics shift over time.
I present: (1) top countries by total incidents with percentages, (2) attacks by region, (3) regional trends over time (top 5 regions to reduce clutter), and (4) an interactive map.

Notes. Counts use `country_txt` and `region_txt`. For time, I use `iyear` (robust to unknown month/day). The map uses clustering/heatmap so we don't overload the browser with millions of markers.






```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/3557185637.py:51: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  .groupby(['iyear','region_txt'])
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/3557185637.py:58: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  year_region.groupby('region_txt')['attacks'].sum()
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/3557185637.py:63: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  for reg, sub in year_region.groupby('region_txt'):
```


Terrorist Attacks by Region Over Time (Top 5 Highlighted)
Number of Attacks
Year
Region
Middle East & North Africa
South America
South Asia
Sub-Saharan Africa
Western Europe
0
1000
2000
3000
4000
5000
6000
7000
1970
1980
1990
2000
2010

Observations (Geography):

- Terrorist activity is highly concentrated in **Middle East & North Africa** and **South Asia**, which hold the largest regional totals.
- By country, **Iraq** leads by a wide margin; **Pakistan**, **Afghanistan**, and **India** are also consistently high, followed by **Colombia**, **Peru**, and the **Philippines**.
- Regional trend lines show a **sharp post-2010 surge** in **MENA and South Asia**; **Sub-Saharan Africa** rises through the 2010s, while **Western Europe** and **South Americ**a show **earlier peaks and long-run declines**.
- The maps reveal **dense hotspots** along conflict corridors: **Iraq–Syria**, **Afghanistan–Pakistan**, **Kashmir/northern India**, **Nigeria & the Horn of Africa**, **Colombia–Peru**, and **Mindanao (Philippines)**.

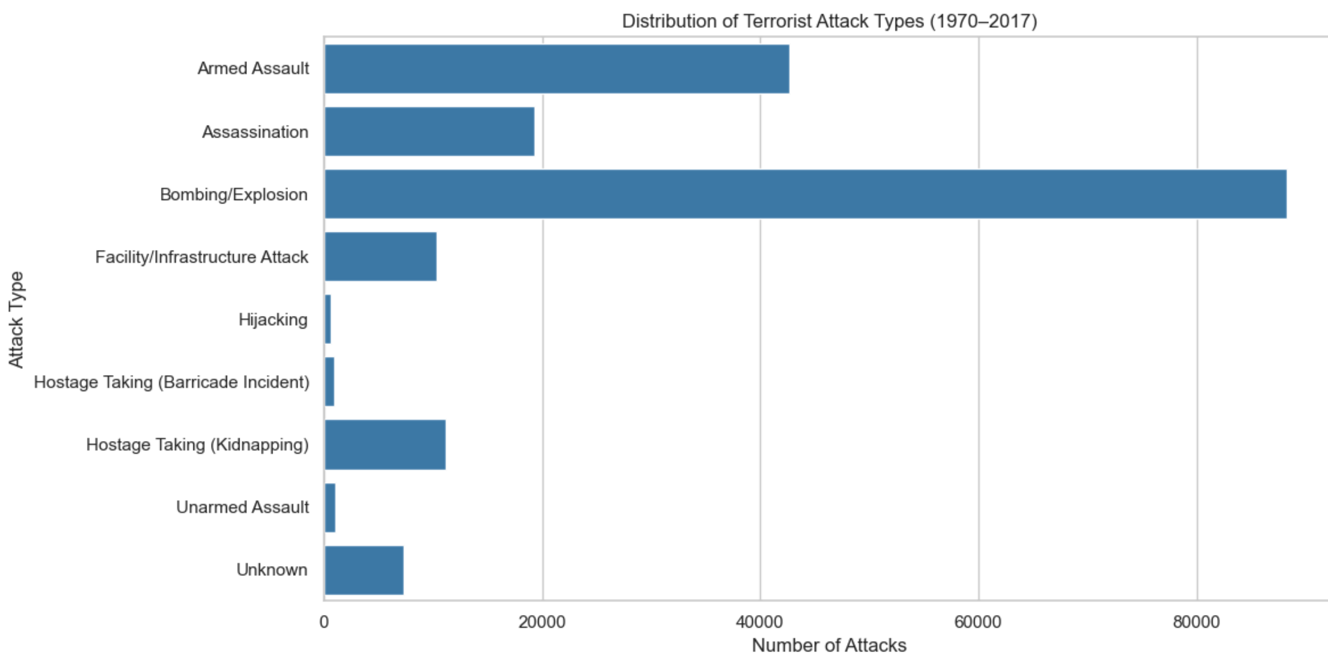
Notes: These are volume counts (not per-capita risk); some locations are approximate and reporting intensity varies by place and time.

4.3 By attack type

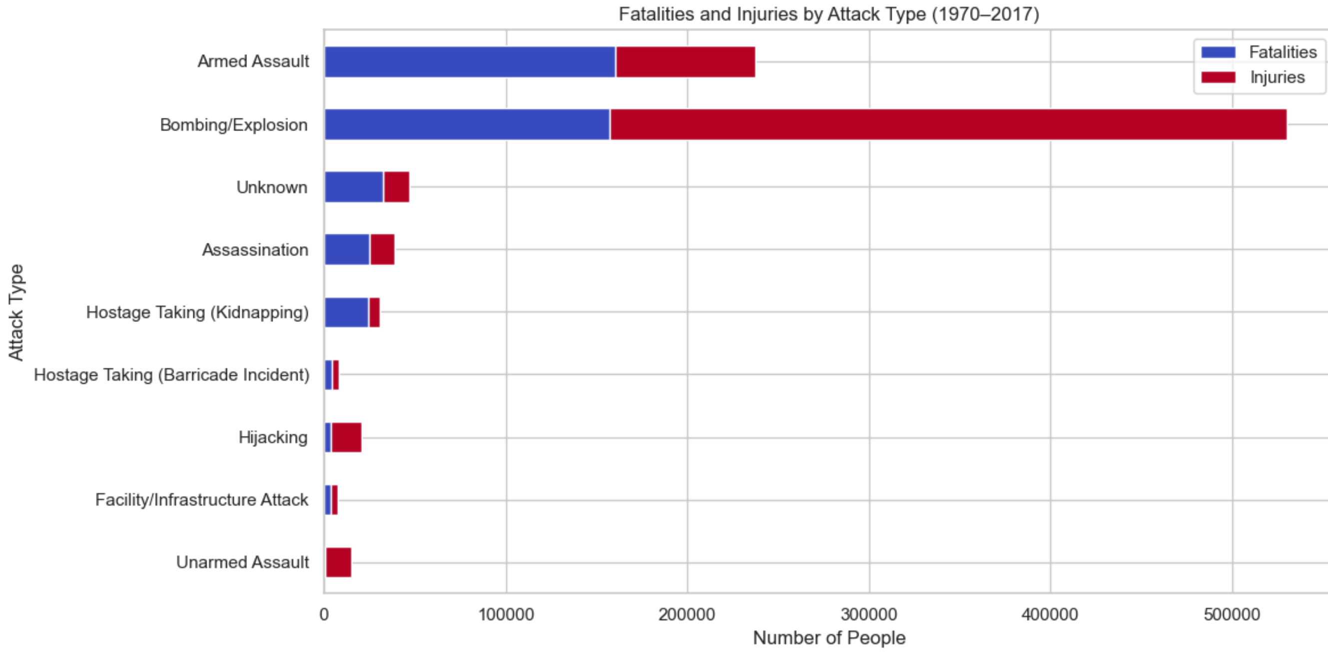
Why this matters. Attack type shows *how* terrorism is carried out. Different tactics have different frequencies and impacts — e.g., bombings are most common, but armed assaults often cause more fatalities per incident.

I analyse:

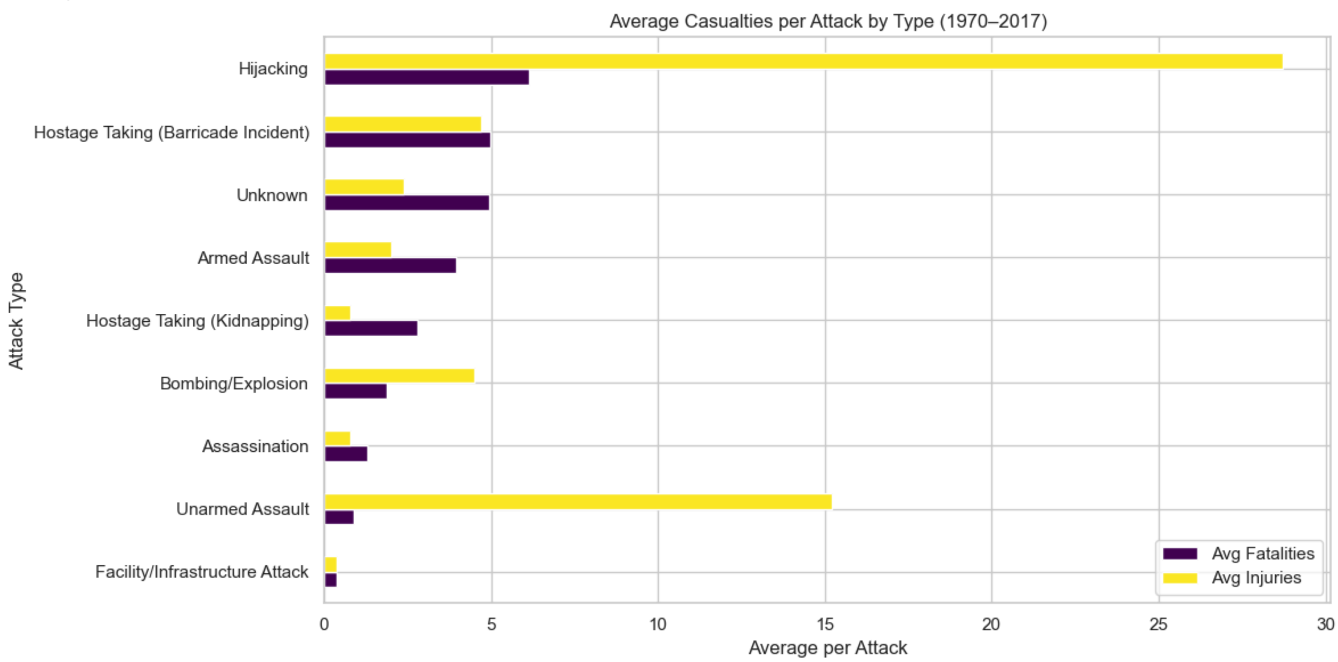
1. Distribution of attack types (counts + % share).
2. Fatalities & injuries by attack type (impact).
3. Average casualties per attack type (severity).




```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/3291311011.py:40: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  impact_df = (df_clean.groupby('attacktype1_txt')[['nkill','nwound']]
<Figure size 1200x600 with 0 Axes>
```




```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/3291311011.py:56: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  severity_df = (df_clean.groupby('attacktype1_txt')[['nkill','nwound']]
<Figure size 1200x600 with 0 Axes>
```




```
Most common attack type: Bombing/Explosion (88,255 attacks)
Deadliest type (total fatalities): Armed Assault
Most severe type (avg fatalities/attack): Hijacking
```

Observations (By Attack Type):

- **Bombings/Explosions** dominate the modality mix (roughly half of incidents), followed by **Armed Assault**s and **Hostage-taking/Kidnapping**. Rarer modes such as Hijacking and Barricade incidents form a small share.

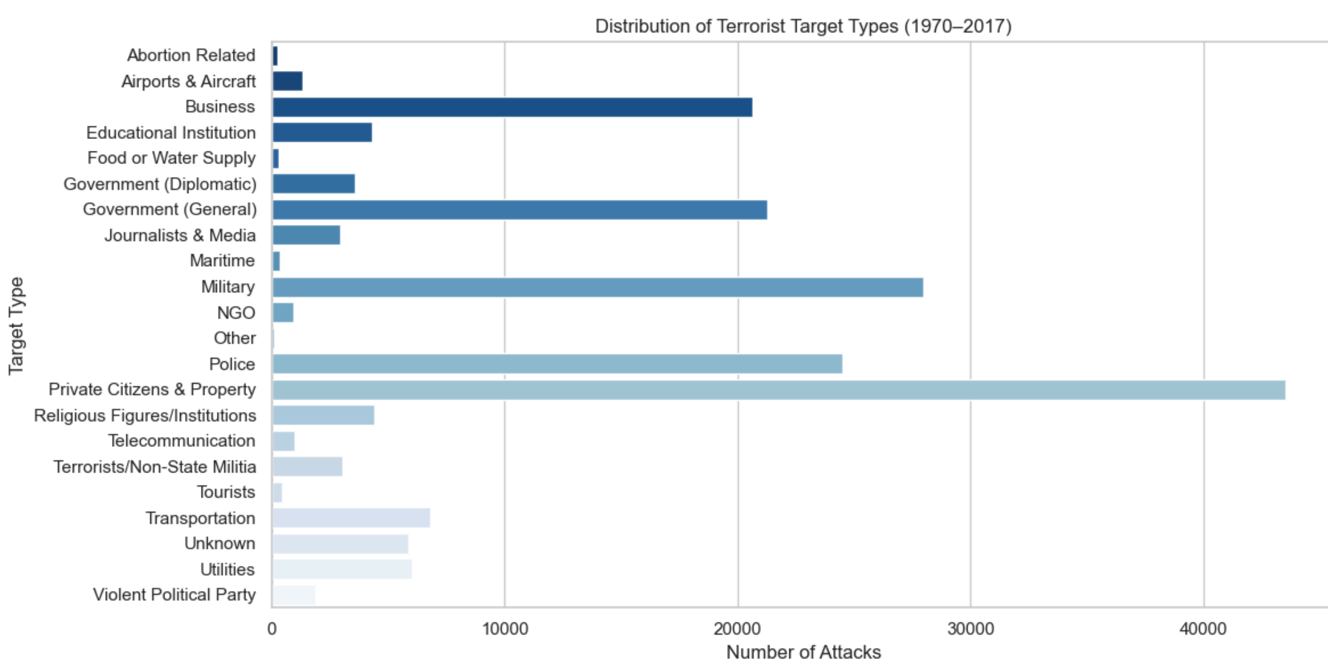
- For total harm, **Bombings/Explosions** drive the **most injuries**, while **Armed Assaults** contribute the most fatalities overall. Together they account for the bulk of casualties.
- **Severity ≠ frequency**: **Hijackings** show the **highest casualties** per incident; **Unarmed Assaults and Barricade-type hostage** takings also have **relatively high** per-attack injuries. **Facility/Infrastructure attacks and Assassinations** tend to have lower casualties per event.
- The “Unknown” category still produces substantial totals and non-trivial averages—flag it for careful handling or sensitivity checks in downstream analysis.
- Takeaway: Focusing only on common types (e.g., bombings) reduces many events, but guarding against low-frequency, high-severity modes (e.g., hijacking) is critical for worst-case risk.

4.4 By target

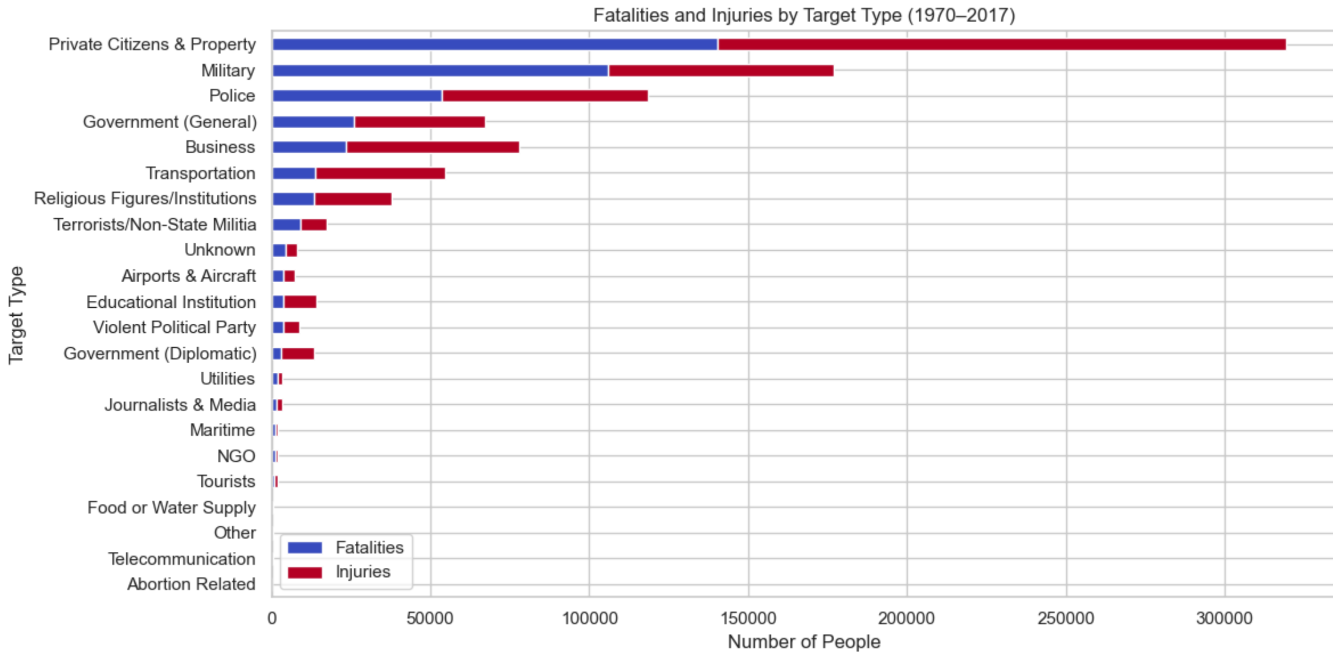
Why this matters. Target category shows *who or what* is attacked. Governments and security forces face frequent attacks, but the largest human cost often falls on **Private Citizens & Property**. I examine:

1. distribution of targets (counts + % share),
2. fatalities & injuries by target (impact), and
3. average casualties per attack (severity).

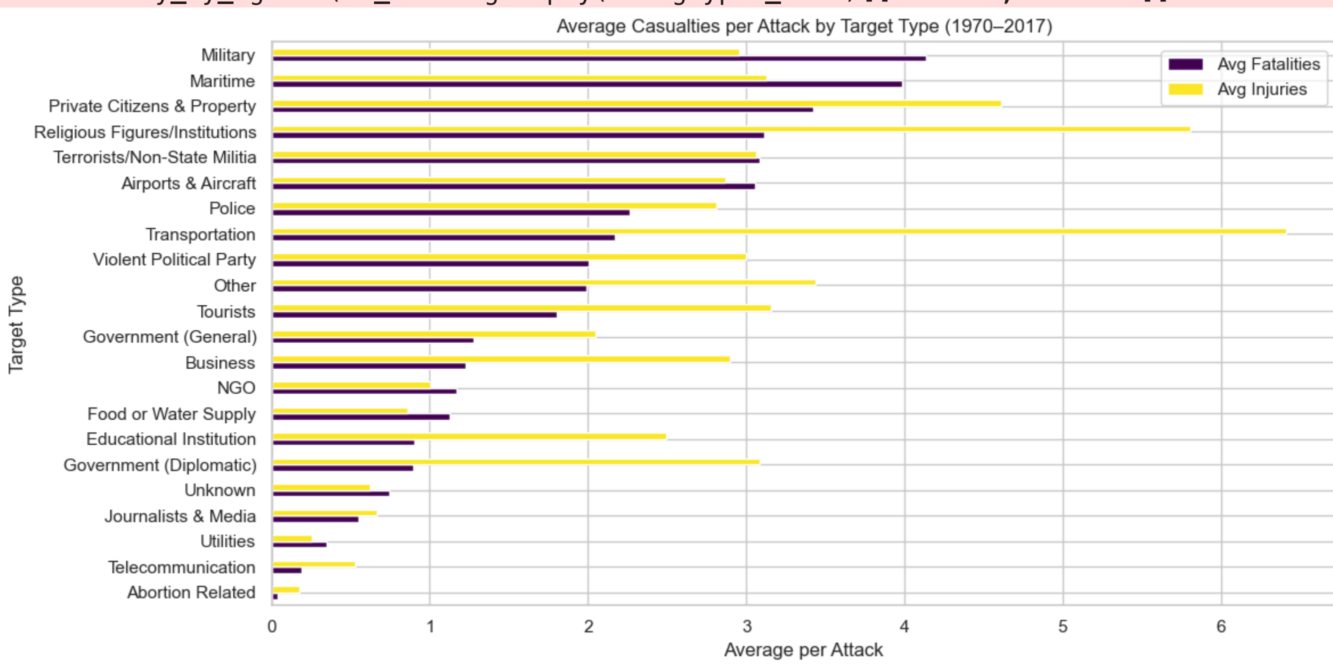
Note. I retain the original `targtype1_txt` labels and keep unknowns explicit rather than dropping them, to avoid bias.



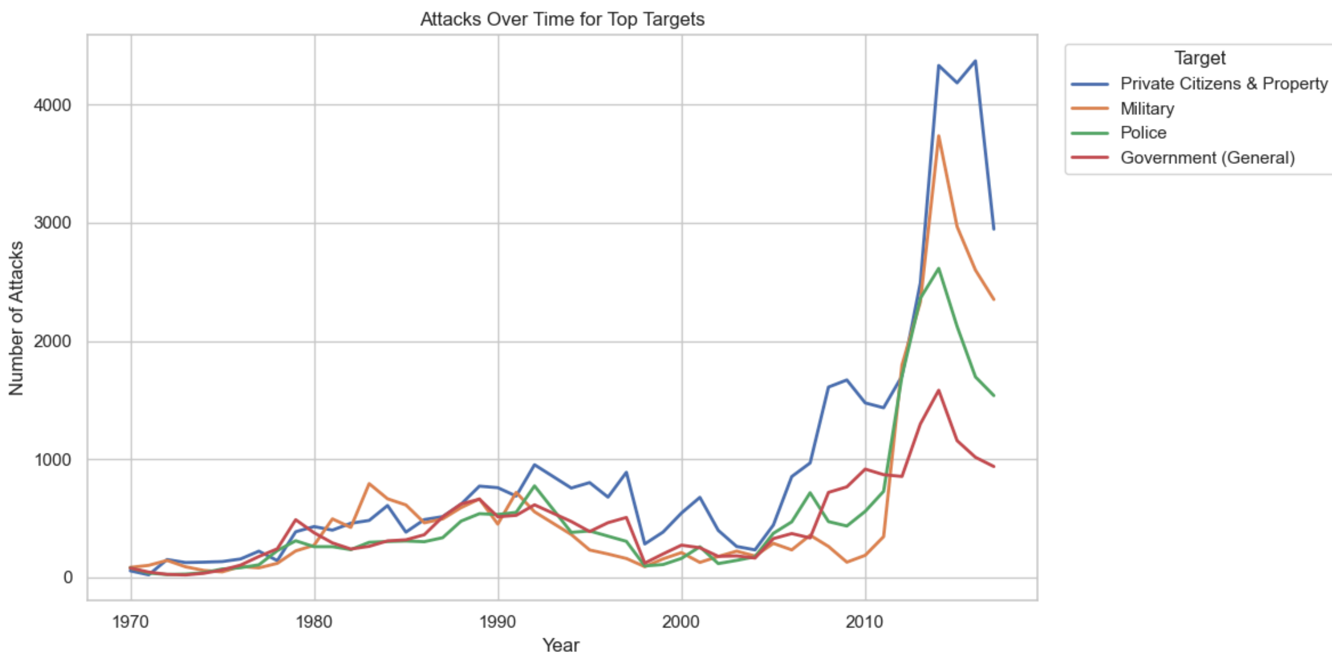
```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/3719769204.py:44: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  impact_by_tgt = (df_clean.groupby('targtype1_txt')[['nkill','nwound']]
```




```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/3719769204.py:61: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  severity_by_tgt = (df_clean.groupby('targtype1_txt')[['nkill','nwound']]
```




```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/3719769204.py:78: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  .groupby(['iyear','targtype1_txt']).size()
```




```
Most frequently targeted: Private Citizens & Property (43,511 attacks)
Highest total fatalities: Private Citizens & Property
Highest severity (avg fatalities/attack): Military
```

Observations (By Target):

- **Private Citizens & Property** are the most frequently targeted category by a wide margin, and also account for the **largest total casualties (fatalities + injuries)**.
- **State actors—Military, Police, and Government (General)**—collectively make up a large share of incidents and casualties, indicating persistent attacks on security and governance institutions.
- On a **per-attack severity** basis, targets with crowd density or confined settings—**Transportation, Airports & Aircraft, and Tourists**—show higher average casualties than most other categories (mass-casualty potential), even though they occur less often.
- **Religious Figures/Institutions** and **Terrorists/Non-State Militia** sit in the middle: not the most frequent, but with meaningful average harm per incident.
- The "Unknown" target category is non-trivial; results involving it should be interpreted cautiously as they reflect incomplete attribution.
- Trends over time: all top target groups surge after 2010, peaking around 2014, and then decline into 2016–2017—mirroring the overall timeline pattern seen earlier.

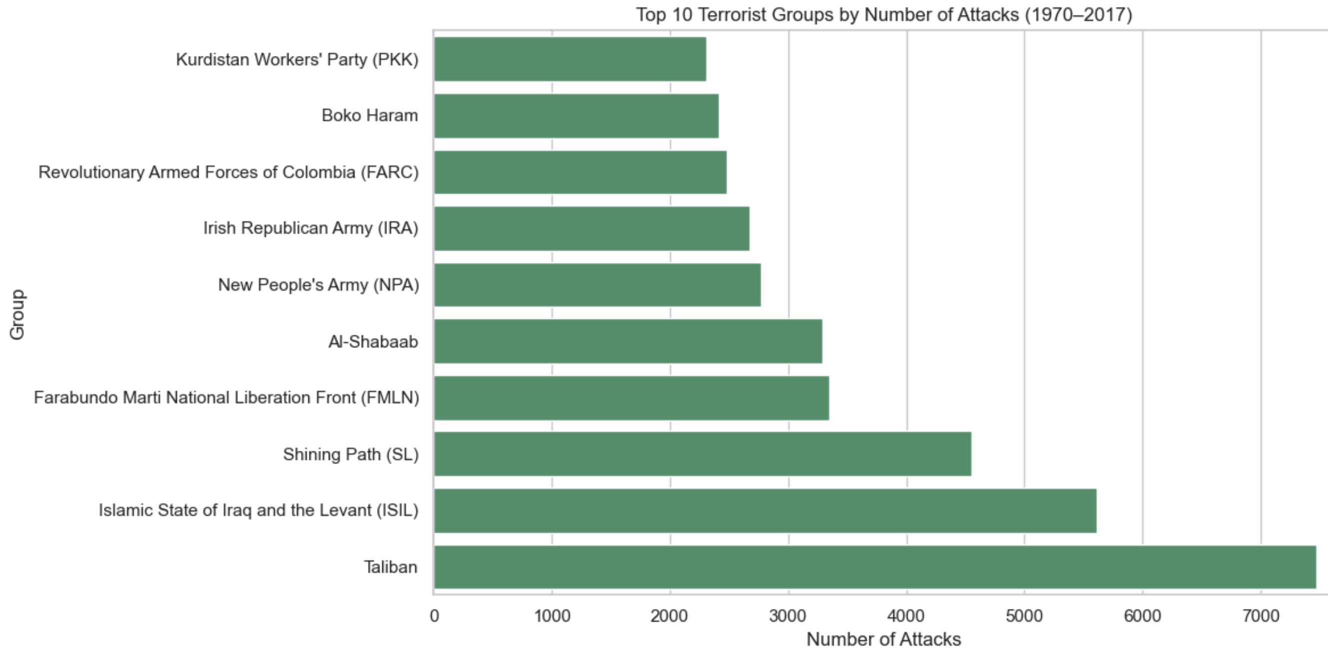
4.5 By group

Why this matters. Group analysis shows which organizations are most active, most lethal, and how their activity changes over time. Because many incidents have **Unknown** perpetrators, I (a) report that share explicitly, then (b) focus comparisons on named groups only.

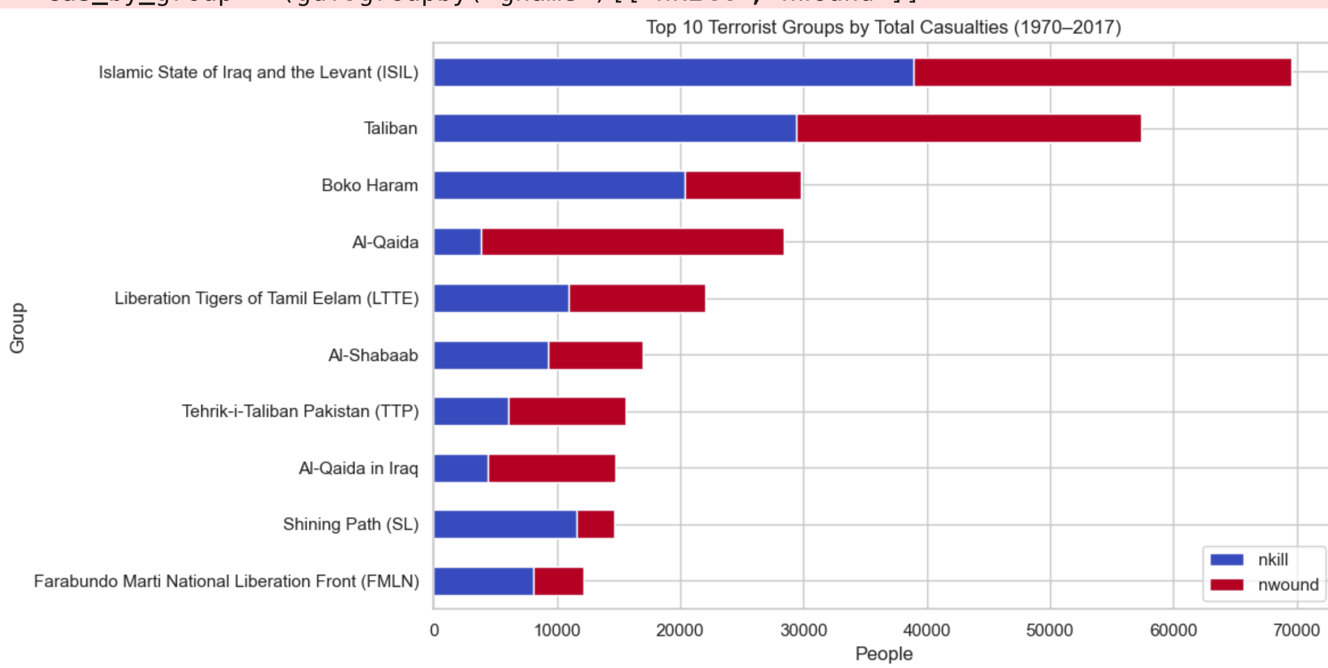
I present:

1. top groups by number of attacks (with % share),
2. total casualties (fatalities+injuries) by group,
3. activity over time for top groups.

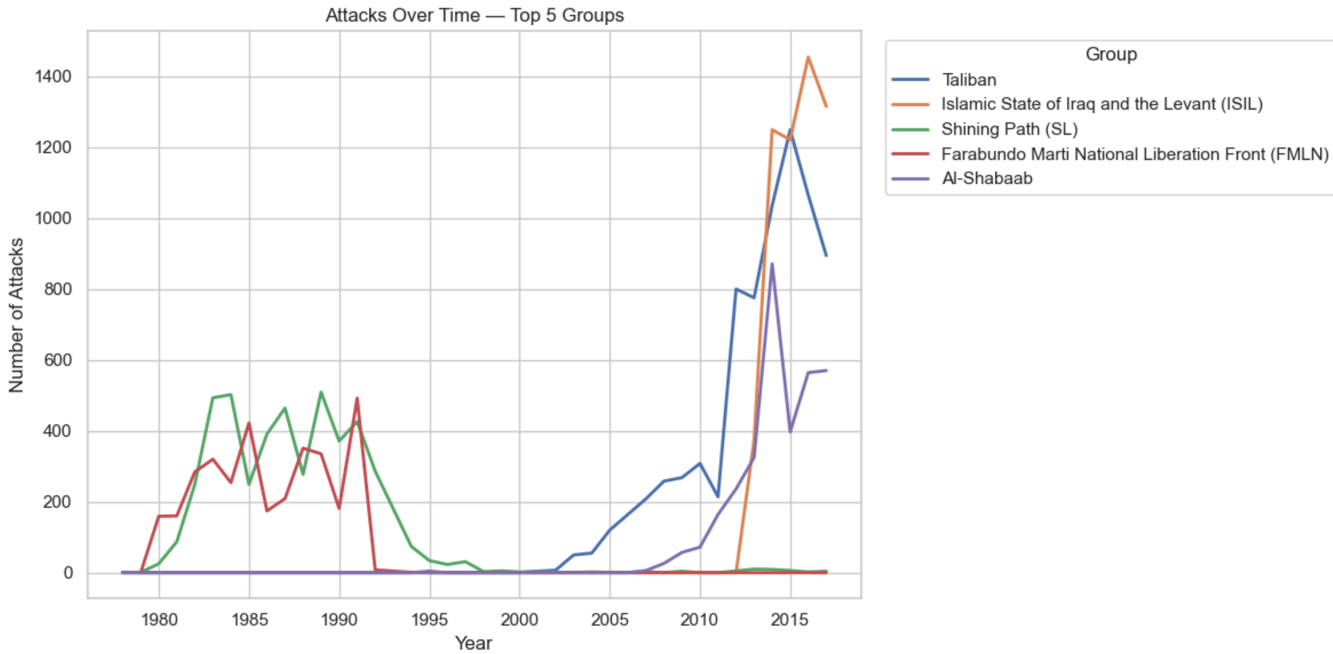
```
Share of incidents with Unknown group: 45.6%
```




```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/2041323191.py:41: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  cas_by_group = (gdf.groupby('gname')[['nkill','nwound']]
```



```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/2041323191.py:59: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
  .groupby(['iyear','gname']).size()
```




```
Most active group: Taliban (7,478 attacks)
Highest total casualties: Islamic State of Iraq and the Levant (ISIL)
```

Observations (By Group):

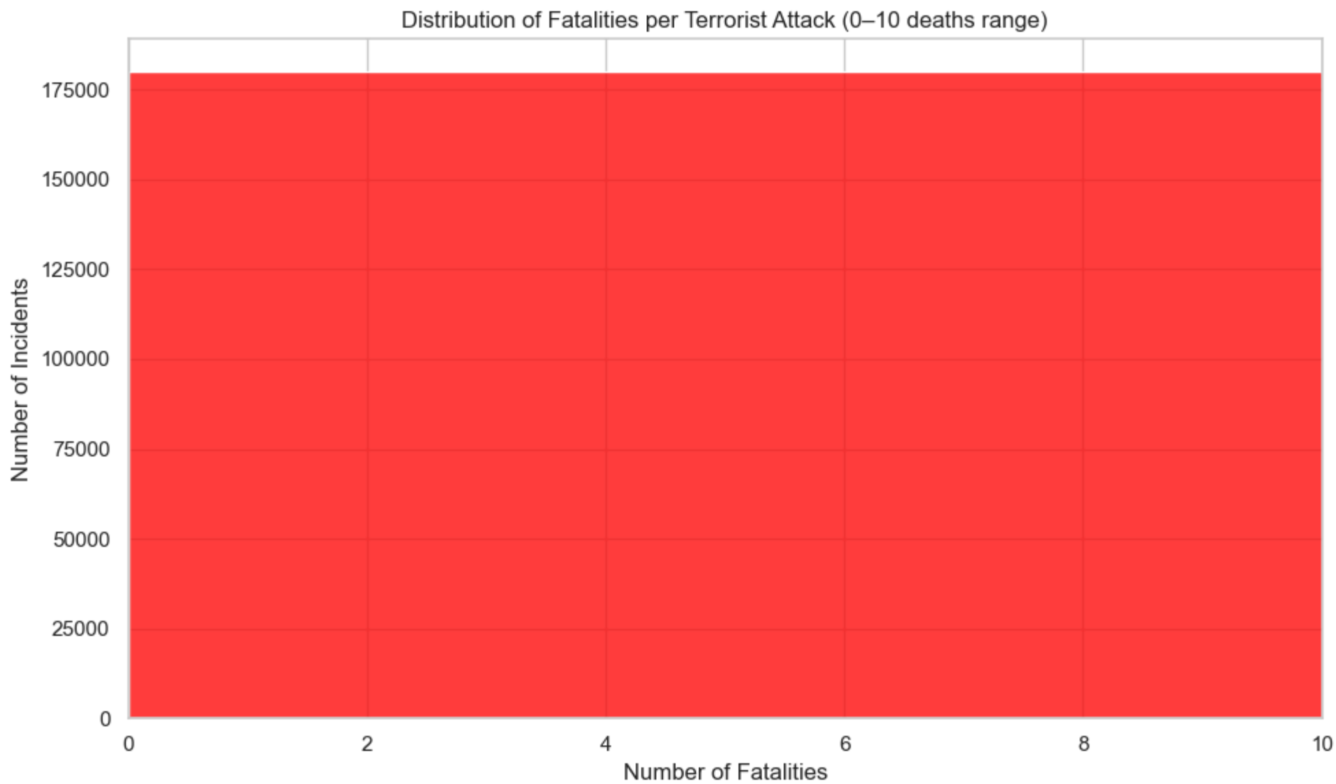
- Attribution coverage: A large share of incidents are unattributed (~45.6% “Unknown”), so **rankings reflect roughly the other half of the data**.
- **Most active by number of attacks** (descending): Taliban (clear #1), ISIL, Shining Path (SL), FMLN, Al-Shabaab, NPA, IRA, FARC, Boko Haram, PKK.
- **Highest total casualties** (fatalities + injuries): ISIL leads, followed by Taliban and Boko Haram; next are Al-Qaida, LTTE, Al-Shabaab, TTP, Al-Qaida in Iraq, Shining Path, FMLN.
- Temporal waves:
  - 1980s–early 1990s: activity centered on SL/FMLN/IRA.
  - Post-2010: sharp rise driven by Taliban/ISIL/Al-Shabaab, aligning with the global peak around 2014–2016.
  - Earlier insurgent groups taper off after the early 1990s; newer jihadist groups dominate the later surge.

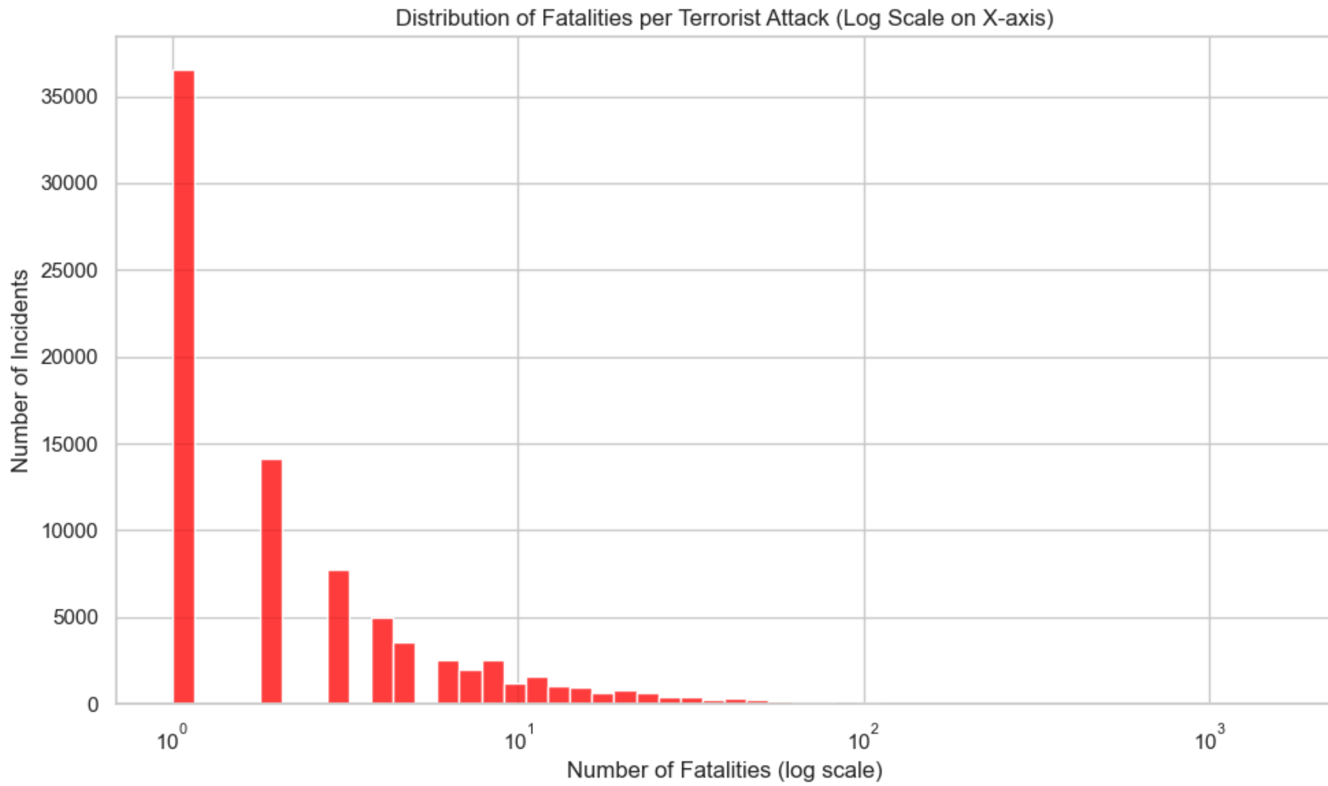
4.6 By impact

Why this matters. Impact measures the human and material cost of terrorism. While most attacks cause few casualties, rare catastrophic events dominate the headlines and global perception. By studying distributions, trends, and outliers, we see both everyday realities and exceptional extremes.

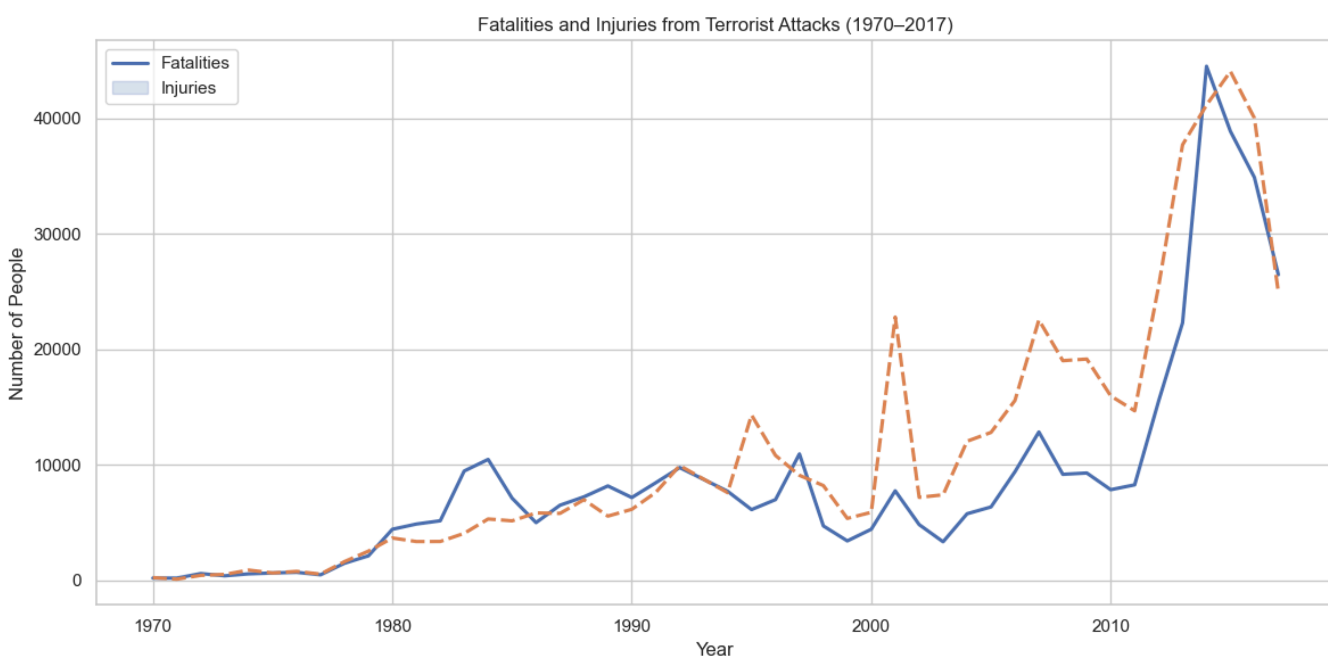
I analyse:

1. distribution of fatalities per attack (linear + log scale),
2. fatalities and injuries over time,
3. the top 10 deadliest individual incidents.





Distribution of Fatalities per Terrorist Attack (Log Scale on X-axis)
Number of Incidents
35000
30000
25000
20000
15000
10000
5000
0
10^0
10^1
10^2
10^3
Number of Fatalities (log scale)



Fatalities and Injuries from Terrorist Attacks (1970–2017)
Fatalities
Injuries
Number of People
40000
30000
20000
10000
0
1970
1980
1990
2000
2010
Year

	iyear	country_txt	city	attacktype1_txt	gname	nkill
0	2014	Iraq	Tikrit	Hostage Taking (Kidnapping)	Islamic State of Iraq and the Levant (ISIL)	1570
1	2001	United States	New York City	Hijacking	Al-Qaida	1384
2	2001	United States	New York City	Hijacking	Al-Qaida	1383
3	1994	Rwanda	Gikoro	Armed Assault	Hutu extremists	1180
4	2014	Iraq	Sinjar	Hostage Taking (Kidnapping)	Islamic State of Iraq and the Levant (ISIL)	953
5	2014	Iraq	Badush	Armed Assault	Islamic State of Iraq and the Levant (ISIL)	670
6	2017	Somalia	Mogadishu	Bombing/Explosion	Al-Shabaab	588
7	2004	Nepal	Dhading District	Armed Assault	Communist Party of Nepal-Maoist (CPN-M)	518
8	2014	Syria	Unknown	Hostage Taking (Kidnapping)	Islamic State of Iraq and the Levant (ISIL)	517
9	2016	Syria	Palmyra	Hostage Taking (Kidnapping)	Islamic State of Iraq and the Levant (ISIL)	433

```
Deadliest incident year: 2014
Deadliest incident fatalities: 1570
Group responsible: Islamic State of Iraq and the Levant (ISIL)
```

Observations (Impact):

- Fatalities per attack are highly skewed—most incidents cause 0–1 deaths, with a long tail of rare mass-casualty events.
- Injuries usually exceed fatalities each year, and both series move together.
- Sharp peaks occur in 2001 and 2014–2016, when totals hit their highs.
- Annual spikes are driven by a few catastrophic attacks (e.g., 9/11; ISIL events in 2014–2016), not by broadly higher lethality per incident.

4.7 visual telling

Executive summary: Terrorist incidents rose from the 1970s and peaked in 2014 (16,903 attacks) before declining. The burden concentrates in MENA and South Asia; Iraq alone accounts for ~13–14% of all incidents. Bombings/Explosions are the most common method, while Private citizens & property are the most frequently targeted. Casualty impact is heavy-tailed—most attacks cause few deaths, but rare events drive the totals. Among named actors, Taliban is the most active; ISIL contributes the most casualties over the period. The post-2014 drop is visible across regions, especially in MENA.

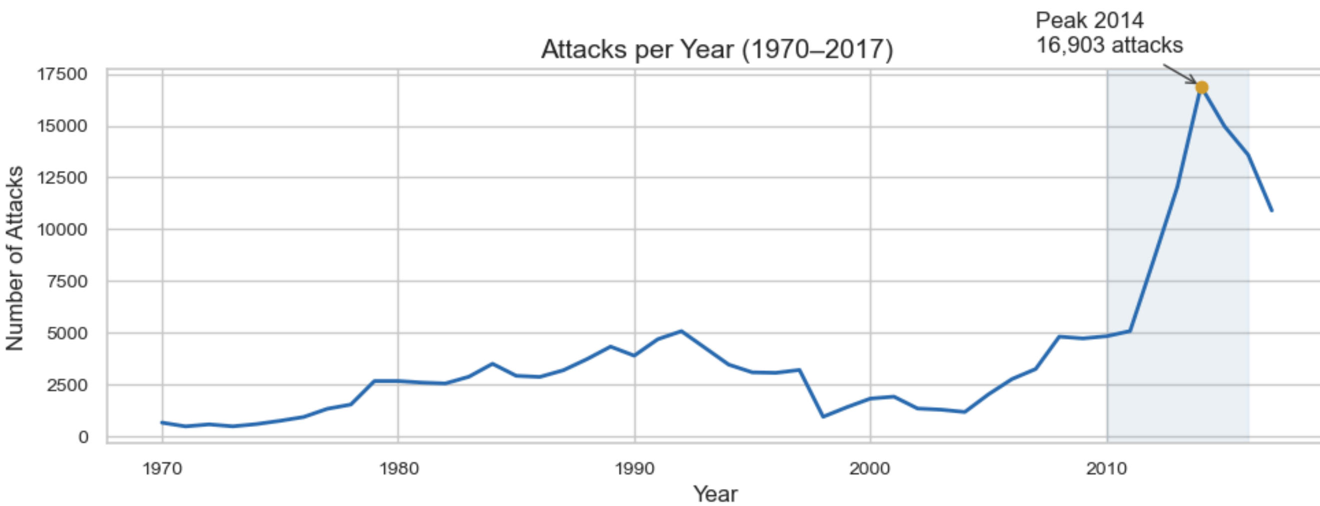
Global terrorism, 1970–2017

181,691
Total attacks

2014 (16,903)
Peak attacks year

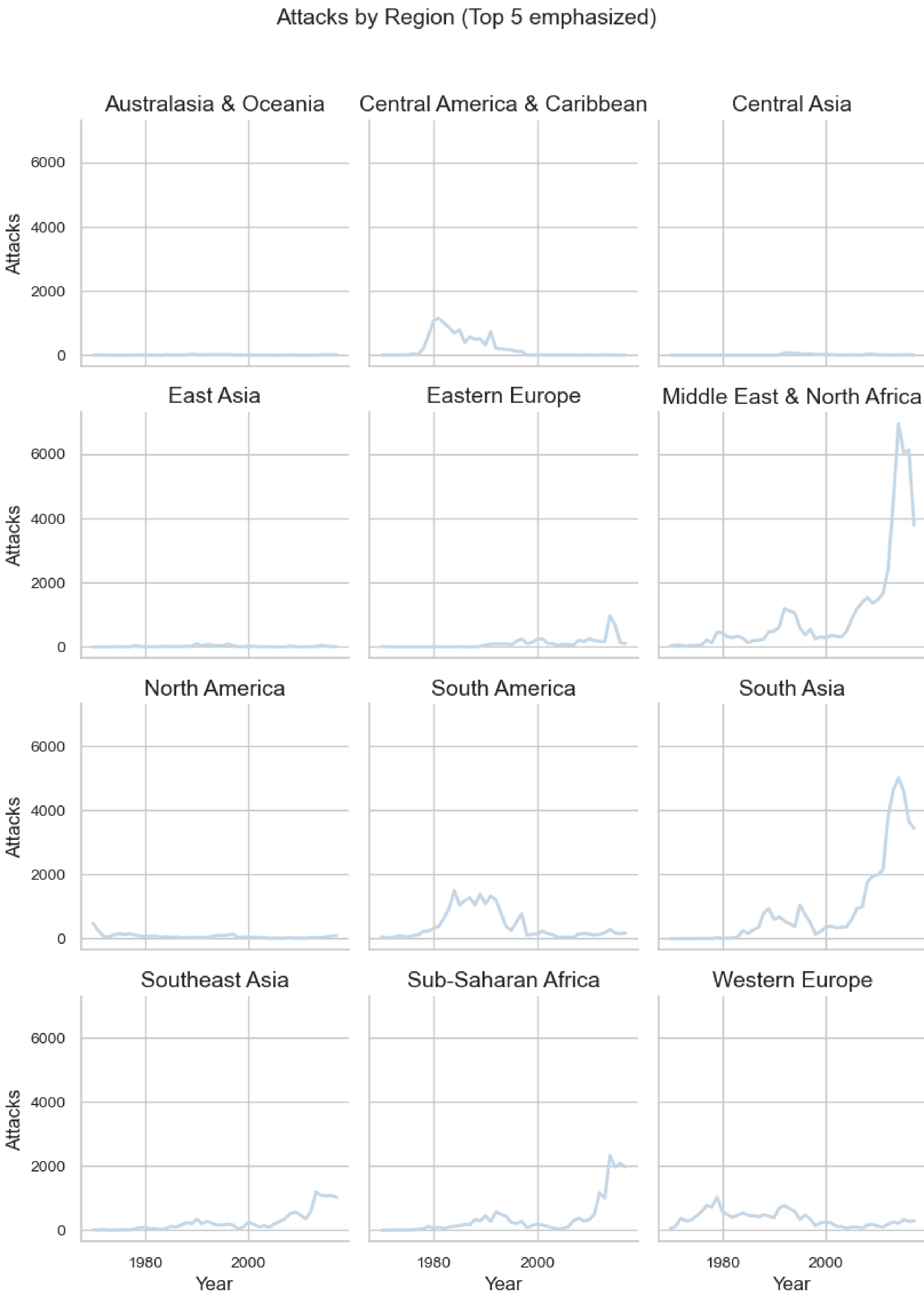
2014 (85,618)
Peak casualties year

Most incidents are low-lethality; spikes come from rare, catastrophic events.

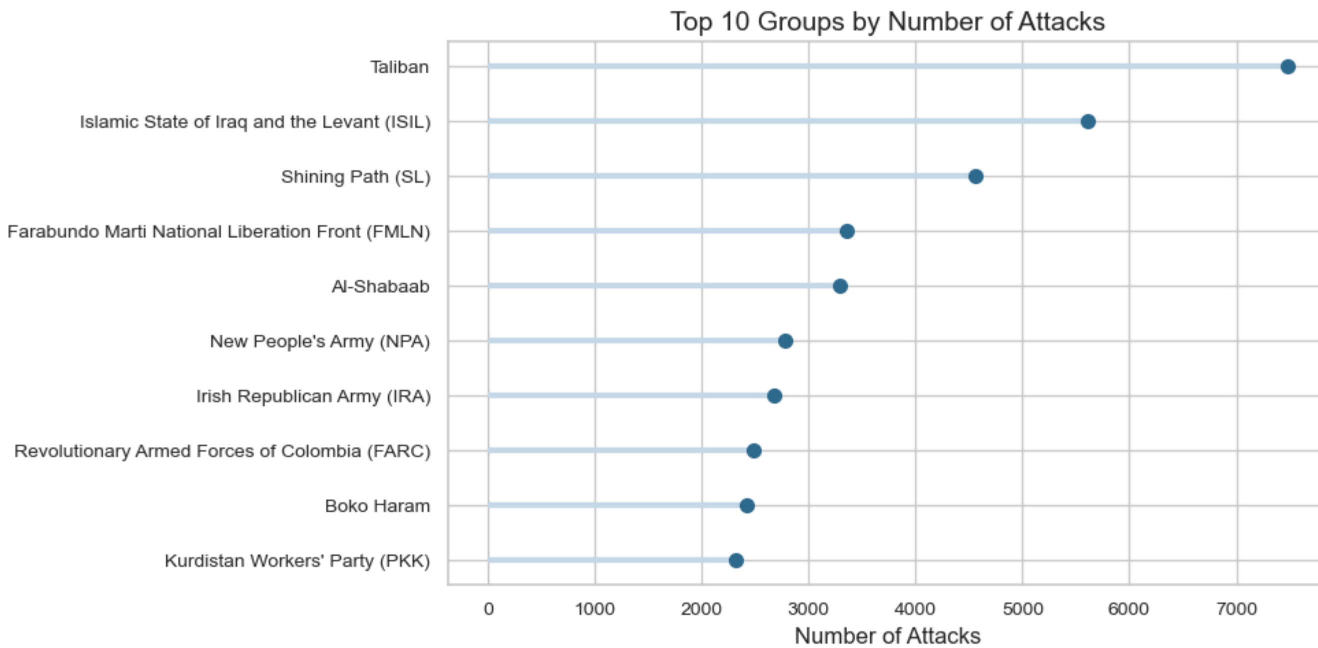


Incidents surged after 2010, peaking in 2014.

```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/361891141.py:9: Futur
eWarning: The default of observed=False is deprecated and will be changed to True in
a future version of pandas. Pass observed=False to retain current behavior or observe
d=True to adopt the future default and silence this warning.
  yr_reg.groupby('region_txt')['attacks'].sum()
```

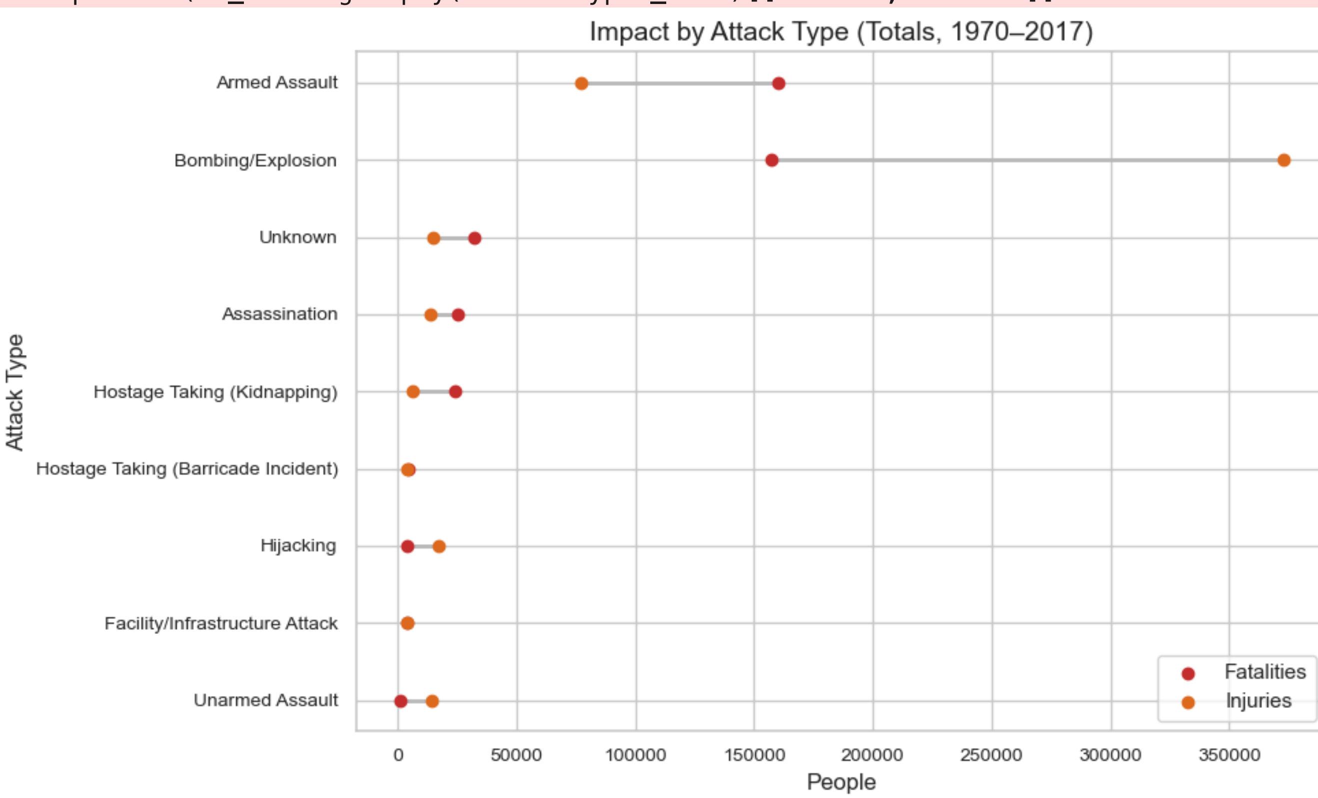

Attacks by Region (Top 5 emphasized)
Australasia & Oceania
Central America & Caribbean
Central Asia
East Asia
Eastern Europe
Middle East & North Africa
North America
South America
South Asia
Southeast Asia
Sub-Saharan Africa
Western Europe
Attacks
0
2000
4000
6000
1980
2000
Year

The burden concentrates in MENA and South Asia; other regions stay comparatively low



A few groups account for a large share of activity

```
/var/folders/m9/p4hgm4s12ss7ncmyttqf3_380000gn/T/ipykernel_1437/908221168.py:1: Futur
eWarning: The default of observed=False is deprecated and will be changed to True in
a future version of pandas. Pass observed=False to retain current behavior or observe
d=True to adopt the future default and silence this warning.
  impact = (df_clean.groupby('attacktype1_txt')[['nkill','nwound']]
```



Bombings/Explosions are common; injuries generally exceed fatalities

5. Conclusion

- Global terrorism rose from the 1970s, **peaked** in 2014, then declined but remains above pre-2005 levels.
- Burden is concentrated in **MENA and South Asia**; **Iraq** alone contributes **~13–14%** of incidents.
- **Bombings/Explosions dominate method**; Private citizens & property are the most frequent targets.
- Impact is heavy-tailed—most attacks cause few deaths, but a small set of events drives totals.
- Among named actors, **the Taliban is the most active by count**; ISIL accounts for the largest total casualties.
- The post-2014 drop is visible across multiple regions, especially MENA.

```
  Cell In[155], line 1
    jupyter nbconvert "Global Terrorism.ipynb" --to html --TemplateExporter.exclude_i
nput_prompt=True
            ^
SyntaxError: invalid syntax
```