# GLOBAL TERRORISM

A Project based on Exploratory and Predictive Analysis

## CSSM – 502
### ADVANCED DATA ANALYSIS WITH PYTHON

❖


**Final Project**
**by Selin Susem**
0078637


❖

**GitHub Page**
https://github.com/SelinSusem/CSSM-502.git

# Table of Contents

## I. BACKGROUND

To be able to understand the details and analysis about terrorist acts in this Project, one needs a clear and consistent definition of what terrorism is, and how it's different from any other form of violence.

International terrorism is defined as "violent, criminal acts committed by individuals and/or groups who are inspired by, or associated with, designated foreign terrorist organizations or nations (state-sponsored)", while domestic terrorism is defined as "violent, criminal acts committed by individuals and/or groups to further ideological goals stemming from domestic influences, such as those of a political, religious, social, racial, or environmental nature".

Over the past decade, terrorists killed an average of 21,000 people worldwide each year. The global death toll from terrorism over the past decade ranged from 8,000 in 2010 to a high of 44,000 in 2014. In 2017, terrorism was responsible for 0.05% of global deaths.

Public concern about terrorism is high. In many countries more than half say they are concerned about being a victim. Moreover, media coverage of terrorism is often disproportionate to its frequency and share of deaths. So, we try to understand how the number of terrorist acts varies around the world and how it has changed over time and try to discover whether we can predict some characteristics of it for the purpose of risk mitigation.

## II. PURPOSE

The purpose of this Project is to touch upon a societal issue with data analysis in Python which is **the problem of terrorism** as a hot button problem. The Project will be composed of two parts:

**(1) exploratory analysis**

**(2) predictive analysis**

After data pre-processing process, in the first part of the Project, dynamics and patterns of global terrorism is tried to be understood and visualized. For that purpose, this first part is focused on exploratory data analysis (EDA) by asking critical questions, exploring the data in accordance with those initial questions and more and then visualizing the answers. During these analysis, most important questions asked in that part are as follows:

- How has the number of terrorist acts changed over time?
- Has the number of attacks increased during the recent years?
- What is the percentage of successful attacks in total?
- What is the proportion of successful/unsuccessful terrorist attacks over the years?
- Where is most of the casualties?
- How have casualties evolved throughout the years?
- Are certain nationalities more targeted? Is this related with freedom schedules?
- Which attack types are popular?
- Which weapon types are popular?
- Who are the targets?
- What are the targets by attack and weapon type?

In the second part of the Project, predictive analysis has been performed. The questions that are asked here are as follows:

- Can casualties be predicted from GTD-exclusive features (our main dataset)?
- Whether a terrorist attack is expected to succeed or fail?

## III. DATA

**Global Terrorism Data (GTD)**, which is an open-source database including information on terrorist events around the world from 1970 through 2017 with annual updates, is used for this Project. The dataset can be download from https://www.start.umd.edu/gtd/contact/download . Unlike many other event databases, the GTD includes systematic data on domestic as well as international terrorist incidents that have occurred during this time period.

For a quick look at the variables of the data, a short data dictionary is commented at the beginning of the Python notebook of the Project. However, the complete descriptions of the variables can be found in the **codebook provided by Global Terrorism Database** which is attached to the appendix of this Report.

The original data used for this Project is composed of **135 columns*181.691 rows.** After pre-processing phase and eliminations these numbers have changes and every changed dataset during the Project is noted here and in the comments of the Python notebook. So, the datasets created throughout the project:

- **data1** = original data - 135 columns*181.691 rows
- **data2** = data after the columns with over-missing values are eliminated (%60 with valid data preserved)- 58 columns*181.691 rows
- **data3** = data after we take and rename the columns, we see relevant with our study - 32 columns*181.691 rows
- **data4** = data after we take only the attacks that were of terrorist nature - 32 columns*138.879 rows
- **data5** = data after we treated missings and outliers and add new variables / pre-processed final data- 30 columns*138.879 rows

Ultimately, descriptive and predictive analysis are performed on this final data set which is **data5.**

## IV. PROJECT IN DETAIL

### a. READ THE DATA

Because the data was not extracted via web scraping for this Project, the data is downloaded from the relevant website in CSV form. So, simply the "pd.read_csv" is used to read the data in Python notebook. Even though web scraping has not been performed, the data was in a form that needed data cleaning (pre-processing.

### b. CLEAN THE DATA / PRE-PROCESSING

As part of the pre-processing phase, several operations have been applied on the data. The original data is read from the CSV file with the name of **data1**.

First, missing values are checked and it is seen that there are some variables that have more missing than the acceptable number of missings for this dataset. To eliminate those, 60% threshold is set so that the variables that 60% or more valid data are kept in the dataset. So, from 135 columns, now it has decreased to 58 columns. The resulting dataset from here is named as **data2**.

Second, out of these remained variables, the variables which are thought that could be meaningful for our analysis are selected and renamed to ease the analysis. After the selection and renaming, dataset columns decreased from 58 to 32. The resulting dataset from here is named as **data3**.

Third, because there is also other types of attacks in the dataset, to avoid confusion, we restrict the dataset to only attacks that were of terrorist nature. We do that by filtering the data with the variables of 'Criteria_1', 'Criteria_2', 'Criteria_3' and 'Doubts' which are the binary variables that are renamed during the Project and their explanations are as follows:

- **Criteria_1:** was the attack aimed at attaining a political, economic, religious, or social goal?
  - **Set to 1**
- **Criteria_2:** was there intent to coerce or intimidate a larger audience than the victims?
  - **Set to 1**
- **Criteria_3 :** was the incident outside legitimate warfare activities (i.e. target non-combattants)?
  - **Set to 1**
- **Doubts:** was there doubt as to whether or not the incident is a terrorist attack?
  - **Set to 0**

By setting 'Criteria_1', 'Criteria_2', 'Criteria_3' as 1 and 'Doubts' as 0, we make gain a dataset that is composed of only attacks that were of terrorist nature. By that way the dataset decreased from 181.691 rows to 138.879 with again 32 columns. The resulting dataset from here is named as **data4**. And indexes are reset in that point after these changes.

Fourth, unique values of some columns are analyzed to be sure about their values in case of there could be illogical values, out of range (outlier) values etc. As a result, there emerged some invalid values in some columns that need pre-processing. For example, in Weapon_Type column, there is one category whose name is too long so it is shortened from 'Vehicle (not to include vehicle-borne explosives, i.e., car or truck bombs)' to 'Vehicle'. Missings of categorical values are filled with 'Unknown' value. Missings of numerical values are filled with medians. Missing values and 'Unknown' string values in binary variables are replaced with '0'. In addition to these missing and outlier treatments, two new features to be used in our analysis and model are added which are (1) total number of casualties named as 'Casualties' and (2) its binary variable (1-yes, 0-no) named as 'Check_Casualties'. Then, in order not to have a problem in future analysis, the textual variables are changed to lowercase to equalize. Finally, relevant variables are selected after these operations by not including variables 'Criteria_1', 'Criteria_2', 'Criteria_3', and 'Doubts' anymore because they have already been used as filters and there left no need. Eventually, the dataset is composed of 30 columns and 138.879 rows and the resulting dataset from here is named as **data5**. As the last addition, indexes are reset. And as the last control, at that point of the Project in Python notebook, the dataset is exported to computer as in the form of Excel file to double-check the final dataset manually too.

## c. EXPLORATORY ANALYSIS (EDA)

As mentioned in the Purpose section, this first part is focused on exploratory data analysis (EDA) by asking critical questions, exploring the data in accordance with those initial questions and more and then visualizing the answers.

### *c.1. Get initial insights from the data*

At the beginning of the analysis, first initial insights are tried to be get from the data and the conclusions are as follows:

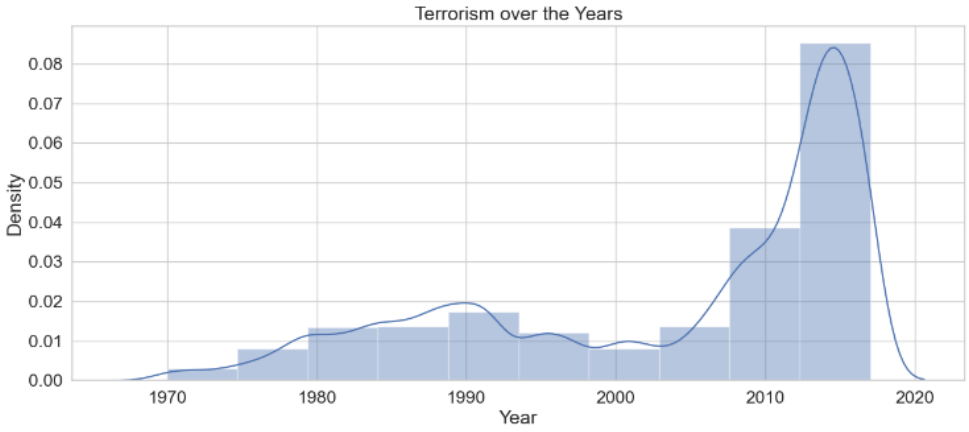
- The region of **Middle East & North Africa** had the highest amount of terrorist attacks totaled at 39.392.
- The most known country with terror attacks was **Iraq** totaled at 21.695.
- **Unknown and Unnamed Cities** consisting of terrorist attacks totaled at 7.221.
- The most known city that had terror attacks was **Baghdad**.
- The most used weapon in terror attacks was **explosives** totaled at 77.361.
- Out of 138.879 total attacks **3.9%** were **suicide attacks.**
- The most preferred method of attack was **bombing/explosion** totaling at 74.073.
- The main targets of terrorists were **private citizens & property** totaling at 26.9%, while the second was **police** at 15.9%.
- **Maximum number of people killed in an attack is 1.384** that took place in United States.
- **Maximum casualties of 9.574** happened in a single attack in United States.
- **Top-5 terrorist groups** by count are;

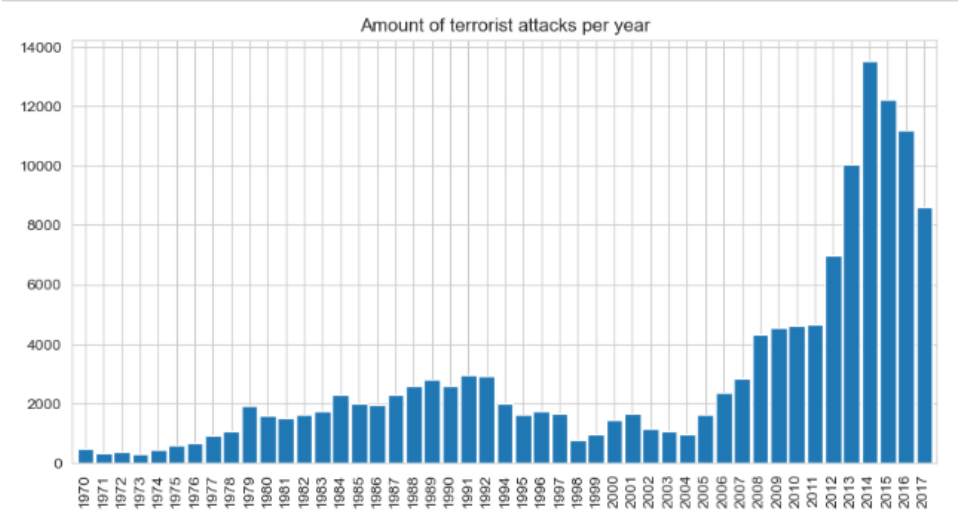
- **(1) Taliban** - 6.314
- **(2) Islamic State of Iraq and The Levant (ISIL)** - 4.409
- **(3) Shining Path (SL)** - 4.138
- **(4) Boko Haram** - 2.166
- **(5) Farabundo Marti National Liberation Front (FMLN)** - 2.129
- Also, there are **61.663 attacks that are unknown attacks** and there is no data on which group did them.

### *c.2. Visualize the data and get deeper insights*

By visualizing the data, the trends and patterns are tried to be discovered. In addition to that, the initial insights taken via first basic inquiries are tried to be improved and detailed here. Besides, other meaningful deductions and inductions are tried to be explored which also leads us to compare this dataset with freedom scores too. After each visualization, conclusions are given.

- **How has the number of terrorist acts changed over time?**
- **Has the number of attacks increased during the recent years?**

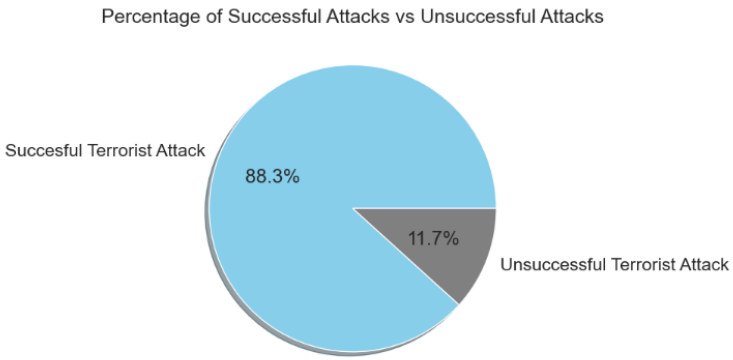



By looking at the graph, one might come to the shocking conclusion that the amount of terrorist attacks has been drastically increasing during the last five years. However, it is important to take into account the effectiveness of data collection since 2012 according to the literature. This implies that the uncertainty in data collection may or may not be responsible for the increase in attacks. So, probably some portion of this observable increase (not all of course) in terrorist activity since 2012 is the result of new advancements in collection methodology. Concluding, results from data analysis should be considered with care.

In that regard, we can also analyze these based on their success-failure rates.

- **What is the percentage of successful attacks in total?**

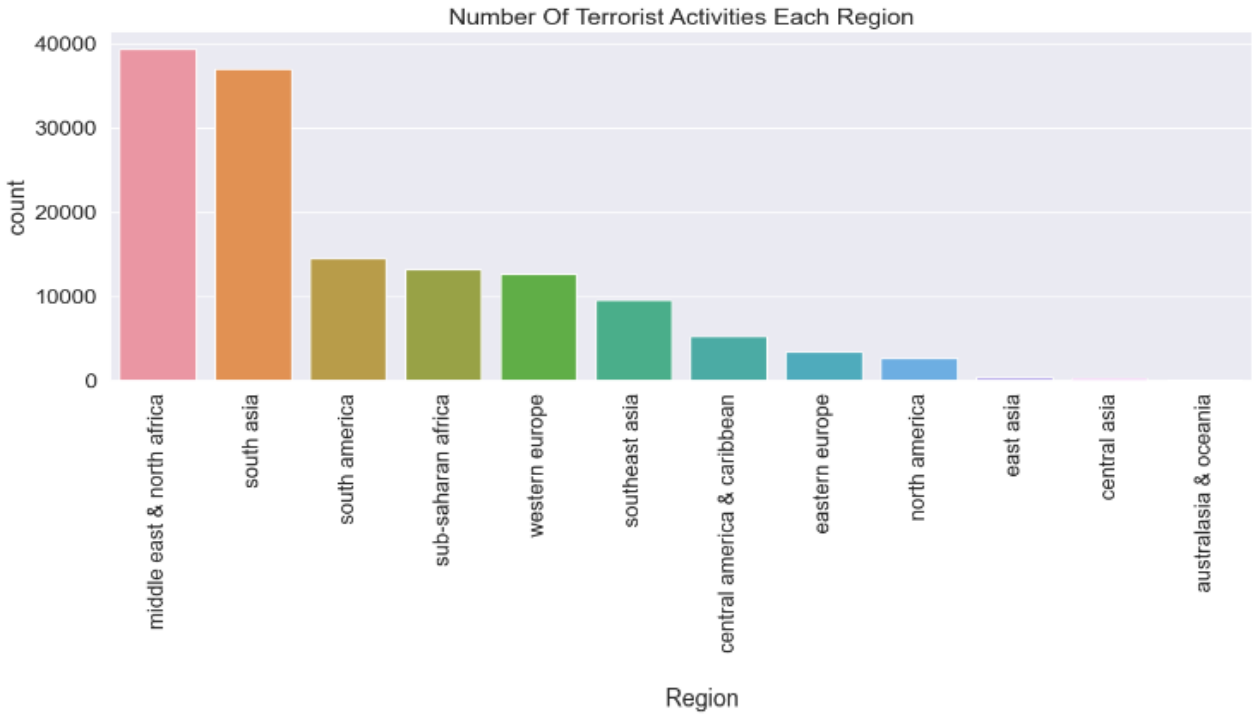


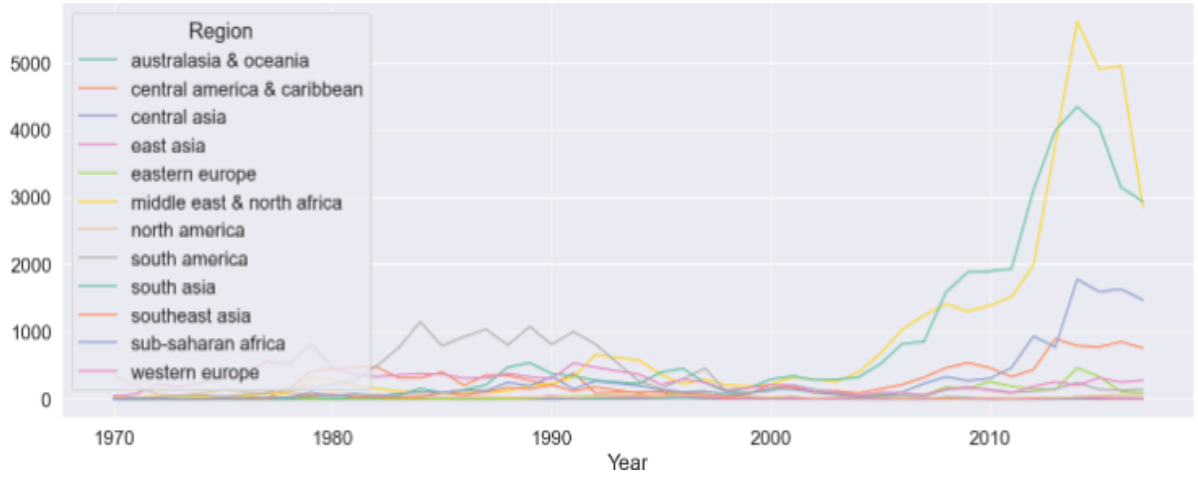

As we can see 88.3 % of attacks were successful over 50 years so only 11.7 % of the attacks were unsuccessful. So, unfortunately the overwhelming majority of attacks end successfully. Now we can analyze this success rate based on over the years

- **What is the proportion of successful/unsuccessful terrorist attacks over the years?**


Immediately noticeable is the drop in both successful and failed attacks in 1998 and as we mentioned before, the increase after 2013. This is a phenomenon shared by all regions. During the last 5 years, there is no clear increase in attacks (so after 2013). We should also look at this with a regional perspective.

- **Where do the terrorist attacks take place?**
- **How the number of terrorist acts vary around the world?**



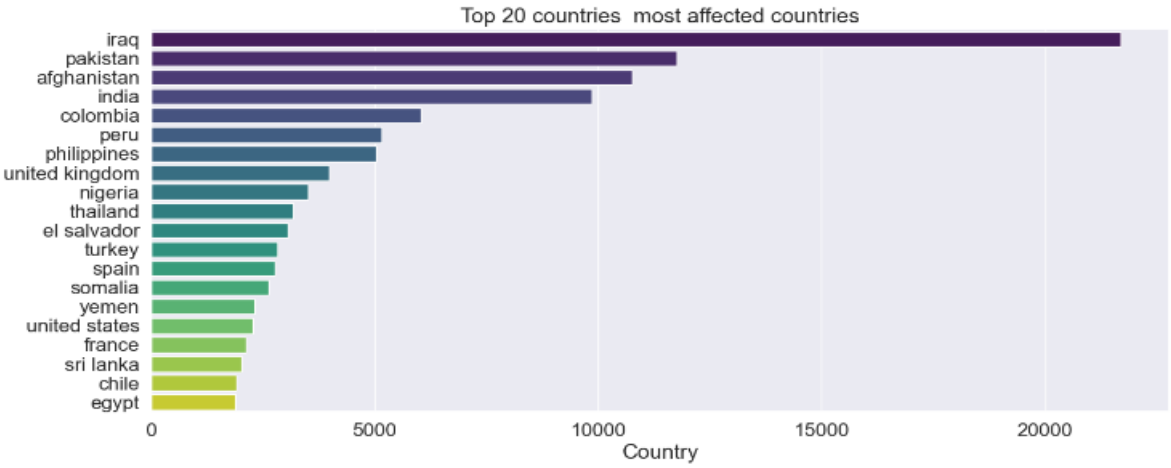




When we plot the number of terrorist activities in each region, we see that the regions that are affected by terrorism the most are as follows:

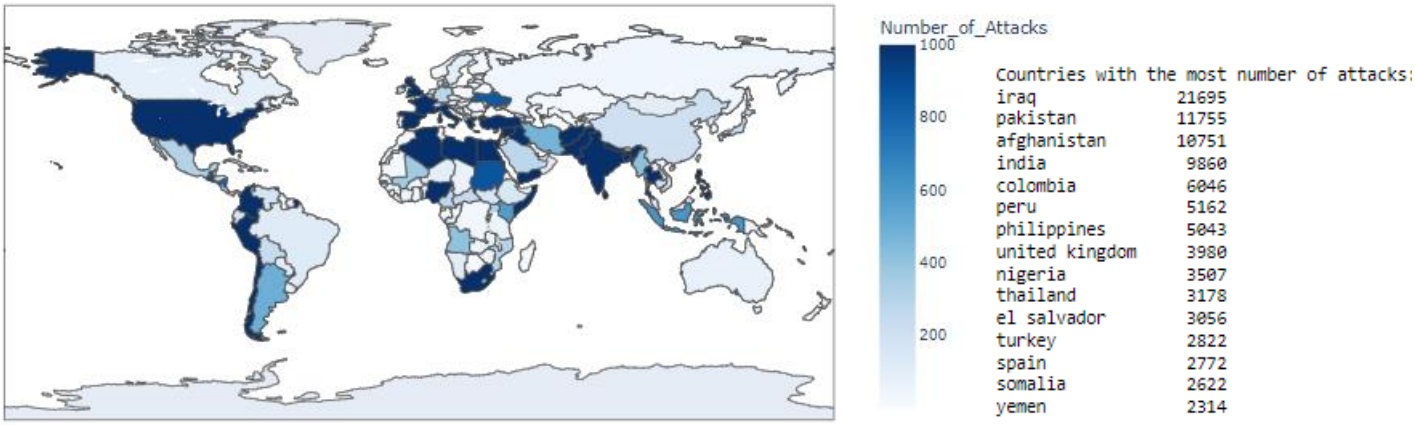
(1) Middle East & North Africa
(2) South Asia
(3) South America
(4) Sub-Saharan Africa
(5) Western Europe
(6) Southeast Asia

As we can see, there is an increase in Middle East & North Africa and South Asia during the latest years. However, in regions like Southern Asia, Easter Europe there is no decline actually, rather their rate is low and constant. Also, it is interesting to note that between 1970 and 1980 Europe was the region with the most acts

of terrorism, followed by North and South America. After 2000s, Asia, Middle East and Africa do have, by far, the most acts on terrorism. Australasia and Oceania have the lowest number of terrorist attacks.




Number of Terrorist Attacks by Country



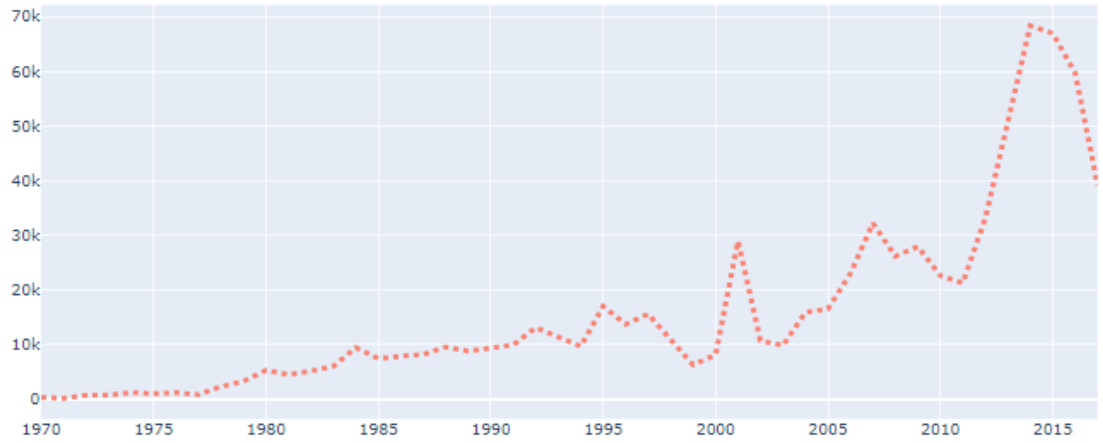

We can see the most affected countries from the above figures. The most known country with terror attacks is Iraq totaled at 21.695 followed by Pakistan and Afghanistan. As we can see and understand, terrorism tends to be very **geographically-focused; most of the attacks were occurred in** the **Middle East, Africa or South Asia.** Then, what about the casualties?

- **Where is most of the casualties?**
- **How have casualties evolved throughout the years?**

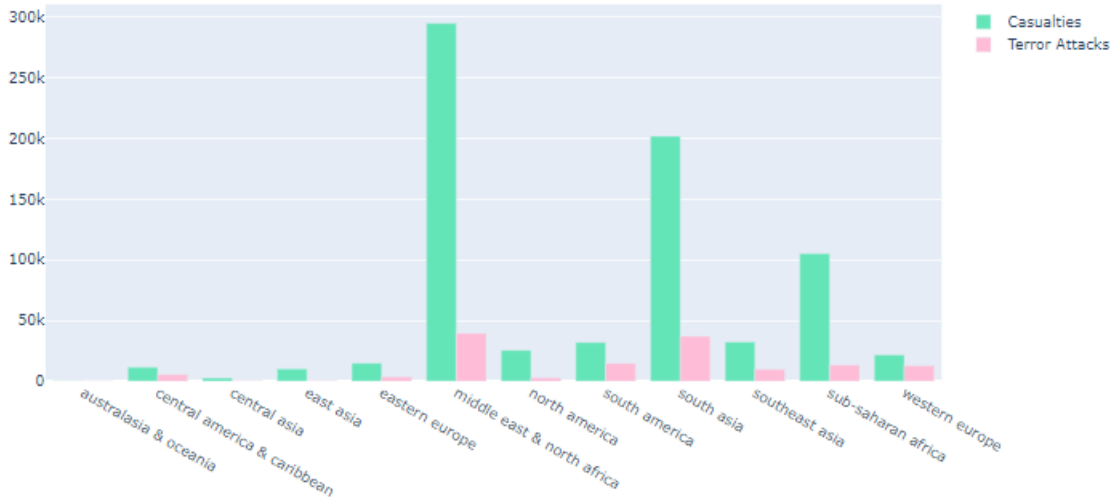
So far, we analyzed the number of attacks based on years and regions/countries/cities, successful/unsuccessful attacks. Now it would be enlightening to analyze the number of casualties based on regions and years, and then target groups, attack/weapons type.

As it is stated in the section of Clean the Data/ Pre-processing, 'Casualties' variable is created for the Project by summing up the number of killed and wounded people.

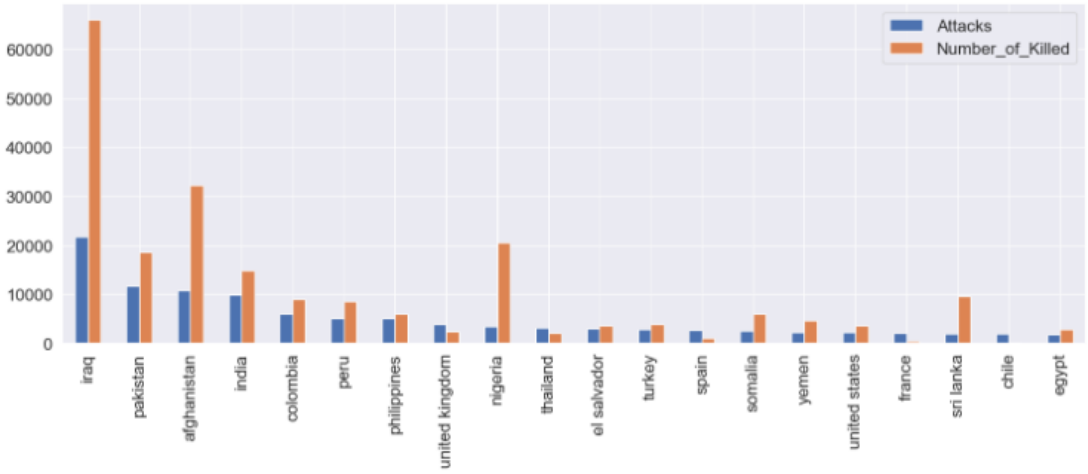
Casualties per Year




Total Casualties and Terror Attacks by Region




| | Attacks | Number_of_Killed |
|---|---|---|
| iraq | 21695 | 66001 |
| pakistan | 11755 | 18804 |
| afghanistan | 10751 | 32325 |
| india | 9860 | 14840 |
| colombia | 6046 | 9035 |
| peru | 5162 | 8548 |
| philippines | 5043 | 5952 |
| united kingdom | 3980 | 2437 |
| nigeria | 3507 | 20603 |
| thailand | 3178 | 2112 |
| el salvador | 3056 | 3674 |
| turkey | 2822 | 3889 |
| spain | 2772 | 1127 |
| somalia | 2622 | 6083 |
| yemen | 2314 | 4679 |
| united states | 2275 | 3641 |
| france | 2113 | 467 |
| sri lanka | 2018 | 9665 |
| chile | 1902 | 192 |
| egypt | 1881 | 2835 |



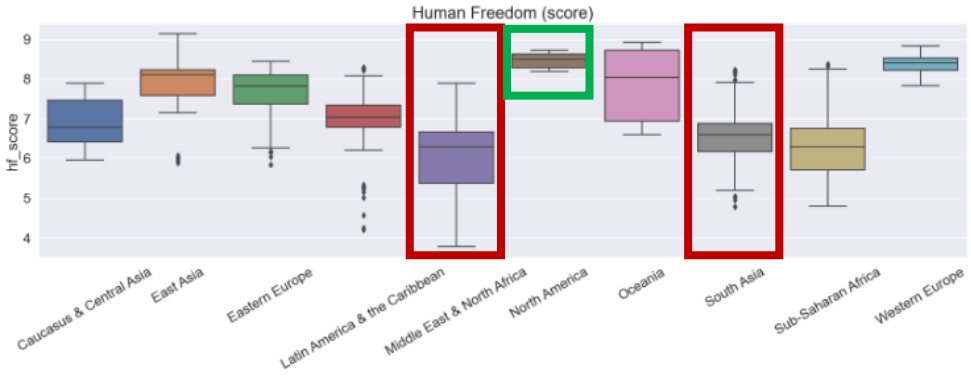
From those figures, we can come to some obvious conclusions. Actually, the number of attacks is lesser than the number of casualties in Middle Eastern countries especially compared to others. High density of population may be the reason. Poor prevention and security may also be a reason for this. Thus, they eventually claim many lives. For example, in Iraq, number of casualties is 191.554, while number of attacks is 21.695. In Ireland, number of casualties is 121, while number of attacks is 269.

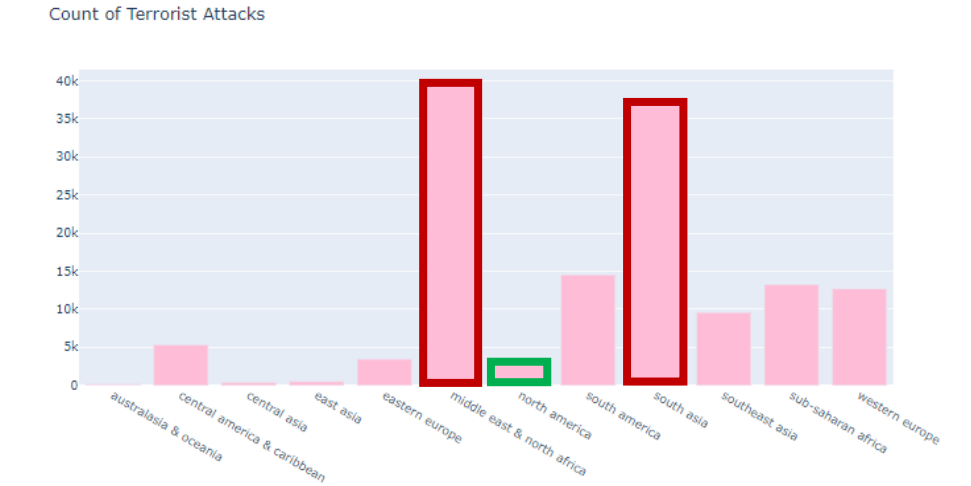
In developed countries like UK, Spain, France, no of attacks is more than the number of casualties. This means that these countries are better at safety, and they are good in prevention before a terror attack happens. Low population density may also be a reason for this.

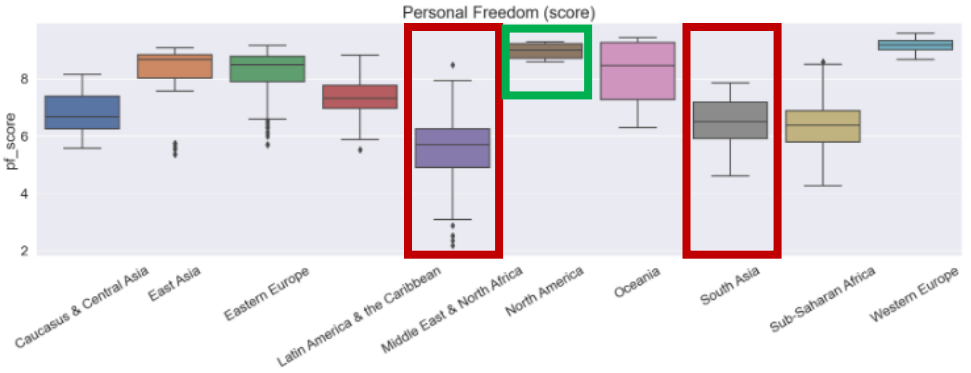
Indeed, this trend one more time proves what we said previously: **"Terrorism tends to be very geographically-focused".** While 95% of deaths in 2017 occurred in the Middle East, Africa or South Asia, in most countries terrorism accounts for less than 0.01% of deaths, but in countries of high-conflict, this can be as much as several percent.

Doesn't that sound like this phenomenon is related with freedom?

- **Are certain nationalities more targeted? Is this related with freedom schedules?**



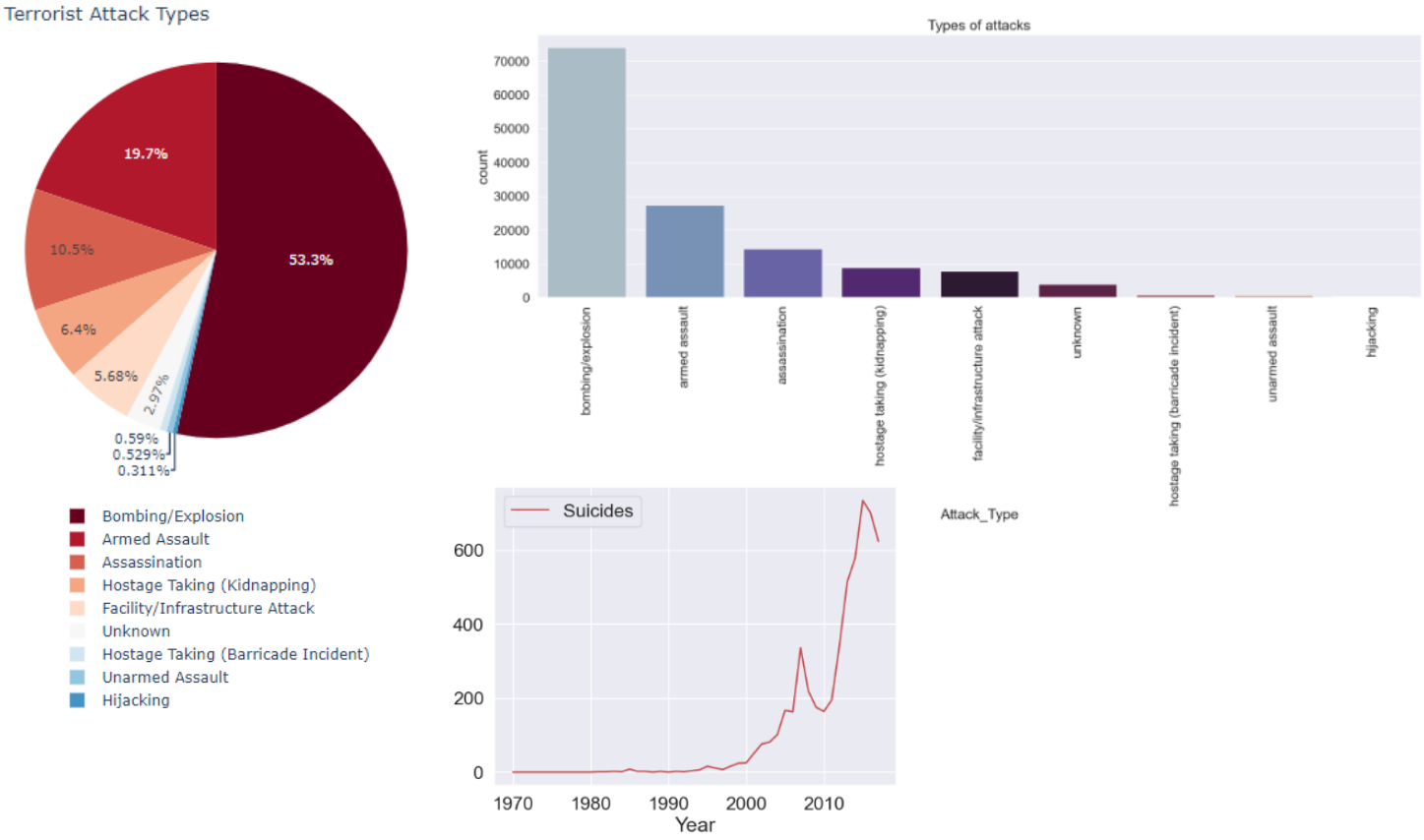



For that analysis, another open-source dataset is used which is **'Human Freedom Index'.** Human Freedom Index Dataset is composed of global measurement of personal, civil, and economic freedoms. Here in our analysis, the figures of human and personal freedom scores are from this dataset, while the number of terrorist attacks figure is from our main global terrorism dataset.

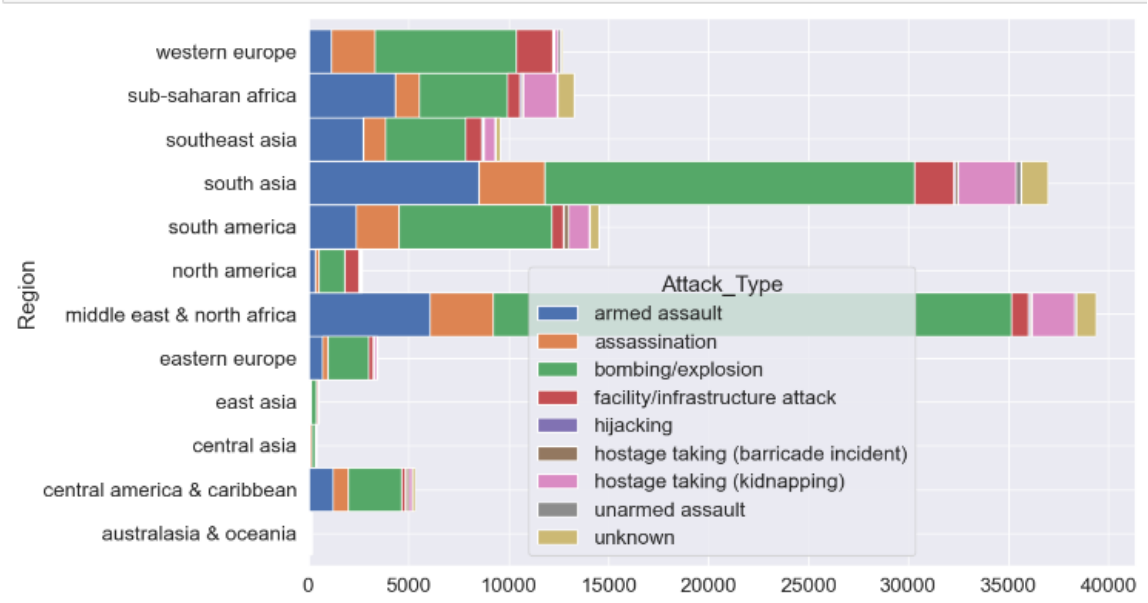
As we can see from these figures, when we compare the number of terrorist attacks based on regions with human freedom and personal freedom scores, we can clearly conclude that "the number of terrorism attacks is related to freedom schedules". For example, while freedom score of Middle East & North Africa is the smallest one, it is the greatest one in the number of terrorist attacks (framed with red). The opposite story is valid for North America for instance. While their freedom score is one of the highest ones, the number of terrorist attacks is very low (framed with green).

- **Which attack types are popular?**



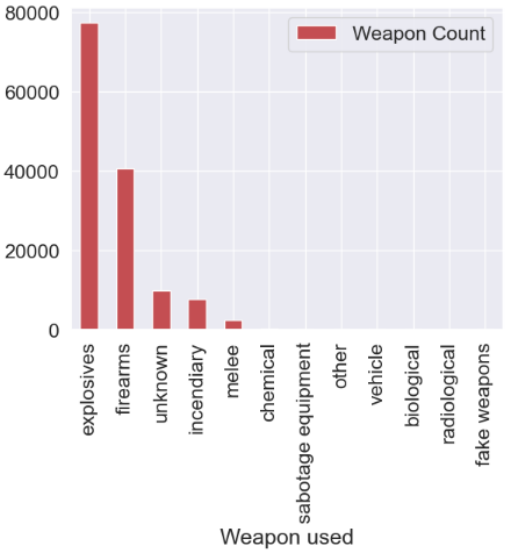

As we can see, the most preferred method of attack was bombing/explosion totaling at 74.073. What is interesting here is that, on contrary to common though, out of 138.879 total attacks just 3.9% were suicide attacks. So, it's highly unlikely to ever find a suicide bomber in a long-drawn-out war. It might pertain to the fact that terrorists are more willing and wanting to live. So, the distribution of suicide bombers is mostly close to zero. This is most likely due to limited manpower and more easier methods for destruction. However, after 2000s, there is an increase even though the magnitude is not high.

Another conclusion here could be places that are in conflict or past conflict tend to have more accessible ways of acquiring things such as explosives. Also, areas that are corrupt, 3rd world or are struggling also tend to influence the masses.




From the chart above, it is clear that bombing and explosion is the favorite attack of terrorist groups in every region. This may be the reason why the greatest number of civilians are killed in the attacks as a single explosion claim lots of lives.

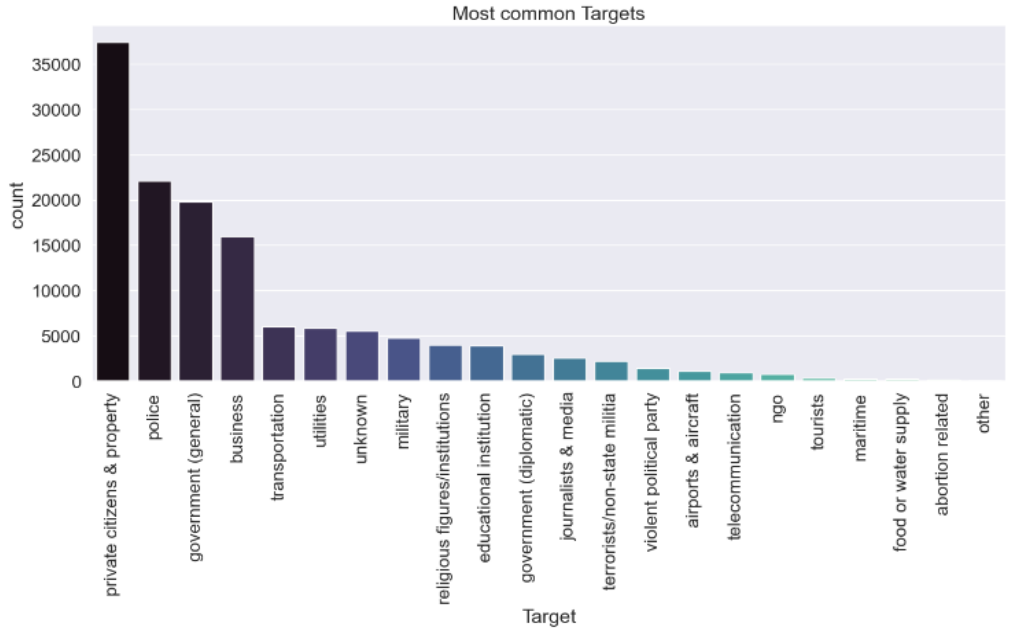
- **Which weapon types are popular?**

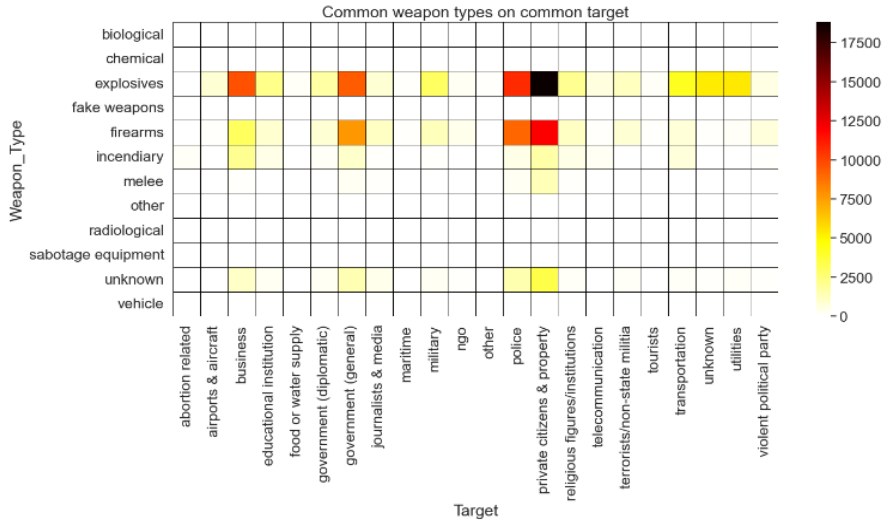


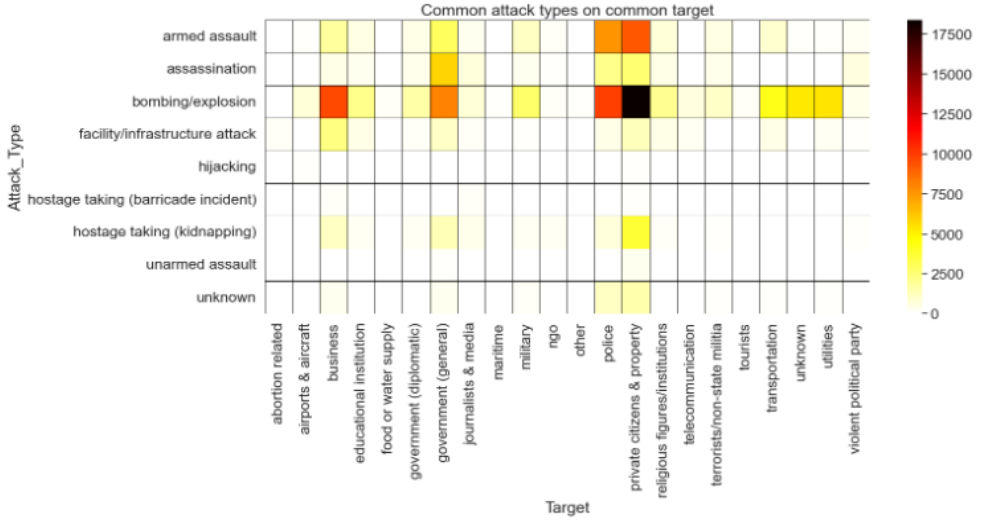

| | Weapon used | Weapon Count |
|---|---|---|
| 0 | explosives | 77361 |
| 1 | firearms | 40636 |
| 2 | unknown | 9895 |
| 3 | incendiary | 7874 |
| 4 | melee | 2481 |
| 5 | chemical | 274 |
| 6 | sabotage equipment | 129 |
| 7 | other | 95 |
| 8 | vehicle | 79 |
| 9 | biological | 30 |
| 10 | radiological | 13 |
| 11 | fake weapons | 12 |

As we can see, just top 5 of these are worth to consider. These are explosives, firearms, incendiary, melee, and unknown weapons. What about the targets?

- **Who are the targets?**
- **What are the targets by attack and weapon type?**



It is obvious that the main targets of terrorists were private citizens & property totaling at 26.9%, while the second was police at 15.9%.

| Target | Number_of_Killed |
|---|---|
| private citizens & property | 129242 |
| police | 50116 |
| government (general) | 24974 |
| business | 21567 |
| military | 17330 |
| transportation | 13312 |
| religious figures/institutions | 12943 |
| terrorists/non-state militia | 6230 |
| airports & aircraft | 3647 |
| unknown | 3585 |
| educational institution | 3456 |
| government (diplomatic) | 2838 |
| violent political party | 1962 |
| utilities | 1830 |
| journalists & media | 1369 |
| maritime | 1148 |
| ngo | 943 |
| tourists | 640 |
| food or water supply | 286 |
| other | 221 |
| telecommunication | 175 |
| abortion related | 9 |

| Target | Number_of_Wounded |
|---|---|
| private citizens & property | 172266 |
| police | 62451 |
| business | 51806 |
| government (general) | 40108 |
| transportation | 39865 |
| religious figures/institutions | 24219 |
| military | 20916 |
| government (diplomatic) | 10107 |
| educational institution | 9411 |
| terrorists/non-state militia | 5909 |
| violent political party | 3411 |
| airports & aircraft | 3353 |
| unknown | 3344 |
| journalists & media | 1676 |
| utilities | 1321 |
| tourists | 1212 |
| maritime | 896 |
| ngo | 745 |
| telecommunication | 478 |
| other | 379 |
| food or water supply | 229 |
| abortion related | 40 |

From the figures, it can be seen that private citizens, police, government, business and military combined have a total share in number of kills of nearly three quarters. However, the share in property damage is more evenly distributed over the different target types. Military has the highest average number of deaths per attacks, but it has a relatively low number of wounded per attack compared to the other target types. Hence, terrorism attacks kill more defense personnel (police and military) than private citizens. Besides, defense organizations suffer more property damage then private citizens. Eventually, when looking at the whole public sector, compared to the private sector, the public sector does suffer more than the private sector.

## d. PREDICTIVE ANALYSIS

### *d.1. The Models*

Predictive analysis section of this Project is composed of two parts.

First, we develop a Random Forest model to check GTD-exclusive features' power which is our main dataset. We ask the question of "Can casualties be predicted from GTD-exclusive features?" and proved it. This kind of a check has been performed before the second part for the purpose of ensuring our dataset on which our second part is developed upon.

Second, we tried to predict the successfulness of terrorism by using Decision Trees and Random Forests and develop the optimum model that predicts whether a terrorist attack is expected to succeed or fail? Such kind of a prediction is valuable especially for intelligence groups to estimate the impact of possible future terrorist attacks and to allocate resources accordingly.

### *d.2. First part: "Can casualties be predicted from GTD-exclusive features?"*

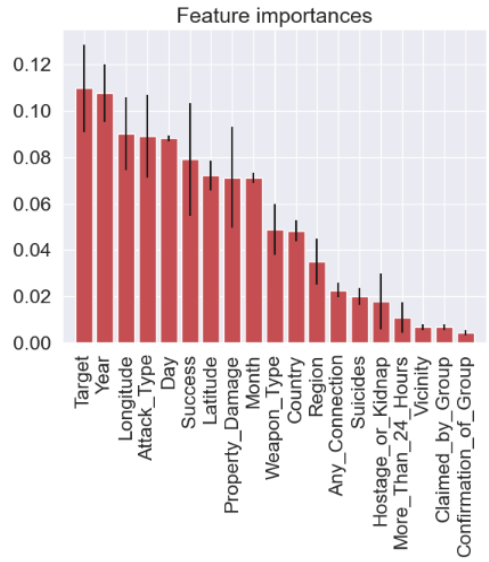
In this part, columns that are exclusively part of the Global Terrorism Database are tested for their significance on predicting casualties. **Our target is a binary value: "Were there any casualties in the attack?"** named as **'Check_Casualties'** in our dataset in the Python notebook (data5).

Relevant GTD-exclusive features are listed as follows: 'Year', 'Month', 'Day', 'Country', 'Region', 'Latitude', 'Longitude', 'More_Than_24_Hours', 'Vicinity','Confirmation_of_Group', 'Claimed_by_Group', 'Any_Connection', 'Property_Damage', 'Hostage_or_Kidnap', 'Success', 'Attack_Type', 'Target', 'Weapon_Type', 'Suicides'

For categorical variables, **LabelEncoder i**s used to prepare them for machine learning algorithm. Data is split train and test with 0.3 train data.

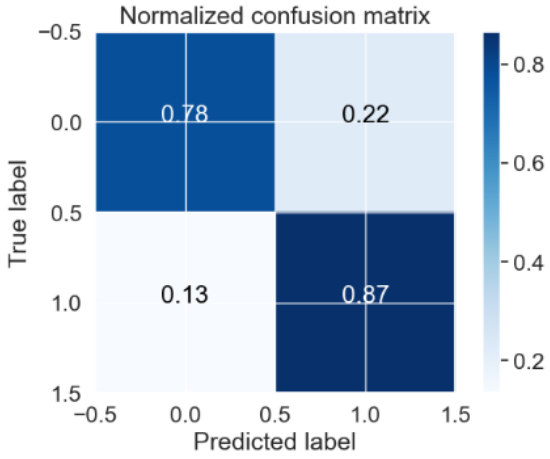
To predict the importance of each column **Random Forest classifier** is used. Random Forest is preferred because the risk of overfit is really small in Random Forest and it has an automatic capacity to fight against overfitting. It fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

This plot at the right-hand-side shows the **significance of each feature of the forest on predicting whether there will be casualties or not.** Spatial-temporal variables seem to be very dominantly present. However, time variables (Year, Month, Day) are not appropriate to keep, because predictions into the future will supply the model with meaningless values (the model never saw year values > 2017 before).




The features are kept based on their scores on accuracy. To avoid overfitting the model, only features with an accuracy score of 0.05 and higher are kept. So, we assign the chosen features to X which are 'Target', 'Longitude', 'Attack_Type', 'Success', 'Latitude', 'Property_Damage', 'Weapon_Type', 'Country'.

Accuracy score of the model has emerged as **0.78 (+/- 0.06).** At the right-hand side the confusion matrix of the model can be seen.




To conclude, because the baseline accuracy is 0.58 based on our calculations and supplied with the chosen features, the Random Forest classifier achieves an accuracy score of 0.78, we answered our question of **"Can casualties be predicted from GTD-exclusive features?"** as yes.

```
# Better validation with K-fold.
model = RandomForestClassifier(n_estimators=20)

scores = cross_val_score(model, X, y, cv=10) # Ten-fold cross validation.
print(scores)
print('Accuracy: %0.2f (+/- %0.2f)' % (scores.mean(), scores.std() * 2))
```

```
[0.73 0.75 0.74 0.8  0.77 0.78 0.8  0.77 0.82 0.82]
Accuracy: 0.78 (+/- 0.06)
```

```
# Simply predict the most frequent value every time.
# This determines the baseline accuracy.
model = DummyClassifier(strategy="most_frequent")

%time model.fit(X_train, y_train)

%time y_pred = model.predict(X_test)

np.mean(y_pred == y_test)
```

```
Wall time: 4 ms
Wall time: 0 ns

0.586093509984639
```

### *d.3. Second part: "Whether a terrorist attack is expected to succeed or fail?"*

In this part, we tried to predict successfulness of terrorism by using Decision Trees and Random Forests and come up with a model that predicts whether a terrorist attack is expected to succeed or fail? Such kind of a prediction is valuable because it can be used by intelligence groups to predict the impact of possible future terrorist attacks, and thus make them able to determine the risky situations in advance and allocate the necessary resources better.

For that part of the Project, we did not continue with the former pre-processed dataset for EDA part (data5), rather pre-process a new version of GTD for our new purpose. As we know now well, one of the binomial variables of GTD is whether an attack has been successful or not (success variable).

Actually, success of a terrorist strike is defined according to the tangible effects of the attack. Success is not judged in terms of the larger goals of the perpetrators. For example, a bomb that exploded in a building would be counted as a success even if it did not succeed in bringing the building down or inducing government repression. The definition of a successful attack depends on the type of attack. Essentially, the key question is whether or not the attack type took place. If a case has multiple attack types, it is successful if any of the attack types are successful, with the exception of assassinations, which are only successful if the intended target is killed.

- 1 = "Yes" The incident was successful.
- 0 = "No" The incident was not successful.

So, this analysis aims to find a model, using the available variables, for predicting the successfulness of terrorist attacks. The methods used are Decision Trees and Random Forests.

As the first step, we start by downloading the data, shaping/cleaning it, visualizing it and choosing the variables. There are many reasons why only about half of the available variables are chosen. One reason, for example, is that there are many string variables such as name of the city or the perpetrator group name (70+ entries) which is something making them difficult to categorize. Another reason, for example, there were too many missing values for some variables and replacing them artificially would be distorting for the results, so they were left out. For instance, 'number of perpetrators' variable was with too many unknowns. So, we get rid of some of the variables such as claimed, nkillter, nwound, nwoundte, because majority of their values were missing and replacing them artificially would be misleading. However, after all these, there were still NaN's in some of the variables. So, we use the means of the available values to replace them. Finally, there were still 1.559 entries missing from nationalities (natlty1_text) but since it's a string variable and will not be used in calculations we decided to get rid of it too. So, the variables after pre-processing and cleaning are as follows:

**# Number (sum) of variables' missing values**

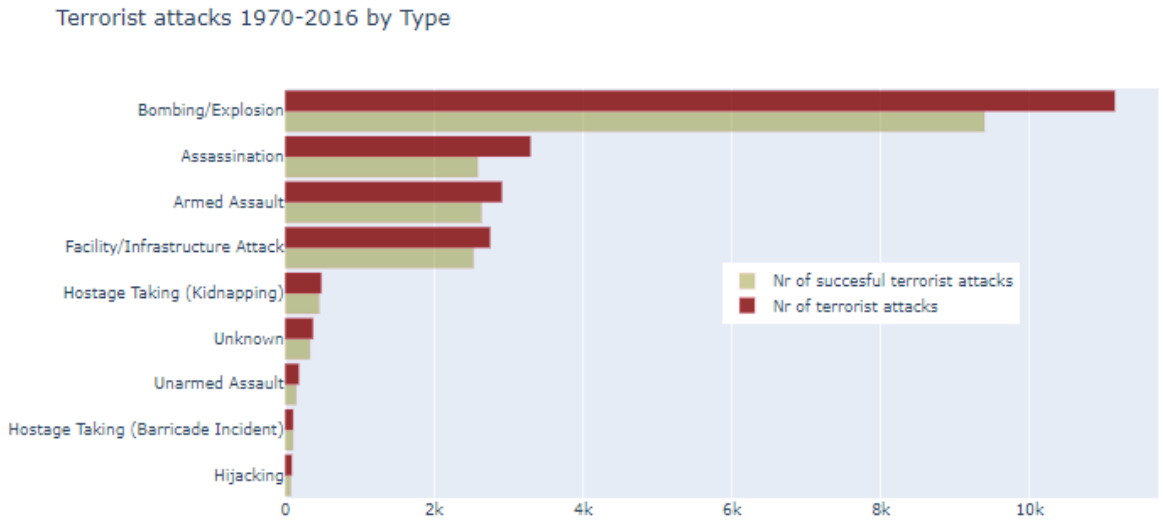
```
iyear               0
imonth              0
iday                0
extended            0
country             0
country_txt         0
region              0
latitude            0
longitude           0
multiple            0
success             0
suicide             0
attacktype1         0
attacktype1_txt     0
targtype1           0
targtype1_txt       0
natlty1             0
individual          0
weaptype1           0
weaptype1_txt       0
nkill               0
```

**# Descriptive statistics of variables**

| | iyear | imonth | iday | extended | country | latitude | longitude | multiple |
|---|---|---|---|---|---|---|---|---|
| count | 181691.000000 | 181691.000000 | 181691.000000 | 181691.000000 | 181691.000000 | 181691.000000 | 1.816910e+05 | 181691.000000 |
| mean | 2002.638997 | 6.467277 | 15.505644 | 0.045346 | 131.968501 | 23.691867 | -4.461515e+02 | 0.137772 |
| std | 13.259430 | 3.388303 | 8.814045 | 0.208063 | 112.414535 | 18.382842 | 2.021946e+05 | 0.344862 |
| min | 1970.000000 | 0.000000 | 0.000000 | 0.000000 | 4.000000 | -53.000000 | -8.618590e+07 | 0.000000 |
| 25% | 1991.000000 | 4.000000 | 8.000000 | 0.000000 | 78.000000 | 12.000000 | 7.000000e+00 | 0.000000 |
| 50% | 2009.000000 | 6.000000 | 15.000000 | 0.000000 | 98.000000 | 31.000000 | 4.300000e+01 | 0.000000 |
| 75% | 2014.000000 | 9.000000 | 23.000000 | 0.000000 | 160.000000 | 35.000000 | 6.800000e+01 | 0.000000 |
| max | 2017.000000 | 12.000000 | 31.000000 | 1.000000 | 1004.000000 | 75.000000 | 1.790000e+02 | 1.000000 |

| success | suicide | attacktype1 | targtype1 | natlty1 | individual | weaptype1 | nkill |
|---|---|---|---|---|---|---|---|
| 181691.000000 | 181691.000000 | 181691.000000 | 181691.000000 | 181691.000000 | 181691.000000 | 181691.000000 | 181691.000000 |
| 0.889598 | 0.036507 | 3.247547 | 8.439719 | 127.457458 | 0.002950 | 6.447325 | 2.266860 |
| 0.313391 | 0.187549 | 1.915772 | 6.653838 | 88.949238 | 0.054234 | 2.173435 | 11.227057 |
| 0.000000 | 0.000000 | 1.000000 | 1.000000 | 4.000000 | 0.000000 | 1.000000 | 0.000000 |
| 1.000000 | 0.000000 | 2.000000 | 3.000000 | 83.000000 | 0.000000 | 5.000000 | 0.000000 |
| 1.000000 | 0.000000 | 3.000000 | 4.000000 | 101.000000 | 0.000000 | 6.000000 | 0.000000 |
| 1.000000 | 0.000000 | 3.000000 | 14.000000 | 168.000000 | 0.000000 | 6.000000 | 2.000000 |
| 1.000000 | 1.000000 | 9.000000 | 22.000000 | 1004.000000 | 1.000000 | 13.000000 | 1570.000000 |

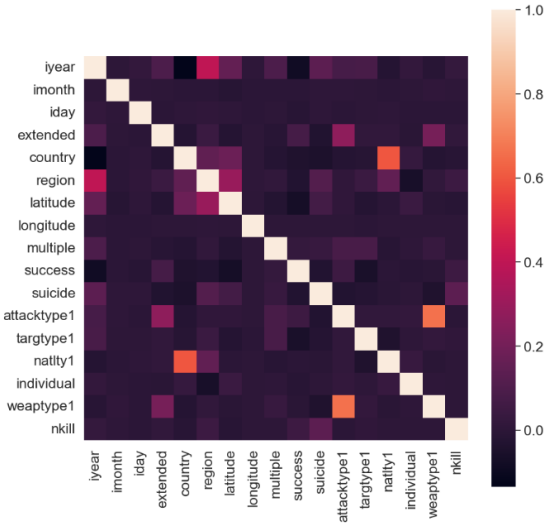
From the summary of the numerical attributes, it can be seen that the mean for success is **0.889598** which is a high success rate. This is also visible in the below figure.




Next, we have a look at the variables' correlations, numerically and graphically.

**# Variables' Correlations**

| | iyear | imonth | iday | extended | country | region | latitude | longitude | multiple | success | suicide | attacktype1 | targtype1 | natlty1 | individual | weaptype1 | nkill |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **iyear** | 1.000000 | 0.000139 | 0.018254 | 0.091754 | -0.135023 | 0.401384 | 0.154389 | 0.003803 | 0.093734 | -0.082963 | 0.137736 | 0.074153 | 0.079082 | -0.019855 | 0.017944 | -0.011737 | 0.021252 |
| **imonth** | 0.000139 | 1.000000 | 0.005497 | -0.000468 | -0.006305 | -0.002999 | -0.015925 | -0.003832 | -0.004420 | -0.002845 | 0.003071 | 0.006705 | -0.000948 | -0.004783 | -0.001661 | 0.007863 | 0.004031 |
| **iday** | 0.018254 | 0.005497 | 1.000000 | -0.004700 | 0.003468 | 0.009710 | 0.002760 | -0.002258 | -0.000394 | -0.011802 | 0.003593 | -0.005333 | -0.000052 | 0.003724 | -0.003788 | -0.003019 | -0.003225 |
| **extended** | 0.091754 | -0.000468 | -0.004700 | 1.000000 | -0.020466 | 0.038389 | -0.022920 | 0.000521 | -0.013440 | 0.073233 | -0.033981 | 0.272272 | 0.011613 | 0.013918 | -0.008929 | 0.207802 | 0.014586 |
| **country** | -0.135023 | -0.006305 | 0.003468 | -0.020466 | 1.000000 | 0.148597 | 0.177108 | -0.000276 | -0.022220 | -0.037827 | -0.050360 | -0.021384 | -0.019703 | 0.598196 | 0.027918 | -0.023708 | -0.014383 |
| **region** | 0.401384 | -0.002999 | 0.009710 | 0.038389 | 0.148597 | 1.000000 | 0.296557 | 0.004239 | 0.011968 | -0.030909 | 0.112672 | 0.007642 | 0.041396 | 0.149508 | -0.059963 | 0.013257 | 0.043113 |
| **latitude** | 0.154389 | -0.015925 | 0.002760 | -0.022920 | 0.177108 | 0.296557 | 1.000000 | 0.001441 | -0.024466 | -0.071192 | 0.066479 | 0.010904 | -0.021161 | -0.003643 | 0.040151 | -0.008530 | -0.011785 |
| **longitude** | 0.003803 | -0.003832 | -0.002258 | 0.000521 | -0.000276 | 0.004239 | 0.001441 | 1.000000 | 0.000937 | -0.000840 | 0.000475 | 0.001532 | -0.003362 | -0.000444 | 0.000111 | 0.001558 | -0.000569 |
| **multiple** | 0.093734 | -0.004420 | -0.000394 | -0.013440 | -0.022220 | 0.011968 | -0.024466 | 0.000937 | 1.000000 | 0.020310 | 0.030751 | 0.081875 | 0.079645 | -0.013450 | 0.004756 | 0.032538 | -0.001068 |
| **success** | -0.082963 | -0.002845 | -0.011802 | 0.073233 | -0.037827 | -0.030909 | -0.071192 | -0.000840 | 0.020310 | 1.000000 | -0.031155 | 0.048408 | -0.059637 | -0.004888 | -0.013868 | -0.008178 | 0.049619 |
| **suicide** | 0.137736 | 0.003071 | 0.003593 | -0.033981 | -0.050360 | 0.112672 | 0.066479 | 0.000475 | 0.030751 | -0.031155 | 1.000000 | -0.029962 | -0.023440 | -0.009709 | 0.000775 | -0.039928 | 0.136385 |
| **attacktype1** | 0.074153 | 0.006705 | -0.005333 | 0.272272 | -0.021384 | 0.007642 | 0.010904 | 0.001532 | 0.081875 | 0.048408 | -0.029962 | 1.000000 | 0.014513 | 0.013605 | 0.016438 | 0.658954 | -0.003448 |
| **targtype1** | 0.079082 | -0.000948 | -0.000052 | 0.011613 | -0.019703 | 0.041396 | -0.021161 | -0.003362 | 0.079645 | -0.059637 | -0.023440 | 0.014513 | 1.000000 | -0.037783 | 0.005267 | 0.019846 | 0.008466 |
| **natlty1** | -0.019855 | -0.004783 | 0.003724 | 0.013918 | 0.598196 | 0.149508 | -0.003643 | -0.000444 | -0.013450 | -0.004888 | -0.009709 | 0.013605 | -0.037783 | 1.000000 | 0.030541 | -0.008848 | 0.001313 |
| **individual** | 0.017944 | -0.001661 | -0.003788 | -0.008929 | 0.027918 | -0.059963 | 0.040151 | 0.000111 | 0.004756 | -0.013868 | 0.000775 | 0.016438 | 0.005267 | 0.030541 | 1.000000 | 0.005754 | -0.001275 |
| **weaptype1** | -0.011737 | 0.007863 | -0.003019 | 0.207802 | -0.023708 | 0.013257 | -0.008530 | 0.001558 | 0.032538 | -0.008178 | -0.039928 | 0.658954 | 0.019846 | -0.008848 | 0.005754 | 1.000000 | -0.001528 |
| **nkill** | 0.021252 | 0.004031 | -0.003225 | 0.014586 | -0.014383 | 0.043113 | -0.011785 | -0.000569 | -0.001068 | 0.049619 | 0.136385 | -0.003448 | 0.008466 | 0.001313 | -0.001275 | -0.001528 | 1.000000 |




As we can see from the left-hand side figure and above table, there are some obvious higher correlations, like between nationality & country and attacktype & weapontype. Other variables are fairly non-correlated, which looks good for the model; we don't have closely related variables basically just capturing the same thing.

As the next step, we create train and test datasets. We did that by splitting our data to a train set with the portion of 80% and test set with the portion of 20%. In that way, we try to predict the variable of success; "What determines whether a terrorist attack will be successful or not?"

The random_state variable of the split is set to a fixed number, randomly '42', thereby keeping the random number generator constant. This way we will always be getting the same split and avoid the risk of introducing sampling bias.

Then, we need to separate the features from the target because we're trying to build a Decision Tree (and later a Random Forest). We skip variable 'iyear', since past years cannot reoccur and have therefore no impact on the prediction. Also, we skip variable 'country', since it must fit together with the variables 'longitude' and 'latitude' anyway and does not bring additional value on its own.

We use a Confusion Matrix first on a Decision Tree then on a Random Forest to evaluate the accuracy of each method.

The **Precision**(=accuracy of the positive predictions), **Recall**(=ratio of positive instances correctly detected by the classifier) and **f1-score** may be more concise metrics, however.

- **Precision** for 'success' = TP/(TP+FP)
- **Precision** for *not* 'success' = TN/(TN+FN)

- **Recall** for 'success' = TP/(TP+FN)
- **Recall** for *not* 'success' = TN/(TN+FP)

- The **f1-score** is the harmonic mean of Precision and Recall.

**# Decision Tree with max node depth of 3**

We built two Decision Trees models: a simple (with max node depth of 3) and a more complex one (no max node depth). After, checking our data, we construct the actual Decision Tree. Just to make the first try more visual, we stick to a max depth of 3 for the decision nodes.

Hence, for the first one, we define a decision tree classifier model using the scikit-learn library, with a maximum depth of 3, and fitting it to the training data (X_train, y_train). The variable 'y' is assigned the column of the dataframe 'df' labeled 'success', and the variable 'X' is assigned a subset of the dataframe 'df' containing the columns specified in the 'features' list. Once the model is trained, it can be used to make predictions on new data. Then, we come to that part. So, we used this trained decision tree classifier (dtc) to make predictions on the test data (X_test) and saving the predictions in the variable 'dtc_pred'. The classification_report and confusion_matrix functions are used to evaluate the performance of the model on the test data by comparing the predicted values (dtc_pred) to the true test labels (y_test). The classification_report function returns various evaluation metrics such as precision, recall, f1-score, and support for each class. The confusion_matrix function returns a matrix that compares the predicted values to the true values which will be used to understand the number of correct and incorrect predictions made by the model. Here are the results for our first decision tree with max node depth of 3:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.85 | 0.20 | 0.32 | 3978 |
| 1 | 0.91 | 1.00 | 0.95 | 32361 |
| accuracy | | | 0.91 | 36339 |
| macro avg | 0.88 | 0.60 | 0.63 | 36339 |
| weighted avg | 0.90 | 0.91 | 0.88 | 36339 |

TN FP

```
[[  778  3200]
 [  132 32229]]
```

FN TP

The confusion matrix compares the predicted values (dtc_pred) to the true test labels (y_test). The matrix has the count of True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN). With these values, we become able to calculate accuracy, precision, recall, F1 score, etc.

**# Decision Tree with no max node depth**

For the second one, we can say that its code is very similar to the first one but with one difference. The maximum depth of the decision tree is not specified in this case. When the maximum depth of a decision tree

is not specified, the tree can continue to grow until all the leaves contain pure samples or until all leaves contain less than the minimum number of samples. This can lead to overfitting, which means that the decision tree is too complex and performs well on the training data but poorly on new unseen data.

Then, the classification report and confusion matrix are again used to evaluate the performance of the model on the test data, however, without specifying the maximum depth it may be possible that the model will not generalize well on new unseen data. It will be important to evaluate the model performance with metrics such as precision, recall, f1-score, and support for each class, as well as the count of True Positives, False Positives, True Negatives and False Negatives. Here are the results:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.48 | 0.54 | 0.51 | 3978 |
| 1 | 0.94 | 0.93 | 0.94 | 32361 |
| accuracy | | | 0.89 | 36339 |
| macro avg | 0.71 | 0.74 | 0.72 | 36339 |
| weighted avg | 0.89 | 0.89 | 0.89 | 36339 |

TN FP

```
[[ 2162  1816]
 [ 2320 30041]]
```

FN TP

We see that True Negatives have become almost tree times greater. False Positives have decreased by half almost. False Negatives are much higher now and True Positives are closer to each other.

**# Random Forest**

Since Decision Trees can suffer from overfitting, we continue with Random Forests to see if we can improve the model. Besides that, Random Forest has an automatic capacity to fight against overfitting.

So, we define a random forest classifier model using the scikit-learn library, with the number of trees in the forest equal to 400, and fitting it to the training data (X_train, y_train). The variable 'rfc_pred' is then used to store the predictions made by the model on the test data (X_test).

The classification_report and confusion_matrix functions from the scikit-learn library are then used to evaluate the performance of the model on the test data by comparing the predicted values (rfc_pred) to the true test labels (y_test). The classification_report function returns the evaluation metrics of precision, recall, f1-score, and support for each class while the confusion_matrix function returns a matrix that compares the predicted values to the true values which will be used to understand the number of correct and incorrect predictions made by the model. So the results are as follows:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.79 | 0.49 | 0.60 | 3978 |
| 1 | 0.94 | 0.98 | 0.96 | 32361 |
| accuracy | | | 0.93 | 36339 |
| macro avg | 0.86 | 0.74 | 0.78 | 36339 |
| weighted avg | 0.92 | 0.93 | 0.92 | 36339 |

TN FP

```
[[ 1937  2041]
 [  516 31845]]
```

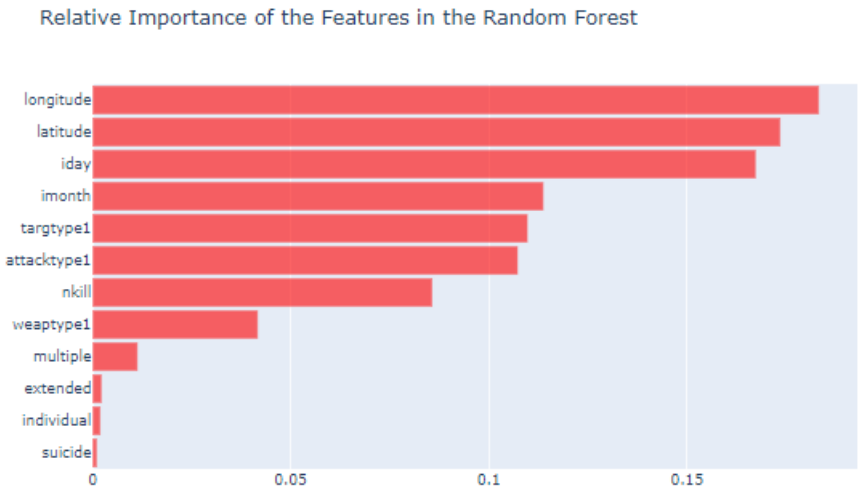
FN TP

Overall, comparing the Confusion Matrix of the latter Decision Tree with the Random Forest, the difference is not very striking, except for the clear reduction in False Negatives. However, It's clear we go with the Random Forest model!

Random Forest Classifier is an ensemble method, which combines the predictions of multiple decision trees. It works by training multiple decision trees on different subsets of the data and then averaging their predictions. This often results in a more robust and accurate model, especially when dealing with high-dimensional datasets.

By the way, after we tried with various n_estimators, i.e. "number of trees in the forest", with 10 being the default, the optimal confusion_matrix was received with (n_estimators=400). Increasing the number to up to 1000 did not improve the outcome, but slightly to the contrary.

As the next step, we check for the relative importance of each attribute for making accurate predictions. With this information, we could drop some of the less useful features, if we decide to fine-tune the model further.




Looks like dropping the features extended, individual and suicide might be considered.

Finally, we implement the model so checked the outcome with inserted data. It is using a trained random forest classifier (succeed_or_fail) to make a prediction about the success or failure of a hypothetical terrorist attack, given certain features of the attack. The features include the month, day, extended (1=yes, 0=no), latitude, longitude, multiple (attack is part of a multiple incident (1), or not (0)), suicide (suicide attack (1) or not (0)), attackType (9 categories), targetType (22 categories), individual (known group/organization (1) or not (0)), weaponType (13 categories) and nkill (number of total casualties from the attack). These features are passed as a list of lists to the predict method of the classifier, where the classifier will make a prediction (1 for successful attack, 0 for unsuccessful attack) and store it in the outcome variable. Now we finally have a model using which we can actually predict whether an attack is expected to succeed or fail. By typing in the twelve variables - here rather randomly chosen - the model gives the predicted outcome.

The outcome is then checked with an if-else statement, and the corresponding message is printed based on the outcome: “The attack based on these features would be succesful”. So we have a model with 93% accuracy which made us able to predict “Whether an terrorist attack is expected to succeed or fail?"

It's worth to note that this outcome should be taken with caution since the accuracy and the generalization of the model is not known and this would depend on the quality of the data and the model performance.

## V. APPENDIX

**a.** Codebook of Global Terrorism Dataset

Codebook.pdf