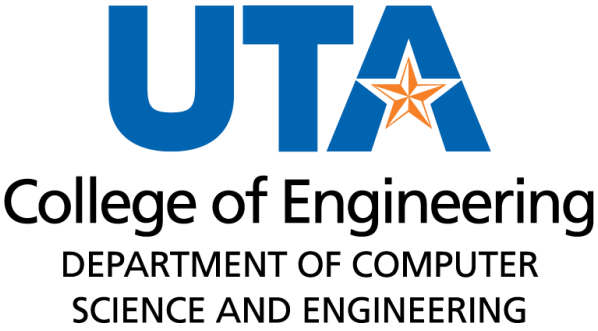
Data Visualization Report

(DASC-5305-002)

Submitted By:

ANUVA NEGI

(1002238067)



College of Engineering
DEPARTMENT OF COMPUTER
SCIENCE AND ENGINEERING

Table of Contents

Introduction

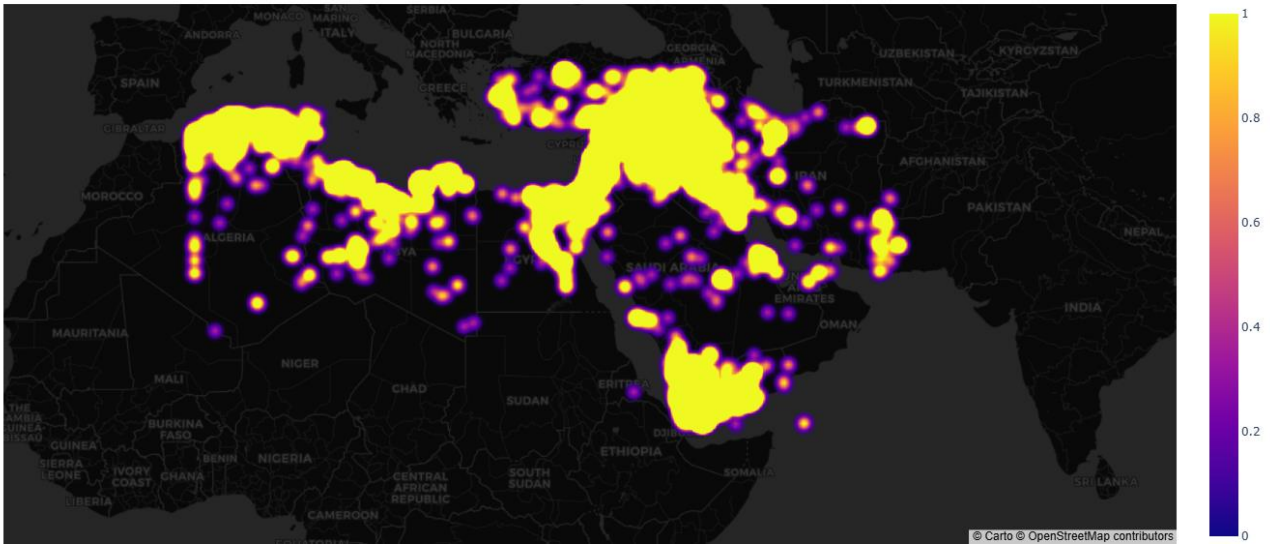
This report examines data on terrorist attacks in the Middle East and North Africa **(MENA) region** (i.e. region-10.2.csv). The dataset offers in-depth insights into different types of attacks, the groups targeted, and the organizations behind them over a defined timeframe. Grasping these patterns is essential for policymakers, security agencies, and researchers like data scientists to observe patterns and evaluate future threats and formulate effective countermeasures.

Objective

To identify patterns and trends that offer useful insights for future decision-making for effective measures. Hence main goals of this analysis are:

- To identify trends in terrorist attacks over time.
- To analyze the groups and locations that are most frequently targeted.
- To examine various types of attacks and the organizations behind them.
- To gain insights into attack patterns, peak activity periods, and possible connections to socio-political events.

Filtered Heatmap of Attacks in MENA Region

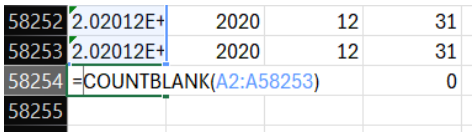


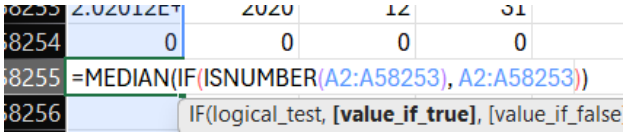
Python versus Excel

1. A Comparative Study of EDA and Data Visualization

Performing EDA (also called Exploratory Data Analysis) in **Excel** was a *slow, tedious process* at times involving the manual identification, filtering, and removal of extraneous columns.

While functions such as *COUNTBLANK(), ISNUMBER(), and MEDIAN()* proved practically useful in cleaning the data, the fact remains that missing values and inconsistencies required a more involved effort. And, of course, with large datasets, the system often struggled in Excel, barely making it through before freezing or crashing.

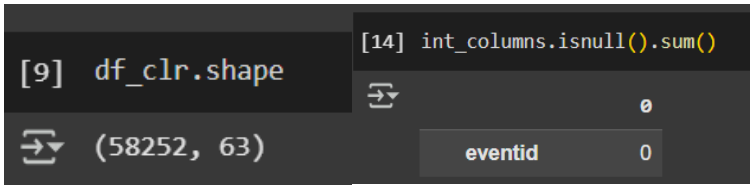




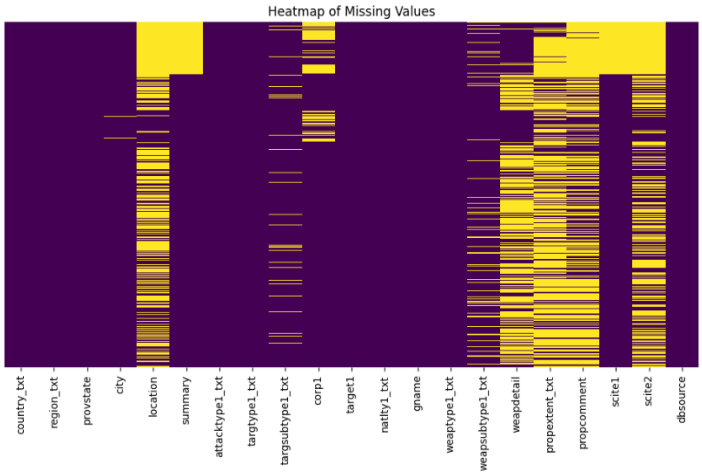
In comparison, **Python** transformed the EDA into far less tedious and *more scalable operations*. Functions such as *isnull().sum(), isna(), fillna(), and median()* did most of the work automatically, leaving less for the human operator to complete. Plots like box plots and heatmaps also helped

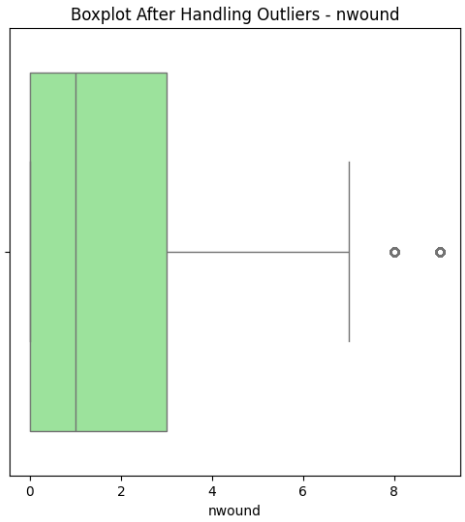
In addition, although it is convenient to roll back your code changes in Python, *rolling back multiple actions in Excel would require some extensive maneuvering*. Python offers other great tools such as *shape(), info(), and describe() that would give instant insights into the dataset after modification*.

Such features allowed for *the faster, more organized, and more organized data exploration that is scalable to a larger data level*.



```
# Impute 'longitude' and 'latitude' with their respective medians
dff['longitude'] = dff['longitude'].fillna(dff['longitude'].median())
dff['latitude'] = dff['latitude'].fillna(dff['latitude'].median())
```





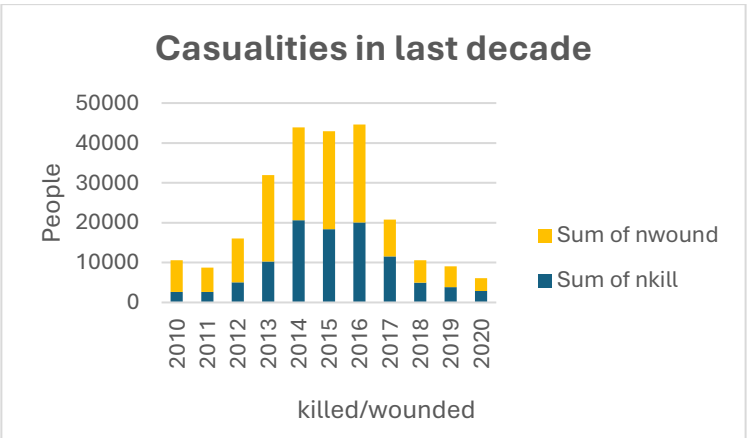
Above are few snips from my Python code where some visualizations even made preprocessing stage more vivid.

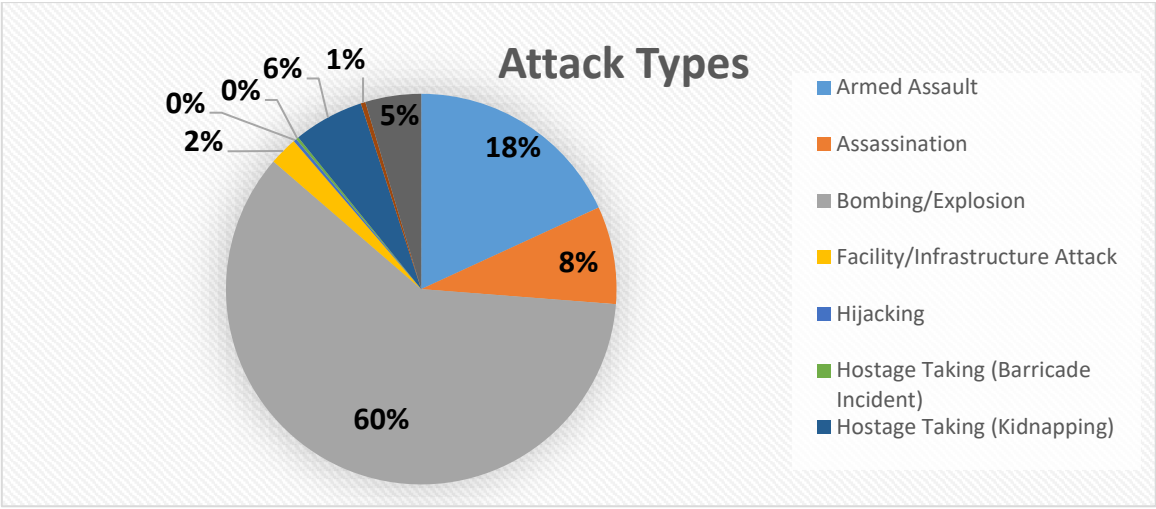
2. Data Visualization

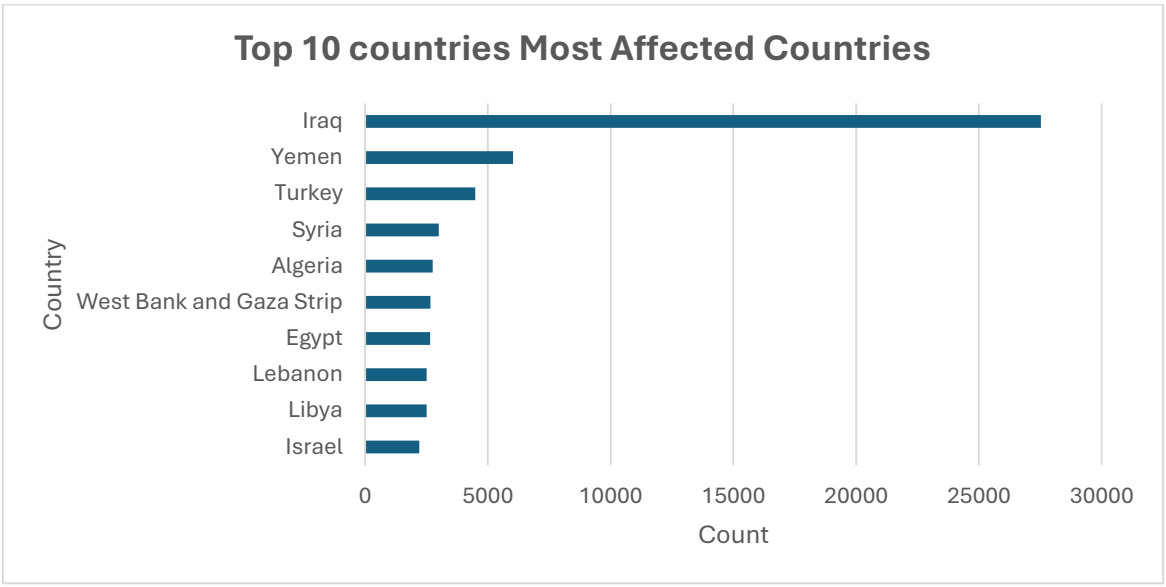
Both Excel and Python offer strong visualization features, but they excel in different aspects.

Excel's *pivot tables* have become a must in the efficient summarization of data and are well-suited for producing quick insights by means *of bar, column, and line charts*. On the flip side, Excel's options for visualization were somewhat limited in terms of interactivity and ability to handle larger datasets.

Row Labels	Sum of nkill	Sum of nwound
2010	2663	7881
2011	2658	6112
2012	5070	10944
2013	10288	21679
2014	20619	23276
2015	18391	24586
2016	20058	24592
2017	11527	9223
2018	4939	5675
2019	3868	5155
2020	2838	3272
Grand Total	102919	142395

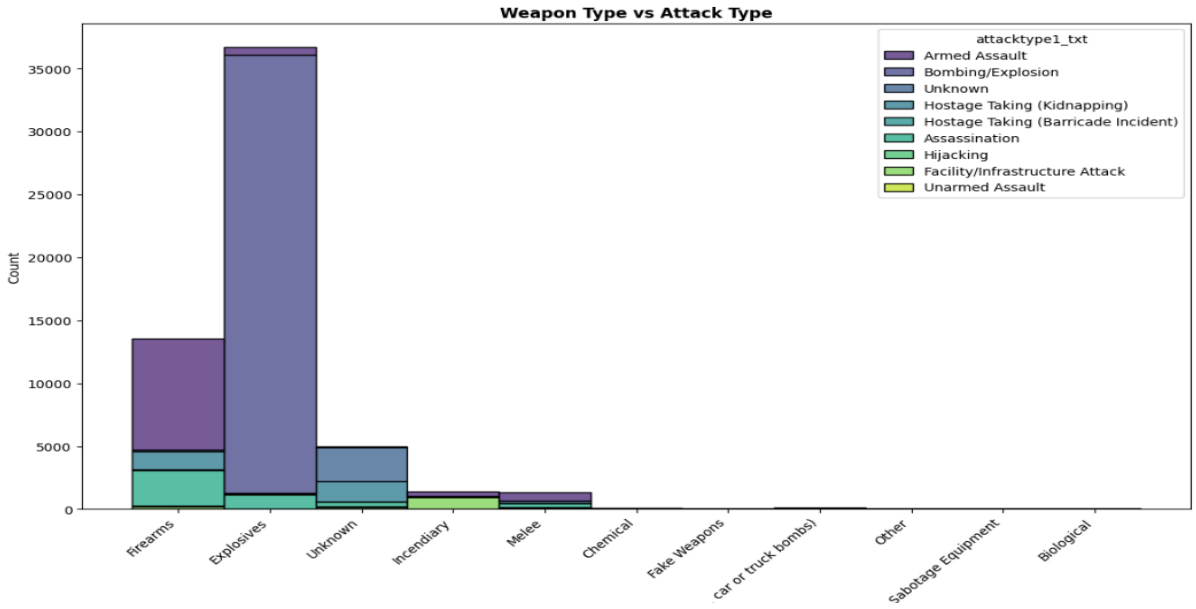


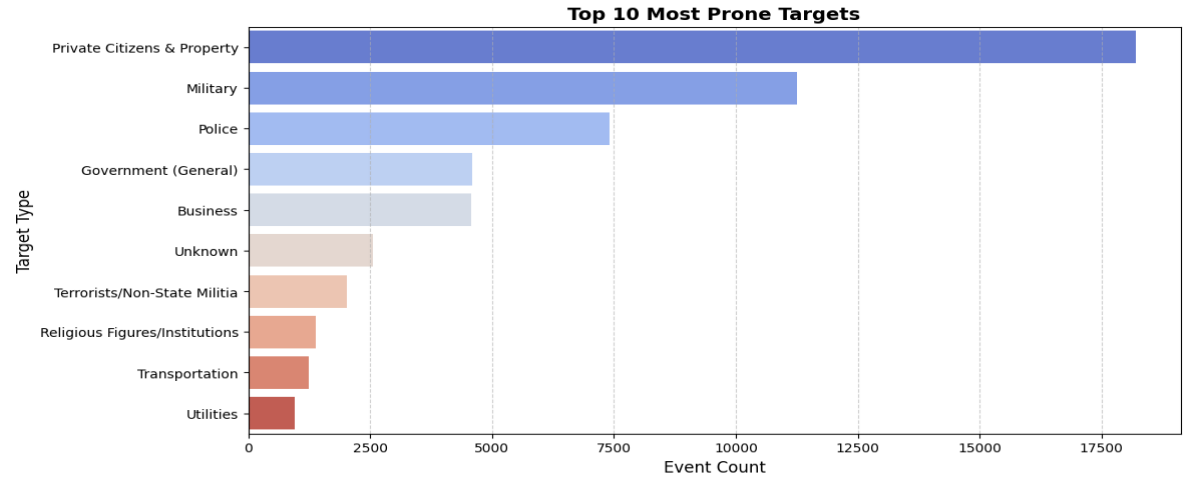


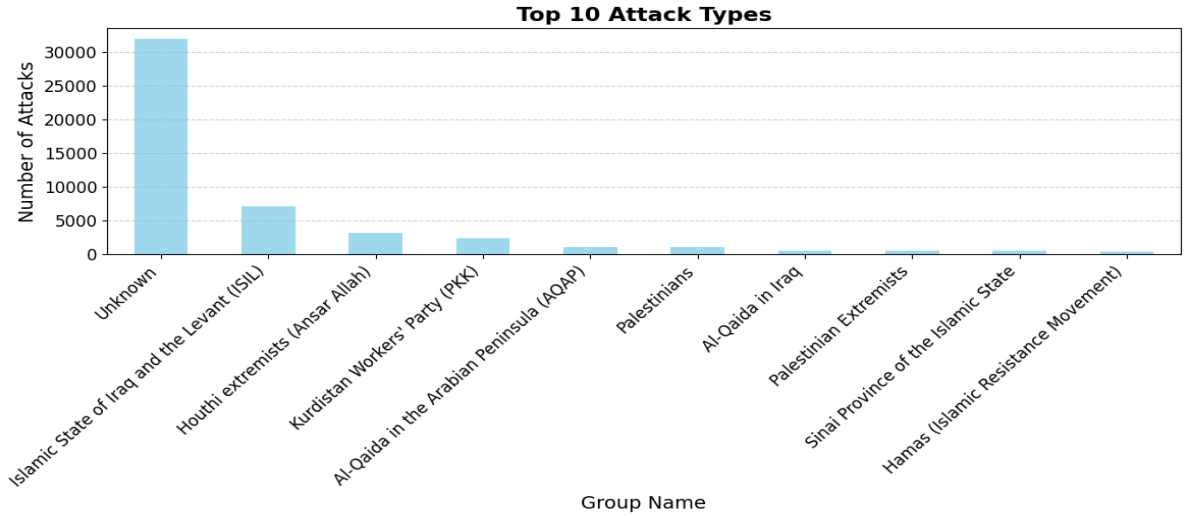


Here we have data for top 10 most affected countries by attacks and types of attacks observed based on the data given.

On the other hand, **Python** has more *flexibility* and power when it comes to data visualization, particularly for libraries like *Matplotlib, Seaborn, Plotly, and Folium*. Using Plotly and Folium I was able to create *interactive heatmaps and geographical visualizations*, providing a much more visually appealing view into the attack patterns across the MENA region (on page 2). Moreover, Python allows for custom styling and enables.



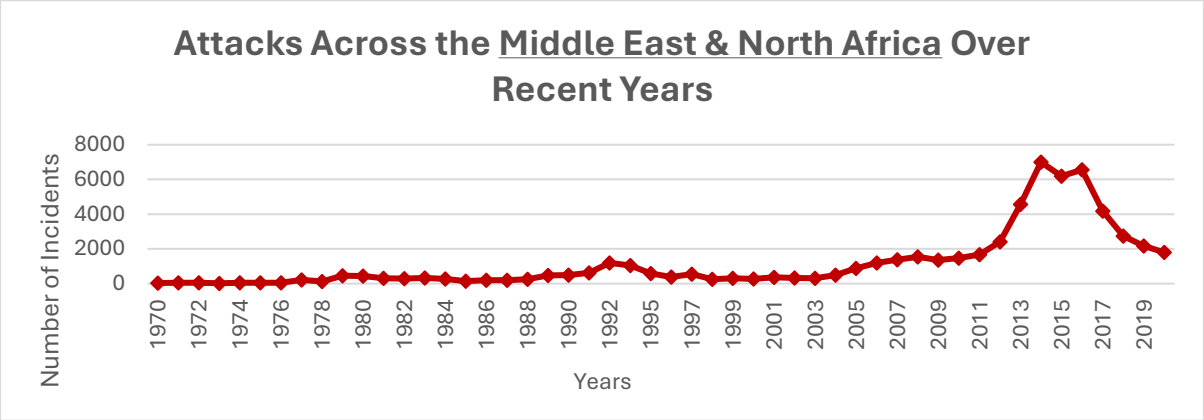


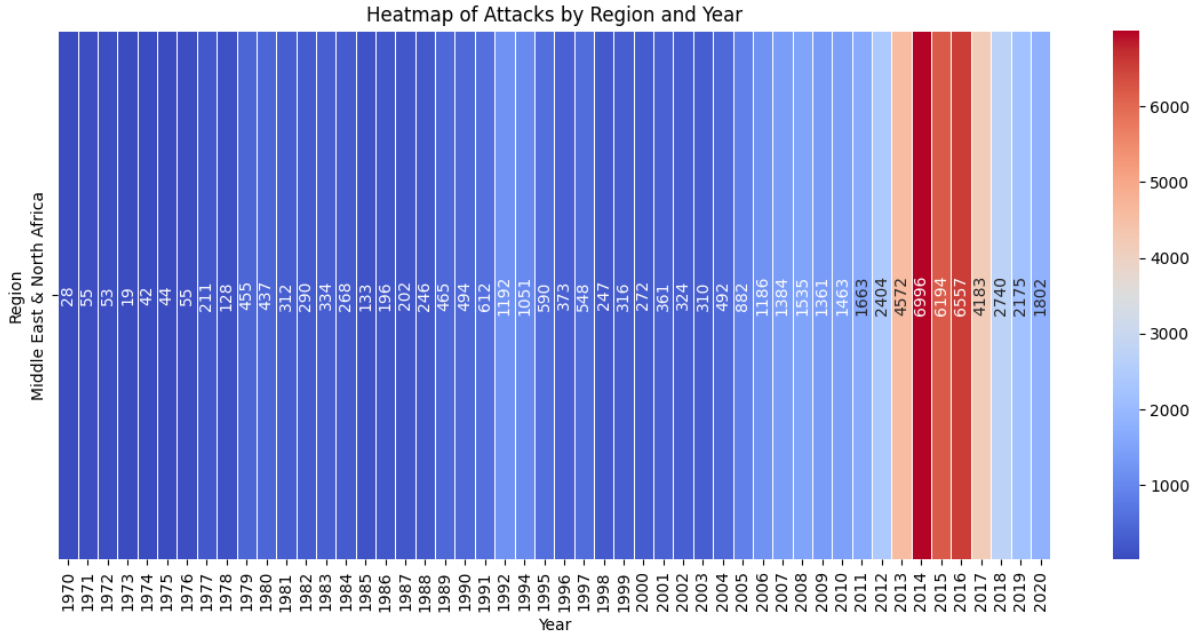


Here, most attacks are aimed at **civilian, military, and law enforcement targets**, highlighting an intent to undermine public safety and stability.

The data reveals a notable drop in event counts following the top three categories, with **government and business targets still experiencing considerable impacts**, while **transportation, utilities, and religious institutions encounter relatively fewer incidents**.

Below we have **Line Plot from Excel and Heatmap from Python**





Overall, while **Excel remains a strong tool for basic data exploration and visualization**, **Python offers greater flexibility, automation, and efficiency**, making it the preferred choice for handling **complex datasets and in-depth analytical tasks**.

Key Findings from the Data

- Surge in Attacks (2013-2017): There was a *particularly high number of attacks conducted between 2013 and 2017*, with **2014 being the most active year in terms of incidents**.

- Iraq as the Hotspot: **Iraq** consistently *reported maximum attacks* even compared to its other top 10 most-affected countries.

- Weapon Usage Trends: *Explosives turned out to be the most often used weapon*, mainly associated with bombing and explosive attacks, while melee, incendiary, and other weapon types used remained far more sporadic, pointing toward the predominant reliance on high-impact weapons such as explosives and guns rather than close-contact or unconventional means.

- Casualty Trends: *The highest number of casualties (fatalities-injuries) were between 2014 and 2017*, followed by the onset of a decline in the following years. Evidently**, there were more injuries compared to deaths**, suggesting that most of the attacks led to a higher number of injuries than to deaths.

- Targeted Sectors: *Civilians, military personnel, and law enforcement agencies were majorly targeted*. This would reflect a conscious **intention to generate a state of insecurity** and unsettle public safety and governance.

- Sectoral Impact Disparity: While the government and commercial sectors bore the brunt of heavy attacks, the incidents in transportation, utilities, and religious establishments were significantly lower.

- Perpetrators and Group Involvement: The Islamic State of Iraq and the Levant (**ISIL**), the **Houthi extremists**, and **PKK** (Kurdistan Workers' Party) were responsible for the *largest number of attacks* while the other groups were connected with comparatively lesser incidents.

Conclusion

According to my analysis, attacks rapidly increased from 2013 to 2017, with 2014 marking the peak year. The country that suffered most from these attacks is Iraq, with explosives as the most used weapon, killing civilians, military personnel, and law enforcement, emphasizing the cutting of public security efforts.

In terms of tools, Excel is easy to use with excellent visualization and pivot table support but struggles with larger datasets and manual data cleaning processes. Python can automate and scale the data analysis process while allowing for greater customization in the types of visualizations an analyst or data scientist may require. Although both programs have such advantages, it must be allowed for Python to win out, as it does this job in a much more efficient way and finds out key insights more easily.

References

Dataset for Region 10-2:
https://uta.instructure.com/files/36089566/download?download_frd=1&verifier=G9r99s7UBd2mGDZJFjw4LAhVbJpdWDroU2AltKMb