Ryerson University

Student Id: 500862456

Student Name: Emil Ibrahim

Course: CKM136XJ0 Capstone

Supervisor: Dr. Can Kavaklioglu

GLOBAL TERRORISM DATA ANALYSIS

Project repository:
https://github.com/emilkaram/CKM136XJ0-Global-Terrorism-Data-Analytics-Capstone

Introduction

Terrorism is one of main problems in the 21st century.

Terrorism has continuously been an ongoing danger all over the world.

Analysing global terrorism historical dataset will help to understand the insights of the problem and be prepared in future to prevent/apprehend or mitigate the number or impact of attacks.

GTD Context and Content

The Global Terrorism Database (GTD) is an open-source database including information on terrorist attacks around the world from 1970 through 2017, *except 1993*.

Information on more than 180,000 Terrorist Attacks.

Geography: Worldwide. (1970 through 2017, *except 1993*)

Variables: 135 variables on location, tactics, perpetrators, targets, and outcomes ,….etc

The GTD includes systematic data on domestic as well as international terrorist incidents that have occurred during this time period and now includes more than 180,000 attacks.

Dataset

- **The Global Terrorism Database (GTD) is an open-source database including information on terrorist attacks around the world from 1970 through 2017. The GTD includes systematic data on domestic as well as international terrorist incidents that have occurred during this time period and now includes more than 180,000 attacks.**

- **Content**:
- Geography: Worldwide.
- Time period: 1970-2017, *except 1993.*
- Variables: 135 variables on location, tactics, perpetrators, targets, and outcomes ,….etc

- **The source of the dataset:**

https://www.kaggle.com/START-UMD/gtd#globalterrorismdb_0718dist.csv

- **Project repository:**

https://github.com/emilkaram/CKM136XJ0-Global-Terrorism-Data-Analytics-Capstone

Research question

Can a success of a terrorism attack be predicted by knowing the attack features?

GTD Definition of Terrorism and Inclusion Criteria

The GTD defines a **terrorist attack** as *the threatened or actual use of illegal force and violence by a non-state actor to attain a political, economic, religious, or social goal through fear, coercion, or intimidation*. In practice this means in order to consider an incident for inclusion in the GTD, *all three* of the following attributes must be present:

- ***The incident must be intentional .***
- ***The incident must entail some level of violence or immediate threat of violence.***
- ***The perpetrators of the incidents must be sub-national actors.***

In addition, *at least two* of the following three criteria must be present for an incident to be included in the GTD:

- **Criterion 1: The act must be aimed at attaining a political, economic, religious, or social goal**.
- **Criterion 2*:* There must be evidence of an intention to coerce, intimidate, or convey some other message to a larger audience (or audiences) than the immediate victims**
- **Criterion 3: The action must be outside the context of legitimate warfare activities**.

Data label / class
(the dependent variable)

- ***Success*** *(Categorical Variable)*
- 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.

Example of Success of a terrorist strike

- **ASSASSINATION**

In order for an assassination to be successful, the target of the assassination must be killed. For example, even if an attack kills numerous people but not the target, it is an unsuccessful assassination.

- **ARMED ASSAULT**

An armed assault is determined to be successful if the assault takes place and if a target is hit (including people and/or property). Unsuccessful armed assaults are those in which the perpetrators attack and do not hit the target. An armed assault is also unsuccessful if the perpetrators are apprehended on their way to commit the assault. To make this determination, however, there must be information to indicate that an actual assault was imminent.

- **BOMBING/EXPLOSION**

A bombing is successful if the bomb or explosive device detonates. Bombings are considered unsuccessful if they do not detonate. The success or failure of the bombing is not based on whether it hit the intended target.

- **HIJACKING**

A hijacking is successful if the hijackers assume control of the vehicle at any point, whereas a hijacking is unsuccessful if the hijackers fail to assume control of the vehicle. The success or failure of the hijacking is not based on whether the vehicle reached the intended destination of the hijackers.

Example of Success of a terrorist strike (continue…)

- **HOSTAGE TAKING (BARRICADE INCIDENT)**

A barricade incident is successful if the hostage takers assume control of the individuals at any point, whereas a barricade incident is unsuccessful if the hostage takers fail to assume control of the individuals.

- **HOSTAGE TAKING (KIDNAPPING)**

A kidnapping is successful if the kidnappers assume control of the individuals at any point, whereas a kidnapping is unsuccessful if the kidnappers fail to assume control of the individuals. (Kidnappings are distinguished from Barricade Incidents (above) in that they involve moving and holding the hostages in another location).

- **FACILITY / INFRASTRUCTURE ATTACK**

A facility attack is determined to be successful if the facility is damaged. If the facility has not been damaged, then the attack is unsuccessful.

- **UNARMED ASSAULT**

An unarmed assault is determined to be successful there is a victim that who has been injured. Unarmed assaults that are unsuccessful are those in which the perpetrators do not injure anyone. An unarmed assault is also unsuccessful if the perpetrators are apprehended when on their way to commit the assault. To make this determination, however, there must be information to indicate that an assault was imminent.

Attributes

GTD ID and Date	eventid	Numeric
	iyear	Numeric
	imonth	Numeric
	iday	Numeric
	approxdate	Text
	extended	Categorical
	resolution	Numeric Date
Incident Information	summary	Text
	crit1	Categorical
	crit2	Categorical
	crit3	Categorical
	doubtterr	Categorical
	alternative	Categorical
	alternative_txt	Categorical
	multiple	Categorical
	related	Text
Incident Location	country	Categorical
	country_txt	Categorical
	region	Categorical
	region_txt	Categorical
	provstate	Text
	city	Text
	vicinity	Categorical
	location	Text
	latitude	Numeric
	longitude	Numeric
	specificity	Categorical
Attack Information	attacktype1	Categorical
	attacktype1_txt	Categorical
	attacktype2	Categorical
	attacktype2_txt	Categorical
	attacktype3	Categorical
	attacktype3_txt	Categorical
	success	Categorical
	suicide	Categorical

Additional Information & Sources	addnotes	Text
	INT_LOG	Categorical
	INT_IDEO	Categorical
	INT_MISC	Categorical
	INT_ANY	Categorical
	scite1	Text
	scite2	Text
	scite3	Text
	dbsource	Text

Weapon Information	weaptype1	Categorical
	weaptype1_txt	Categorical
	weapsubtype1	Categorical
	weapsubtype1_txt	Categorical
	weaptype2	Categorical
	weaptype2_txt	Categorical
	weapsubtype2	Categorical
	weapsubtype2_txt	Categorical
	weaptype3	Categorical
	weaptype3_txt	Categorical
	weapsubtype3	Categorical
	weapsubtype3_txt	Categorical
	weaptype4	Categorical
	weaptype4_txt	Categorical
	weapsubtype4	Categorical
	weapsubtype4_txt	Categorical
	weapdetail	Text
Target/Victim Information	targtype1	Categorical
	targtype1_txt	Categorical
	targsubtype1	Categorical
	targsubtype1_txt	Categorical
	corp1	Text
	target1	Text
	natlty1	Categorical
	natlty1_txt	Categorical
	targtype2	Categorical
	targtype2_txt	Categorical
	targsubtype2	Categorical
	targsubtype2_txt	Categorical
	corp2	Text
	target2	Text
	natlty2	Categorical
	natlty2_txt	Categorical
	targtype3	Categorical
	targtype3_txt	Categorical
	targsubtype3	Categorical
	targsubtype3_txt	Categorical
	corp3	Text
	target3	Text
	natlty3	Categorical
	natlty3_txt	Categorical

Perpetrator Information	gname	Text
	gsubname	Text
	gname2	Text
	gsubname2	Text
	gname3	Text
	gsubname3	Text
	guncertain1	Categorical
	guncertain2	Categorical
	guncertain3	Categorical
	individual	Categorical
	nperps	Numerical
	nperpcap	Numerical
	claimed	Categorical
	claimmode	Categorical
	claimmode_txt	Categorical
	compclaim	Categorical
	claim2	Categorical
	claimmode2	Categorical
	claimmode2_txt	Categorical
	claim3	Categorical
	claimmode3	Categorical
	claimmode3_txt	Categorical
	motive	Text
Casualties and Consequences	nkill	Numeric
	nkillus	Numeric
	nkillter	Numeric
	nwound	Numeric
	nwoundus	Numeric
	nwoundte	Numeric
	property	Categorical
	propextent	Categorical
	propextent_txt	Categorical
	propvalue	Numeric
	propcomment	Text
	ishostkid	Categorical
	nhostkid	Numeric
	nhostkidus	Numeric
	nhours	Numeric
	ndays	Numeric
	divert	Text
	kidhijcountry	Text
	ransom	Categorical
	ransomamt	Numeric
	ransomamtus	Numeric
	ransompaid	Numeric
	ransompaidus	Numeric
	ransomnote	Text
	hostkidoutcome	Categorical
	hostkidoutcome_txt	Categorical
	nreleased	Numeric

Approach

initial Analysis

- **Univariate Analysis** (detailed data dictionary / list all variables and understand what each variable mean and represent)(what is the source of each variable directly measured or calculated based on other variables or does the variable has a time domain associated) - (Decide on the dependent (target) variable) (Assigning the correct data types and appropriate column names)(for numeric attributes check the 5 number summary min 1sQ mean 3rdQ max)(for categorical attributes check and decide on the levels , frequency) (check and deal with data inconsistencies , missing values , errors , duplicates , outliers (boxplots) , numeric signs , upper and lower cases , spaces or special characters in striges) (check distributions of the variables : Normal distribution?)(Low variance filter)(check the imbalance in the dependent variable) (check time variable) (Univariate visualizations)
- **BiVariate Analysis** (pairwise relations) (pairwise visuliazations like scatter plots) (correlation analysis spearman or pearson).
- **Multivariate Analysis** (relations between more than 2 variables)(statistical tools such as one way ANOVA analysis or rank or to compare the means.

Exploratory Analysis

- Normalizing / scaling
- Sub-setting the data
- Decision rules , association rules , n grams
- Clustering such as K-mean
- Hypothesise analysis
- Correlation analysis
- NLP analysis for some text attributes

Dimensionality Reduction

- remove attributes with too many missing values
- remove attributes with zero or very low variance
- remove one of the attributes with high correlations with other - prefer the one with more missing values or lower variance
- Feature selection(decide on the importance of the attribute using statistical measures like information gain or Gini index) (Forward selection and backward elimination)

Experimental Desgin

- Randomizing, Splitting the data into training and test sets to training, validation and testing
- Treatment for imbalance (under-sampling the majority class and over-sampling the minority class)
- Cross validation such as 10 folds

Modeling

- building the Classification Models
- Train the model
- Validate the model

Evaluation

- Classification (Confusion matric , ROC (receiver operating characteristics) , Accuracy , Recall , Precision)
- compare performance between different models (contingency tables , multivariate analysis of variance)

Improving the Model

- deal with very high accuracy(Fixation variables , overfitting)
- deal with very low performance (many need new variables or more observations)
- iterate to improve the model performance

Conclusions

- Inferences , dissuasions
- Threats to validity (Internal , External , construct) and propose the solution that may mitigate these threats
- Next/ future steps

Approach:

Initial Analysis

initial Analysis

- **Univariate Analysis** (detailed data dictionary / list all variables and understand what each variable mean and represent)(what is the source of each variable directly measured or calculated based on other variables or does the variable has a time domain associated) - (Decide on the dependent (target) variable) (Assigning the correct data types and appropriate column names)(for numeric attributes check the 5 number summary min 1sQ mean 3rdQ max)(for categorical attributes check and decide on the levels , frequency) (check and deal with data inconsistencies , missing values , errors , duplicates , outliers (boxplots) , numeric signs , upper and lower cases , spaces or special characters in striges) (check distributions of the variables : Normal distribution?)(Low variance filter)(check the imbalance in the dependent variable) (check time variable) (Univariate visualizations)
- **BiVariate Analysis** (pairwise relations) (pairwise visuliazations like scatter plots) (correlation analysis spearman or pearson).
- **Multivariate Analysis** (relations between more than 2 variables)(statistical tools such as one way ANOVA analysis or rank or to compare the means.

Exploratory Analysis

- Normalizing / scaling
- Sub-setting the data
- Decision rules , association rules , n grams
- Clustering such as K-mean
- Hypothesise analysis
- Correlation analysis
- NLP analysis for some text attributes

Dimensionality Reduction

- remove attributes with too many missing values
- remove attributes with zero or very low variance
- remove one of the attributes with high correlations with other - prefer the one with more missing values or lower variance
- Feature selection(decide on the importance of the attribute using statistical measures like information gain or Gini index) (Forward selection and backward elimination)

Experimental Desgin

- Randomizing, Splitting the data into training and test sets to training, validation and testing
- Treatment for imbalance (under-sampling the majority class and over-sampling the minority class)
- Cross validation such as 10 folds

Modeling

- building the Classification Models
- Train the model
- Validate the model

Evaluation

- Classification (Confusion matric , ROC (receiver operating characteristics) , Accuracy , Recall , Precision)
- compare performance between different models (contingency tables , multivariate analysis of variance)

Improving the Model

- deal with very high accuracy(Fixation variables , overfitting)
- deal with very low performance (many need new variables or more observations)
- iterate to improve the model performance

Conclusions

- Inferences , dissuasions
- Threats to validity (Internal , External , construct) and propose the solution that may mitigate these threats
- Next/ future steps

Initial Analysis

Univariate Analysis

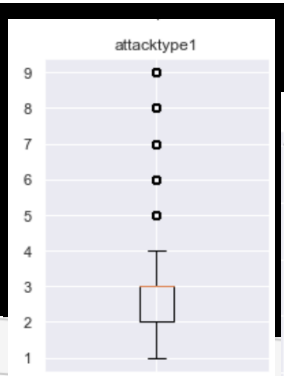
```
mydata.shape
(181691, 135)
```

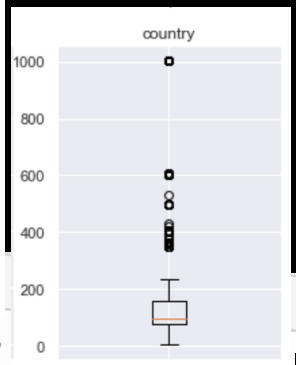
```
obj1.describe()
```

	approxdate	resolution	summary	alternative_txt	related	country_txt	region_txt	provstate	city
count	9239	2220	115562	29011	25038	181691	181691	181270	181257
unique	2244	1859	112492	5	14306	205	12	2855	36674
top	September 18-24, 2016	8/04/98	09/00/2016: Sometime between September 18, 201...	Insurgency/Guerilla Action	201612010023, 201612010024, 201612010025, 2016...	Iraq	Middle East & North Africa	Baghdad	Unknowr
freq	101		100	23410	80	24636	50474	7645	9775

```
mydata.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 181691 entries, 0 to 181690
Columns: 135 entries, eventid to dbsource
dtypes: float64(56), int64(21), object(58)
memory usage: 187.1+ MB
```

Outliers






```
mydata.columns
Index(['eventid', 'iyear', 'imonth', 'iday', 'approxdate',
       'resolution', 'summary', 'crit1', 'crit2',
       ...
       'nreleased', 'addnotes', 'INT_LOG', 'INT_IDEO', 'INT_MISC', 'INT_ANY',
       'scite1', 'scite2', 'scite3', 'dbsource'],
      dtype='object', length=135)
```

```
mydata.describe()
```

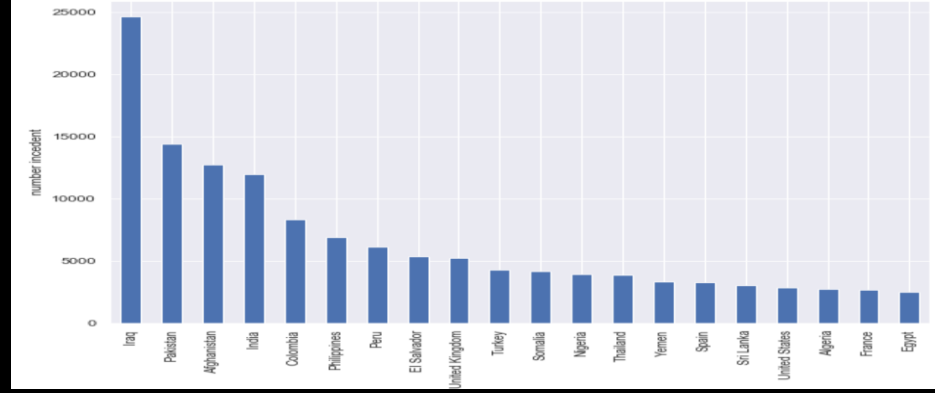
	eventid	iyear	imonth	iday	extended	crit1	crit2
count	1.816910e+05	181691.000000	181691.000000	181691.000000	181691.000000	181691.000000	181691.000000
mean	2.002704e+11	2002.638997	6.467277	15.505644	0.045346	0.988530	0.993093
std	1.325955e+09	13.259430	3.388303	8.814045	0.208063	0.106483	0.082823
min	1.970000e+11	1970.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	1.991020e+11	1991.000000	4.000000	8.000000	0.000000	1.000000	1.000000
50%	2.009020e+11	2009.000000	6.000000	15.000000	0.000000	1.000000	1.000000
75%	2.014080e+11	2014.000000	9.000000	23.000000	0.000000	1.000000	1.000000
max	2.017120e+11	2017.000000	12.000000	31.000000	1.000000	1.000000	1.000000

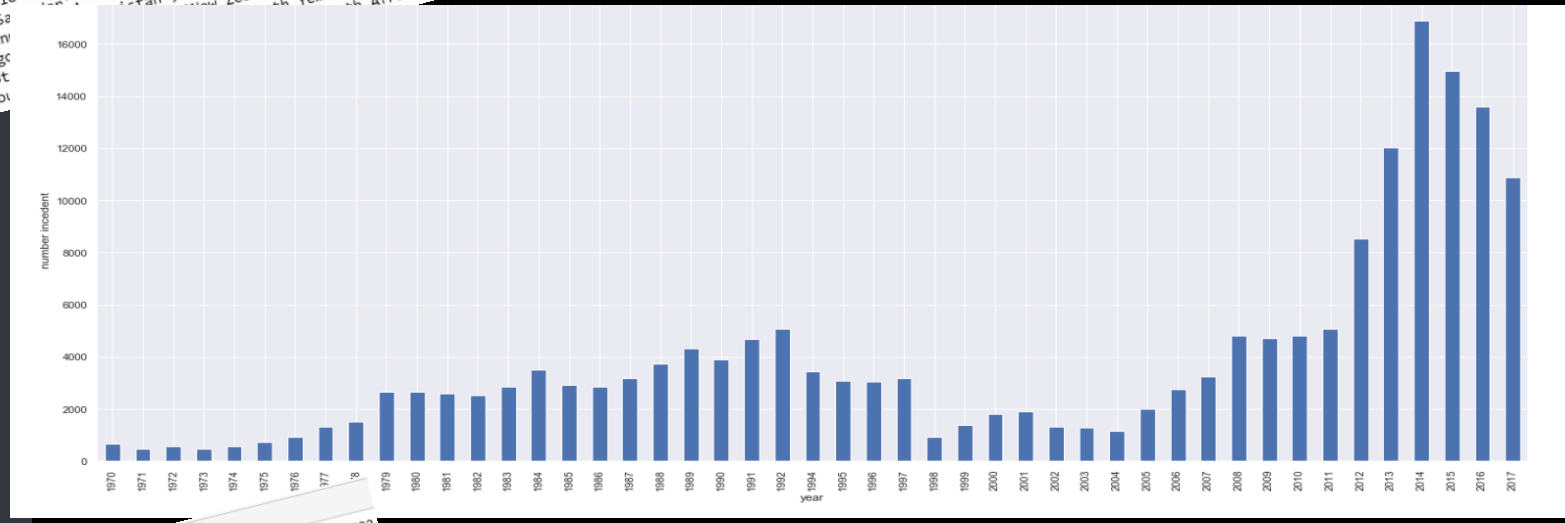
Initial Analysis

Univariate Analysis

```
#List the Levels
mydata['country_txt'].unique()
```

```
array(['Dominican Republic', 'Mexico', 'Philippines', 'Greece', 'Japan',
       'United States', 'Uruguay', 'Italy', 'East Germany (GDR)',
       'Ethiopia', 'Guatemala', 'Venezuela', 'West Germany (FRG)',
       'Switzerland', 'Jordan', 'Spain', 'Brazil', 'Egypt', 'Argentina',
       'Lebanon', 'Ireland', 'Turkey', 'Paraguay', 'Iran',
       'United Kingdom', 'Colombia', 'Bolivia', 'Nicaragua',
       'Netherlands', 'Belgium', 'Canada', 'Australia', 'Pakistan',
       'Zambia', 'Sweden', 'Costa Rica', 'South Yemen', 'Cambodia',
       'Israel', 'Poland', 'Taiwan', 'Panama', 'Kuwait',
       'West Bank and Gaza Strip', 'Austria', 'Czechoslovakia', 'India',
       'France', 'South Vietnam', 'Brunei', 'Zaire', 'Algeria', 'Cyprus',
       "People's Republic of the Congo", 'Portugal', 'Sudan', 'Morocco',
       'El Sa', 'Thailand', 'Haiti', 'Chile', 'Honduras', 'Singapore',
       'Myan', 'Yugo', 'Bot', 'Sou', 'Zealand', 'Malaysia', 'Andorra', 'Syria',
                                                  'Yemen', 'Africa', 'Kenya',
```






```
mydata['gname'].value_counts()
```

```
Unknown                                                  82782
Taliban                                                   7478
Islamic State of Iraq and the Levant (ISIL)               5613
Shining Path (SL)                                         4555
Farabundo Marti National Liberation Front (FMLN)          3351
Al-Shabaab                                                3288
New People's Army (NPA)                                   2772
Irish Republican Army (IRA)                               2671
Revolutionary Armed Forces of Colombia (FARC)             2487
Boko Haram                                                2418
Kurdistan Workers' Party (PKK)                            2310
Basque Fatherland and Freedom (ETA)                       2024
Communist Party of India - Maoist (CPI-Maoist)            1878
Maoists                                                   1630
Liberation Tigers of Tamil Eelam (LTTE)                   1606
National Liberation Army of Colombia (ELN)                1561
Tehrik-i-Taliban Pakistan (TTP)                           1351
Palestinians                                              1125
Houthi extremists (Ansar Allah)                           1064
Al-Qaida in the Arabian Peninsula (AQAP)                  1012
```

```
mydata['success'].describe()
```

```
count    181691.000000
mean          0.889598
std           0.313391
min           0.000000
25%           1.000000
50%           1.000000
75%           1.000000
max           1.000000
Name: success, dtype: float64
```

```
mydata['country_txt'].describe()
```

```
count        181691
unique          205
top            Iraq
freq          24636
Name: country_txt, dtype: object
```

```
# Frequency
mydata['country_txt'].value_counts()
```

```
Iraq                    24636
Pakistan                14368
Afghanistan             12731
India                   11960
Colombia                 8306
Philippines              6908
Peru                     6096
El Salvador              5320
United Kingdom           5235
Turkey                   4292
Somalia                  4142
Nigeria                  3907
Thailand                 3849
Yemen
```

Cleaning data

Criterion 1: The act must be aimed at attaining a political, economic, religious, or social goal.

Criterion 2: There must be evidence of an intention to coerce, intimidate, or convey some other message to a larger audience (or audiences) than the immediate victims.

Criterion 3: The action must be outside the context of legitimate warfare activities.

```
mydata['doubtterr'] = mydata['doubtterr'].replace(-9,0)
```

```
#mydata.doubtterr
```

```
mydata = mydata[(mydata.crit1 == 1) & (mydata.crit2 == 1) & (mydata.crit3 == 1) & (mydata.doubtterr == 0)]
```

```
mydata.shape
```

```
(152622, 135)
```

Cleaning data: Dealing with null values

```
# find Null value , calculate % of missing value from total , sort de-ascend
list1 =100*mydata.isnull().sum()/mydata.shape[0]
list1.sort_values(ascending=False)
```

```
alternative_txt     99.996069
alternative         99.996069
gsubname3           99.994103
weapsubtype4_txt    99.961342
weapsubtype4        99.961342
weaptype4_txt       99.958722
weaptype4           99.958722
claimmode3          99.929892
claimmode3_txt      99.929892
gsubname2           99.911546
guncertain3         99.836197
claim3              99.836197
gname3              99.833576
divert              99.803436
attacktype3_txt     99.741191
attacktype3         99.741191
ransomnote          99.701223
claimmode2_txt      99.678945
claimmode2          99.678945
ransompaidus        99.676325
ransomamtus         99.669772
ransompaid          99.549213
corp3               99.384754
targsubtype3_txt    99.340200
targsubtype3        99.340200
```

```
#droping more than 75% Null values attributes
mylist = []
for i in mydata:
    if 100*mydata[i].isnull().sum()/mydata.shape[0] > 75:
        mylist.append(i)
```

```
mydata = mydata.drop(mylist,axis=1)
```

```
mydata.shape
```

```
(152622, 65)
```

Cleaning data: Filling null values and Text to lower case

```
#fill null vaules
mydata['weapsubtype1_txt'].fillna('No Record', inplace=True)
mydata['natlty1_txt'].fillna('unknown', inplace=True)
mydata['target1'].fillna('unknown', inplace=True)
mydata['city'].fillna('unknown', inplace=True)
mydata['provstate'].fillna('unknown', inplace=True)

mydata['country_txt'].fillna('unknown', inplace=True)
mydata['region_txt'].fillna('unknown', inplace=True)
mydata['attacktype1_txt'].fillna('unknown', inplace=True)
mydata['targtype1_txt'].fillna('unknown', inplace=True)
```

```
#replaceve unk with unknown
mydata.target1 = mydata.target1.replace('unk','unknown')
```

```
# fill missing value for nkill and nwound with the median
mydata.nkill = np.round(mydata.nkill.fillna(mydata.nkill.median()))
mydata.nwound = np.round(mydata.nwound.fillna(mydata.nwound.median(
```

```
mydata.weaptype1_txt.replace(
    'Vehicle (not to include vehicle-borne explosives, i.e., car or t    bombs)',
    'Vehicle', inplace = True)
```

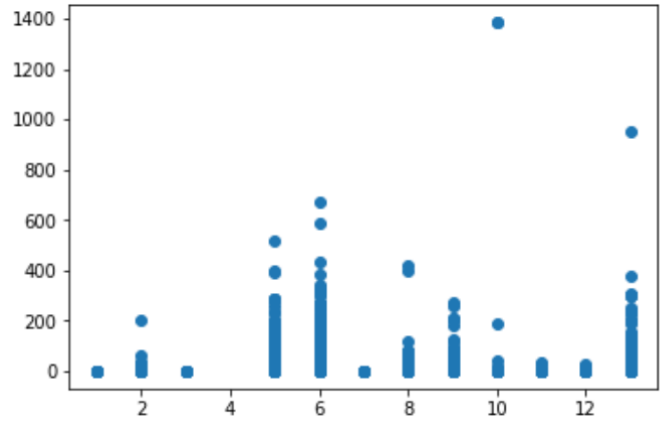
```
# set text to lower case
mydata.target1 = mydata.target1.str.lower()
mydata.gname = mydata.gname.str.lower()
mydata.summary = mydata.summary.str.lower()
mydata.city = mydata.city.str.lower()
mydata.weapsubtype1_txt = mydata.weapsubtype1_txt.str.lower()
mydata.natlty1_txt = mydata.natlty1_txt.str.lower()
mydata.provstate = mydata.provstate.str.lower()

mydata.country_txt = mydata.country_txt.str.lower()
mydata.region_txt = mydata.region_txt.str.lower()
mydata.attacktype1_txt = mydata.attacktype1_txt.str.lower()
mydata.targtype1_txt = mydata.targtype1_txt.str.lower()
```

Initial Analysis

Bivariate Analysis

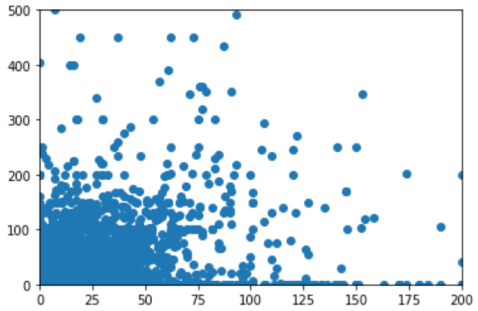
```
plt.scatter(mydata['weaptype1'] , mydata['nkill'])
```

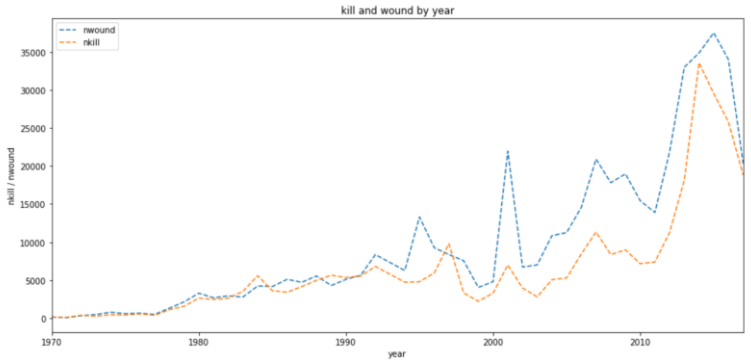
```
<matplotlib.collections.PathCollection at 0x1dcae84b780>
```




```
ax =plt.subplot(1,1,1)
ax.scatter(mydata['nkill'] , mydata['nwound'])
ax.set_xlim([0,200])
ax.set_ylim([0,500])
```

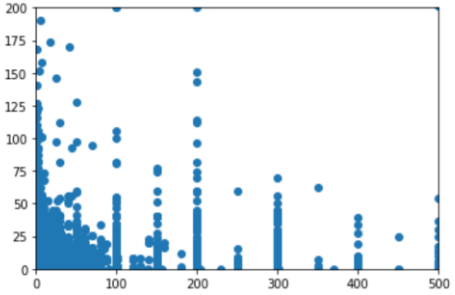
```
(0, 500)
```



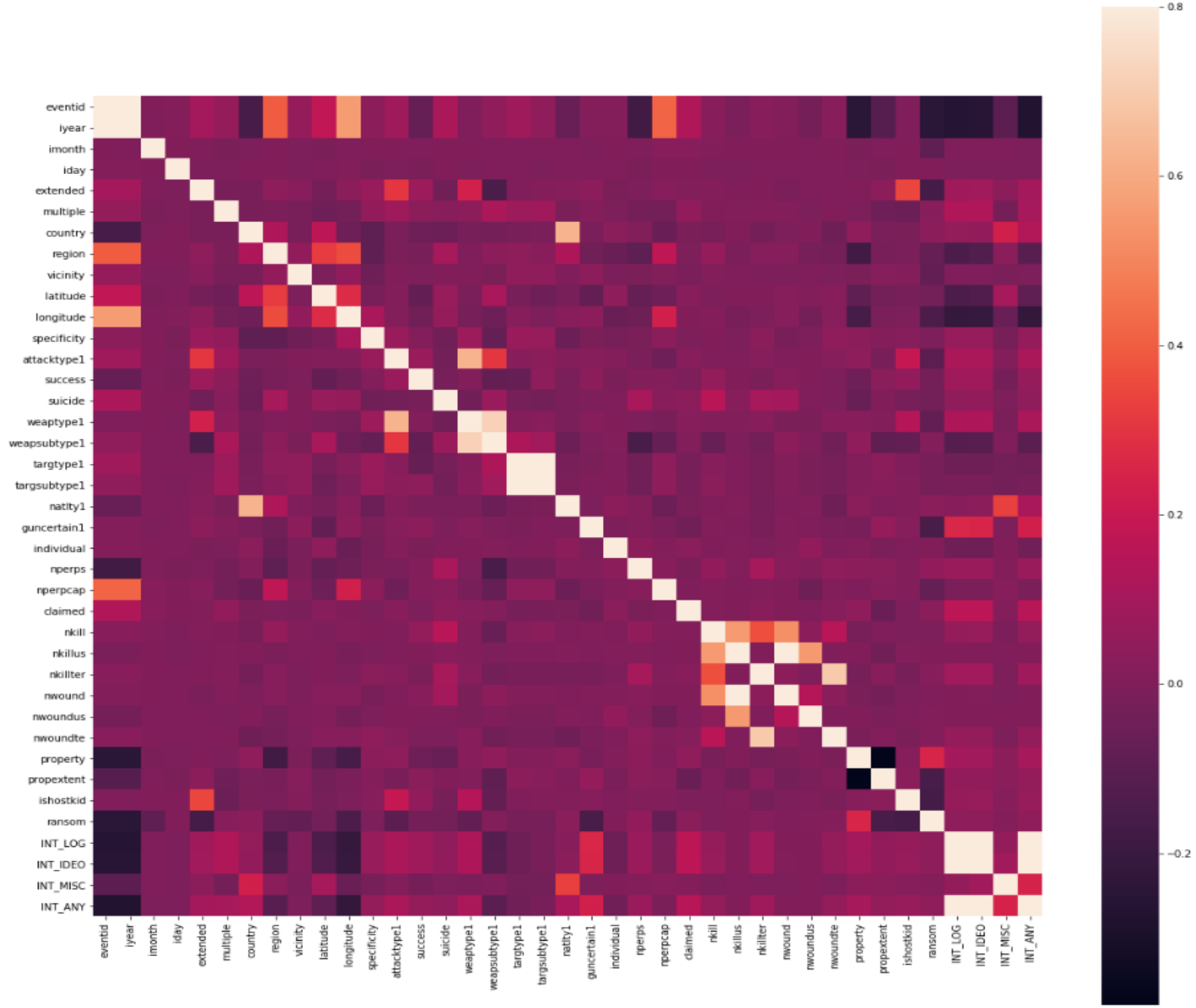



```
ax =plt.subplot(1,1,1)
ax.scatter(mydata['nperps'] , mydata['nkill'])
ax.set_xlim([0,500])
ax.set_ylim([0,200])
```

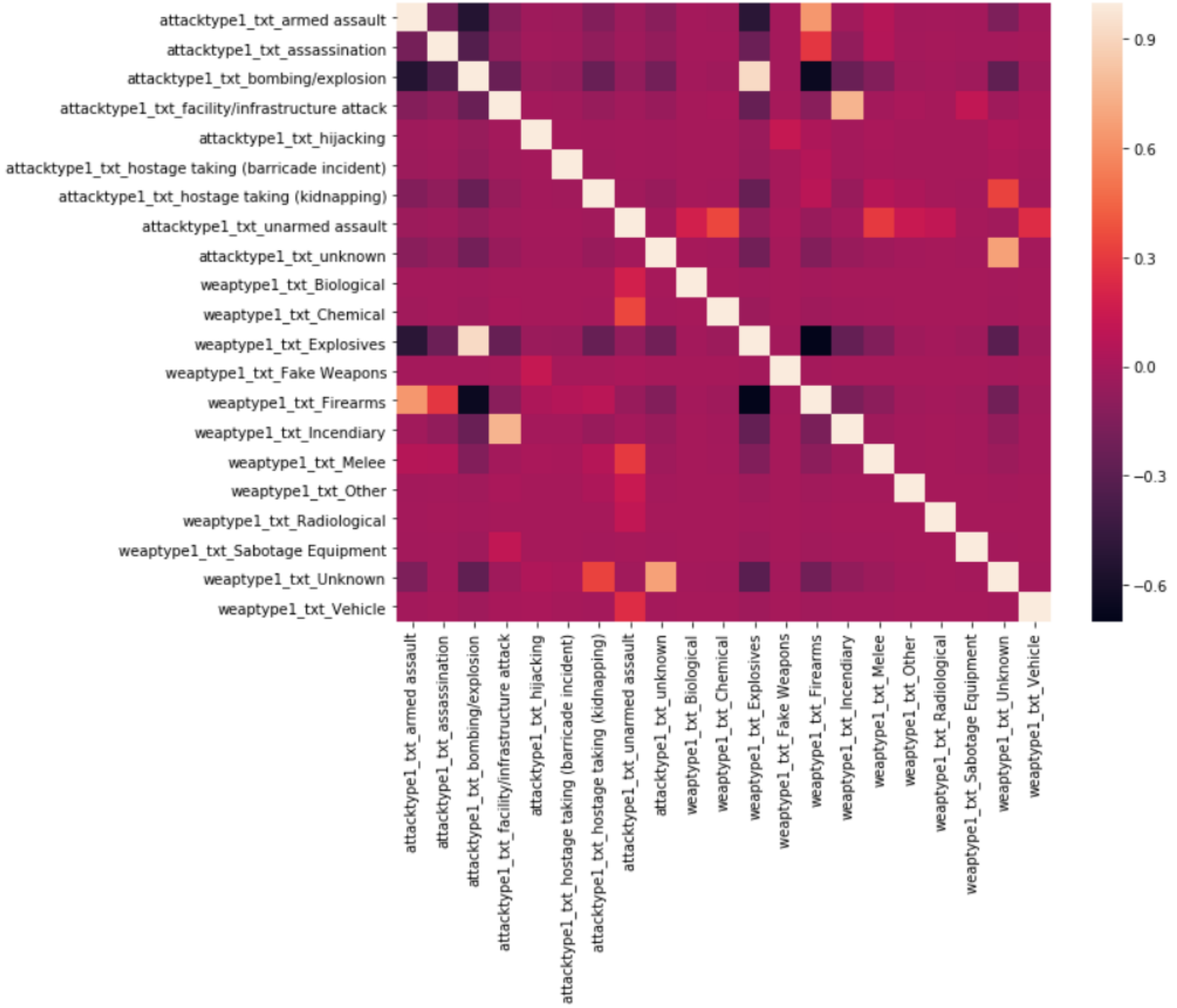
```
(0, 200)
```



Initial Analysis
Multivariate Analysis



Initial Analysis
Multivariate Analysis



Approach:

Exploratory Analysis

initial Analysis

- **Univariate Analysis** (detailed data dictionary / list all variables and understand what each variable mean and represent)(what is the source of each variable directly measured or calculated based on other variables or does the variable has a time domain associated) - (Decide on the dependent (target) variable) (Assigning the correct data types and appropriate column names)(for numeric attributes check the 5 number summary min 1sQ mean 3rdQ max)(for categorical attributes check and decide on the levels , frequency) (check and deal with data inconsistencies , missing values , errors , duplicates , outliers (boxplots) , numeric signs , upper and lower cases , spaces or special characters in striges) (check distributions of the variables : Normal distribution?)(Low variance filter)(check the imbalance in the dependent variable) (check time variable) (Univariate visualizations)
- **BiVariate Analysis** (pairwise relations) (pairwise visuliazations like scatter plots) (correlation analysis spearman or pearson).
- **Multivariate Analysis** (relations between more than 2 variables)(statistical tools such as one way ANOVA analysis or rank or to compare the means.

Exploratory Analysis

- Normalizing / scaling
- Sub-setting the data
- Decision rules , association rules , n grams
- Clustering such as K-mean
- Hypothesise analysis
- Correlation analysis
- NLP analysis for some text attributes

Dimensionality Reduction

- remove attributes with too many missing values
- remove attributes with zero or very low variance
- remove one of the attributes with high correlations with other - prefer the one with more missing values or lower variance
- Feature selection(decide on the importance of the attribute using statistical measures like information gain or Gini index) (Forward selection and backward elimination)

Experimental Desgin

- Randomizing, Splitting the data into training and test sets to training, validation and testing
- Treatment for imbalance (under-sampling the majority class and over-sampling the minority class)
- Cross validation such as 10 folds

Modeling

- building the Classification Models
- Train the model
- Validate the model

Evaluation

- Classification (Confusion matric , ROC (receiver operating characteristics) , Accuracy , Recall , Precision)
- compare performance between different models (contingency tables , multivariate analysis of variance)

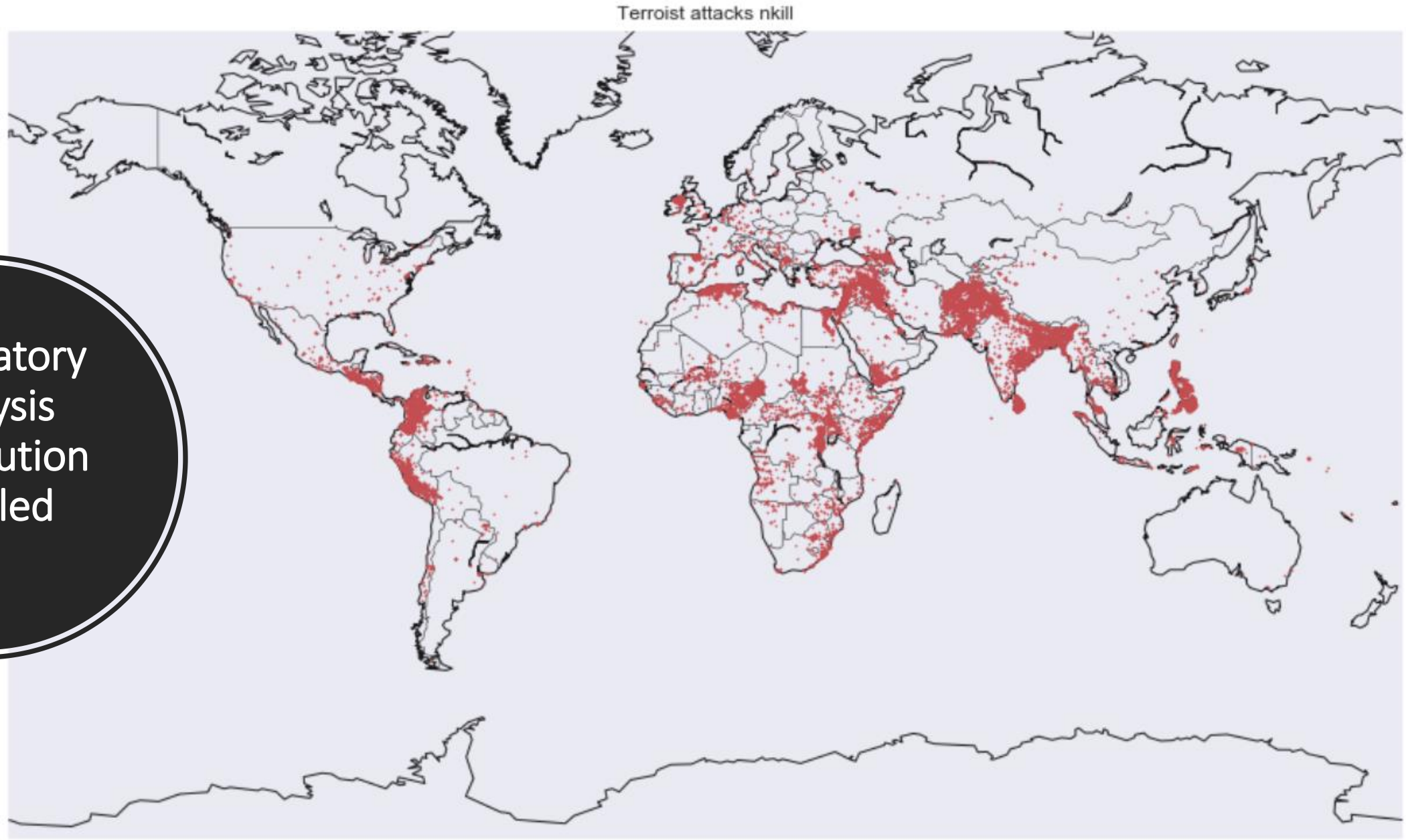
Improving the Model

- deal with very high accuracy(Fixation variables , overfitting)
- deal with very low performance (many need new variables or more observations)
- iterate to improve the model performance

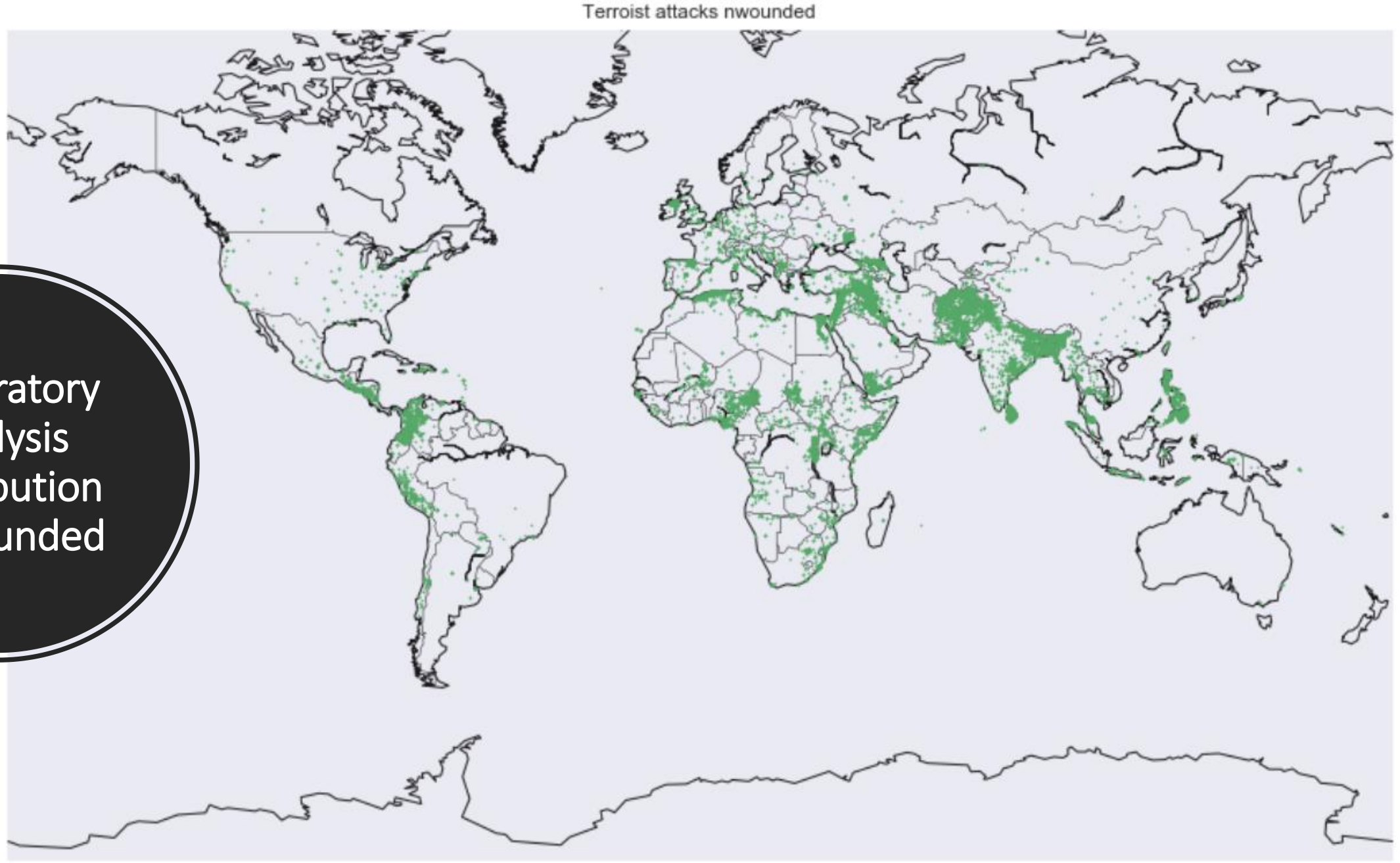
Conclusions

- Inferences , dissuasions
- Threats to validity (Internal , External , construct) and propose the solution that may mitigate these threats
- Next/ future steps

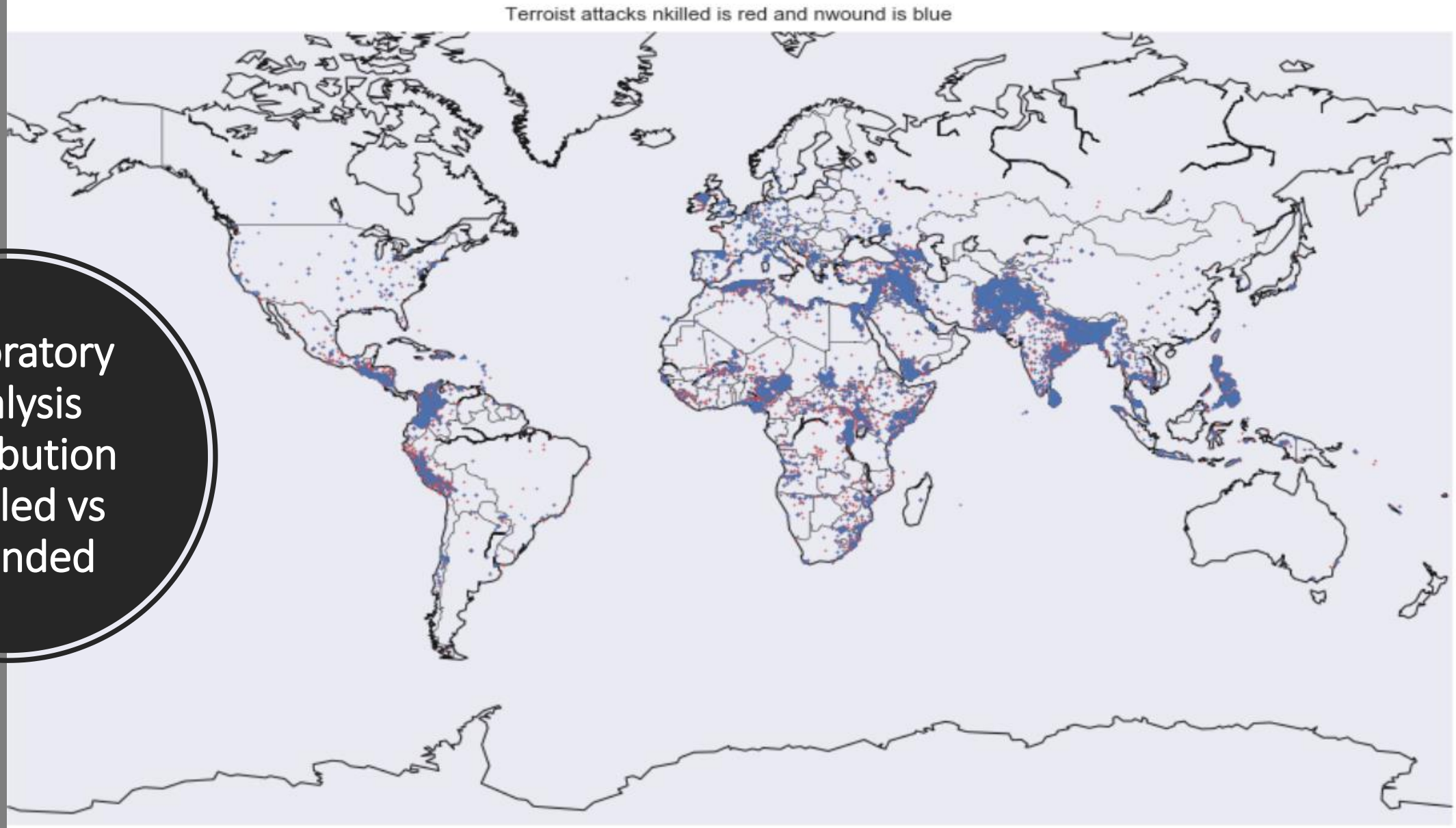
Exploratory Analysis Distribution of killed



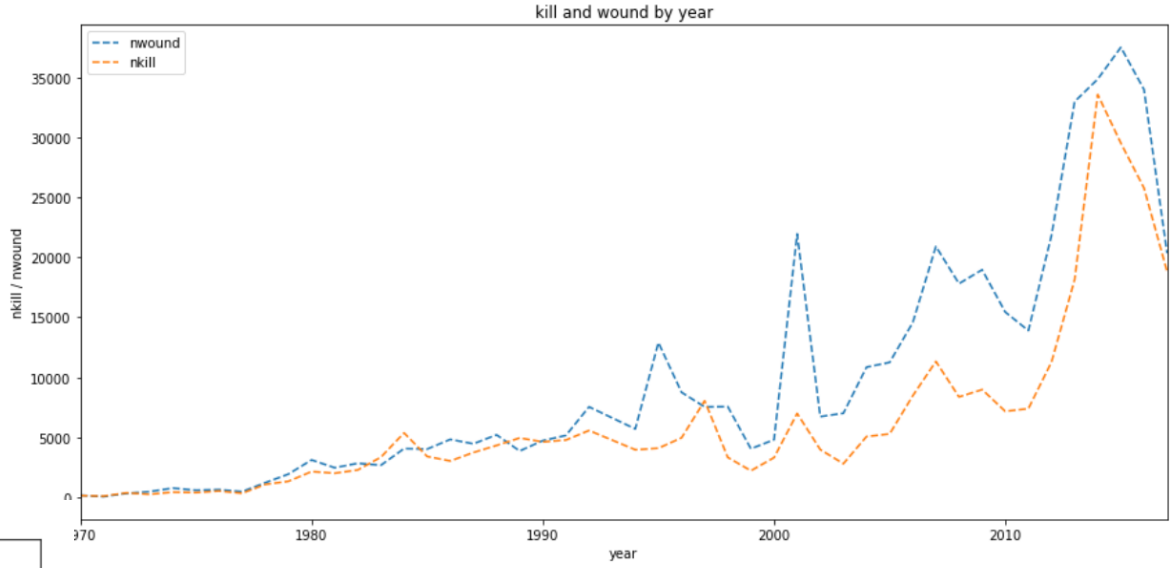
Exploratory Analysis Distribution of wounded

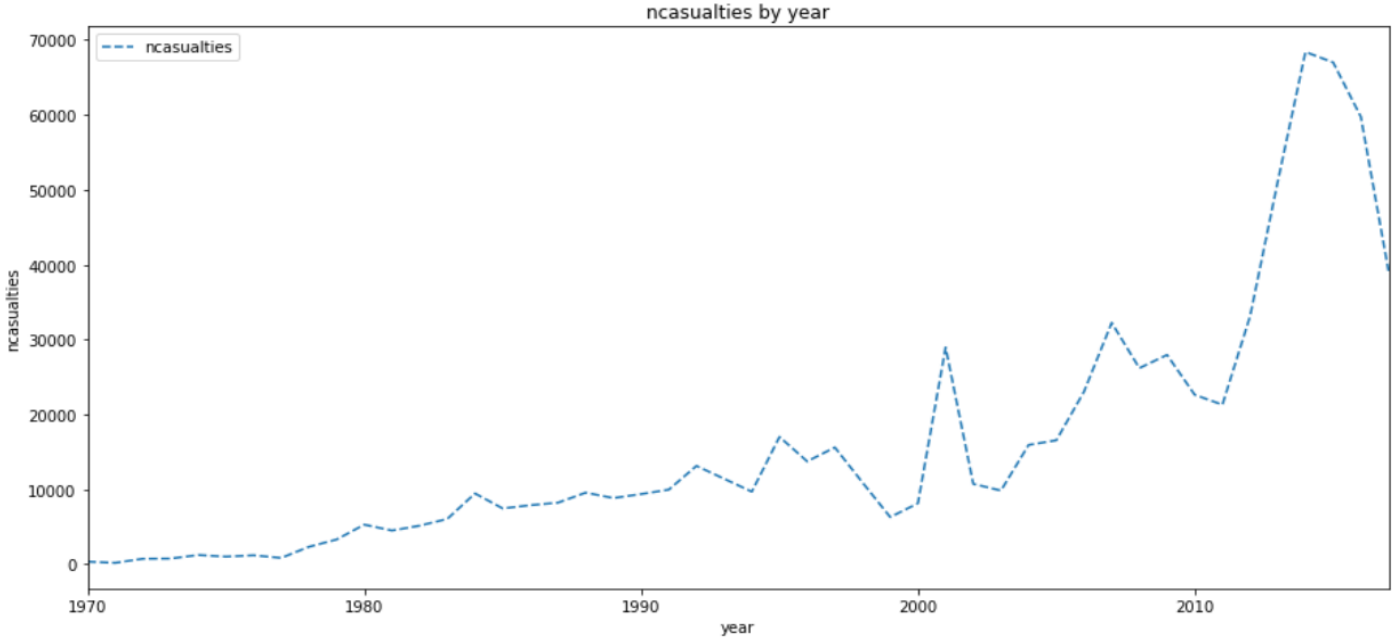


Exploratory Analysis Distribution of killed vs wounded

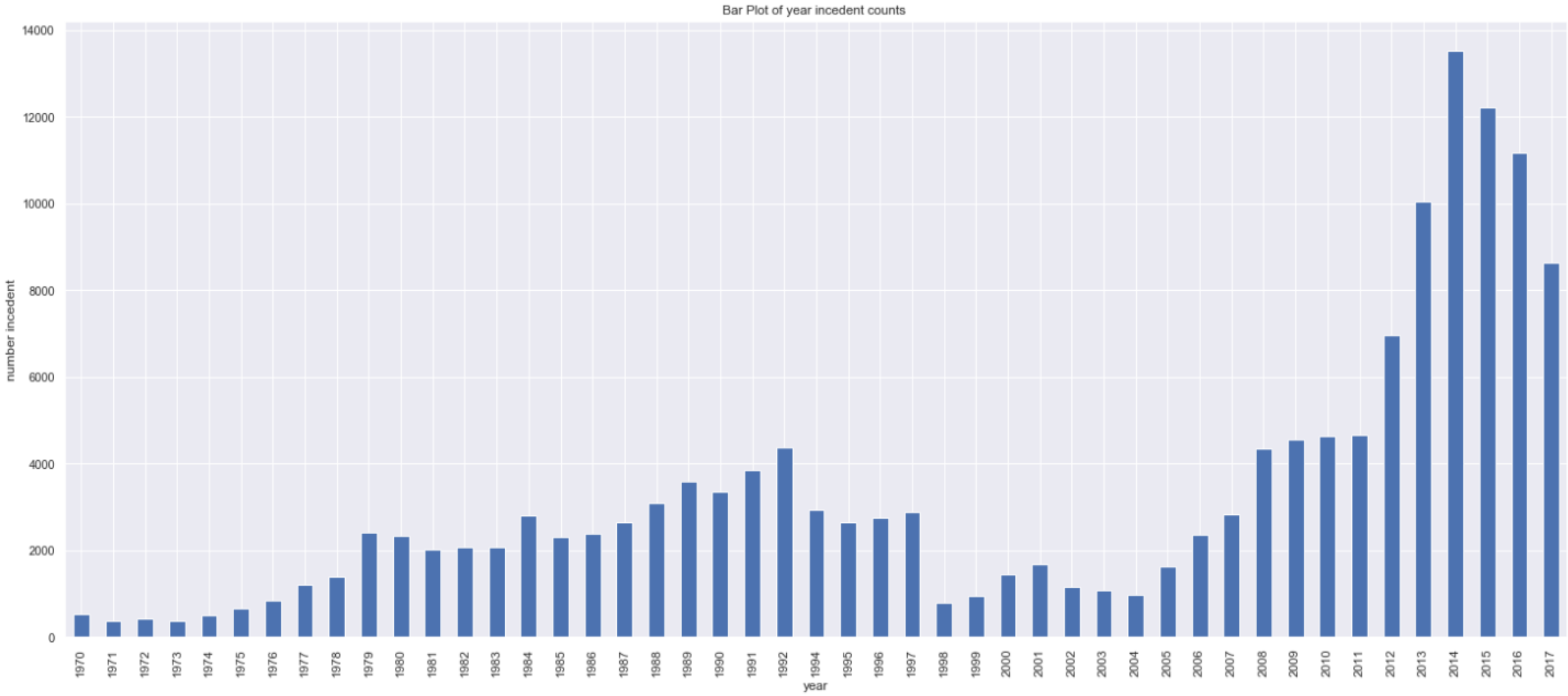


Exploratory Analysis

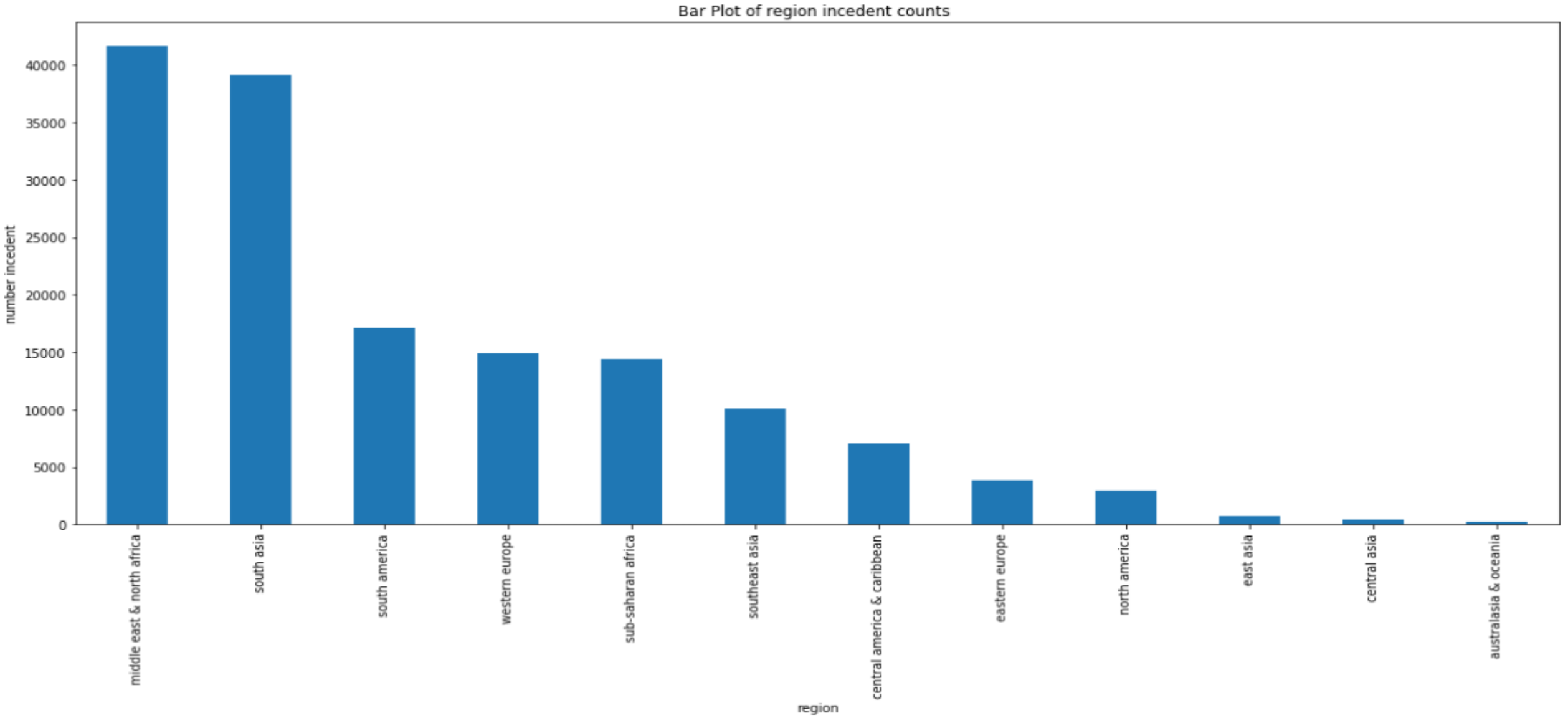




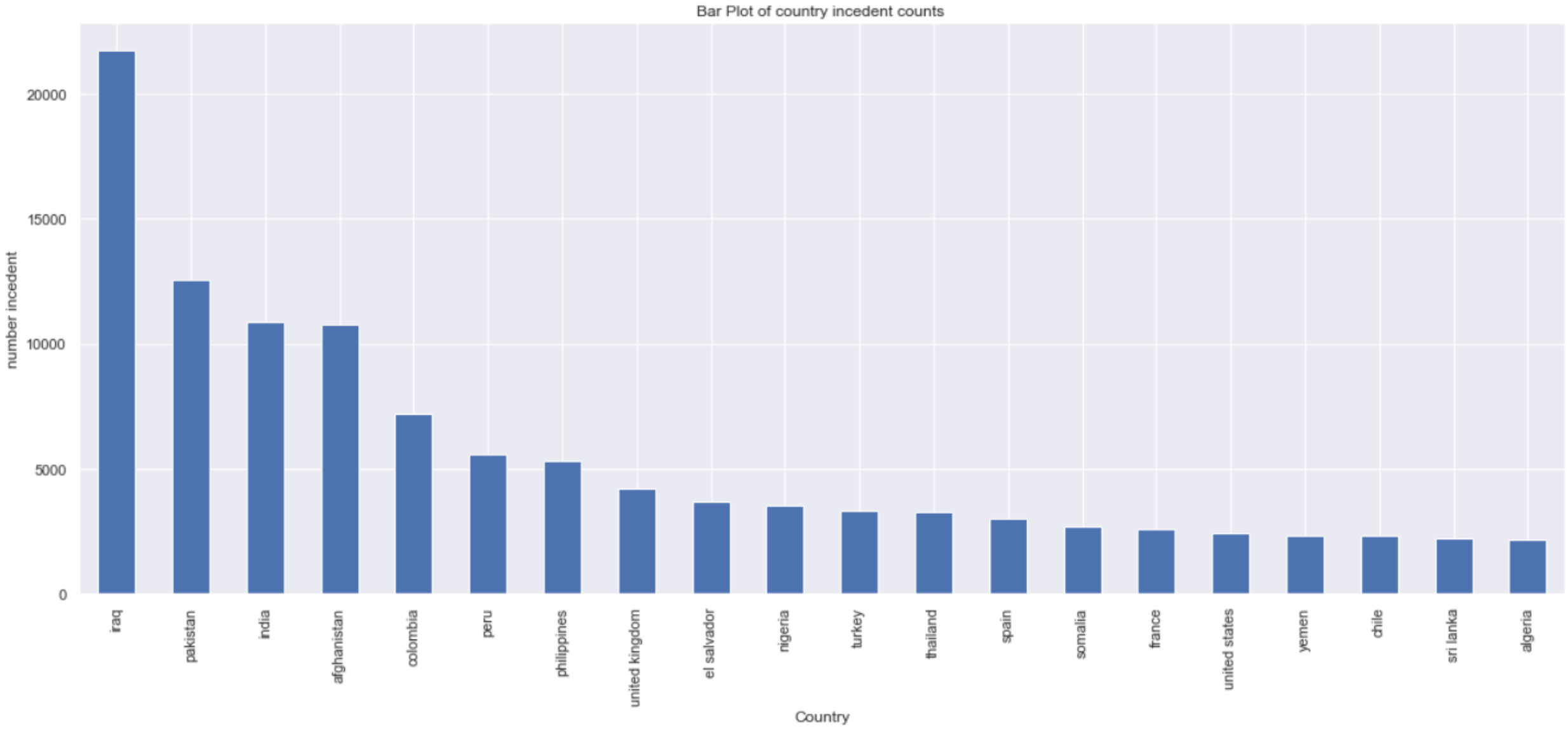
Exploratory Analysis



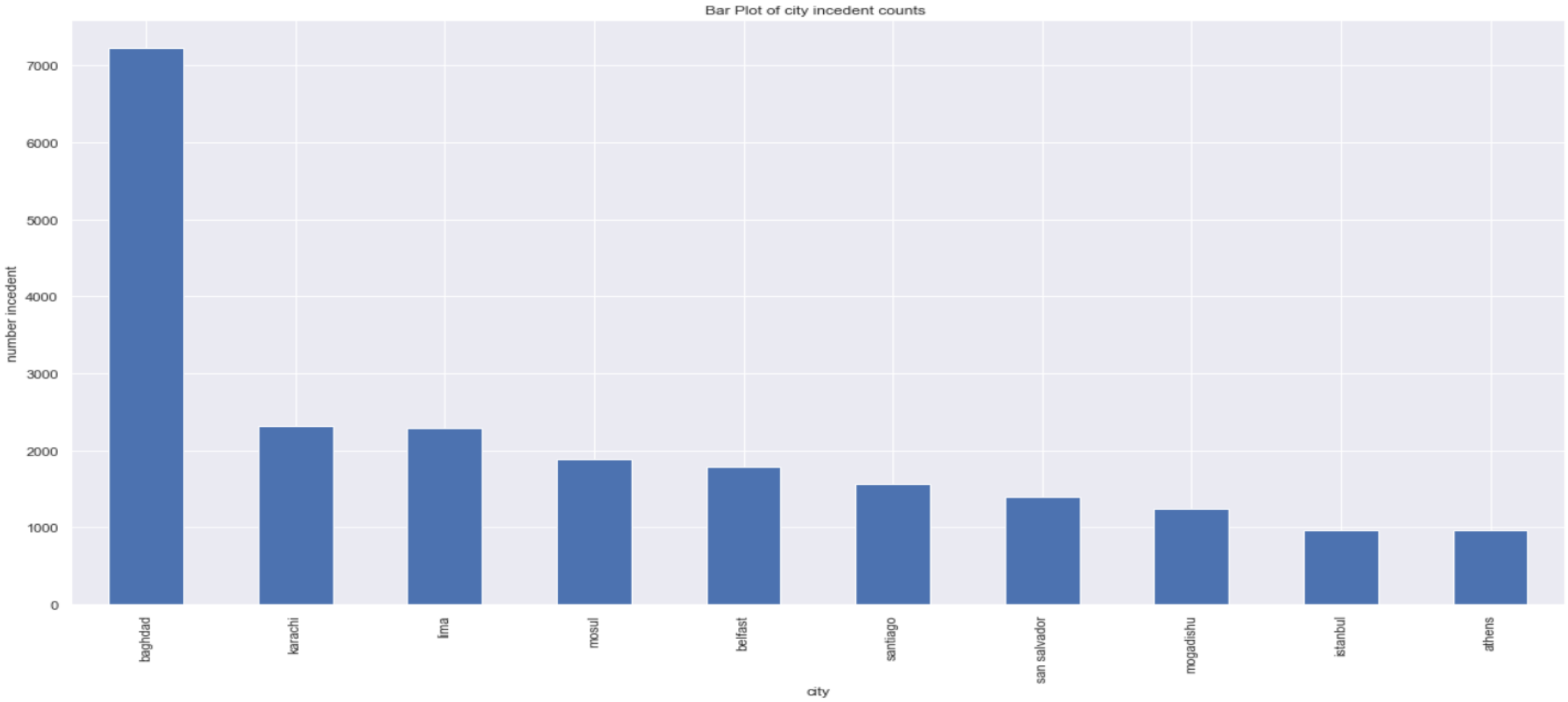
Exploratory Analysis



Exploratory Analysis

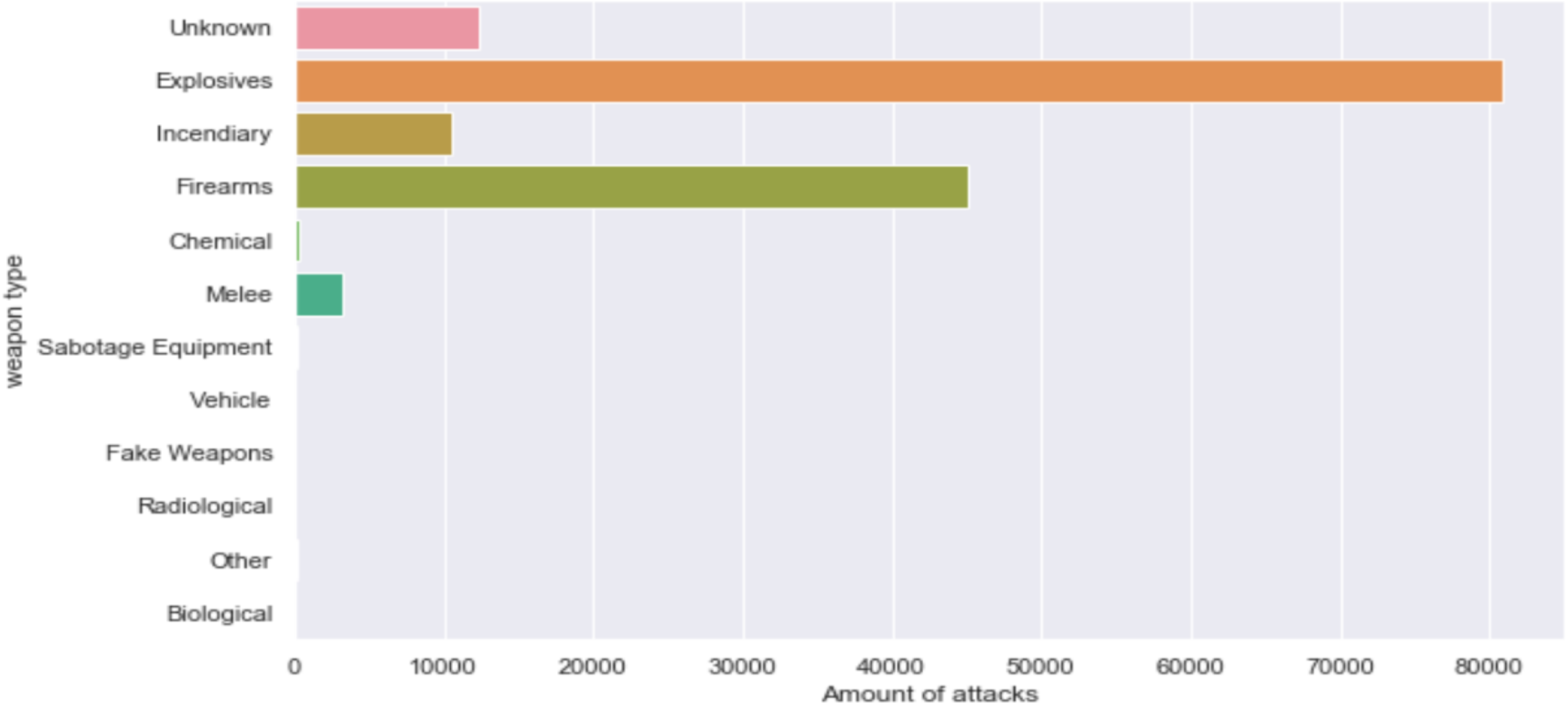


Exploratory Analysis

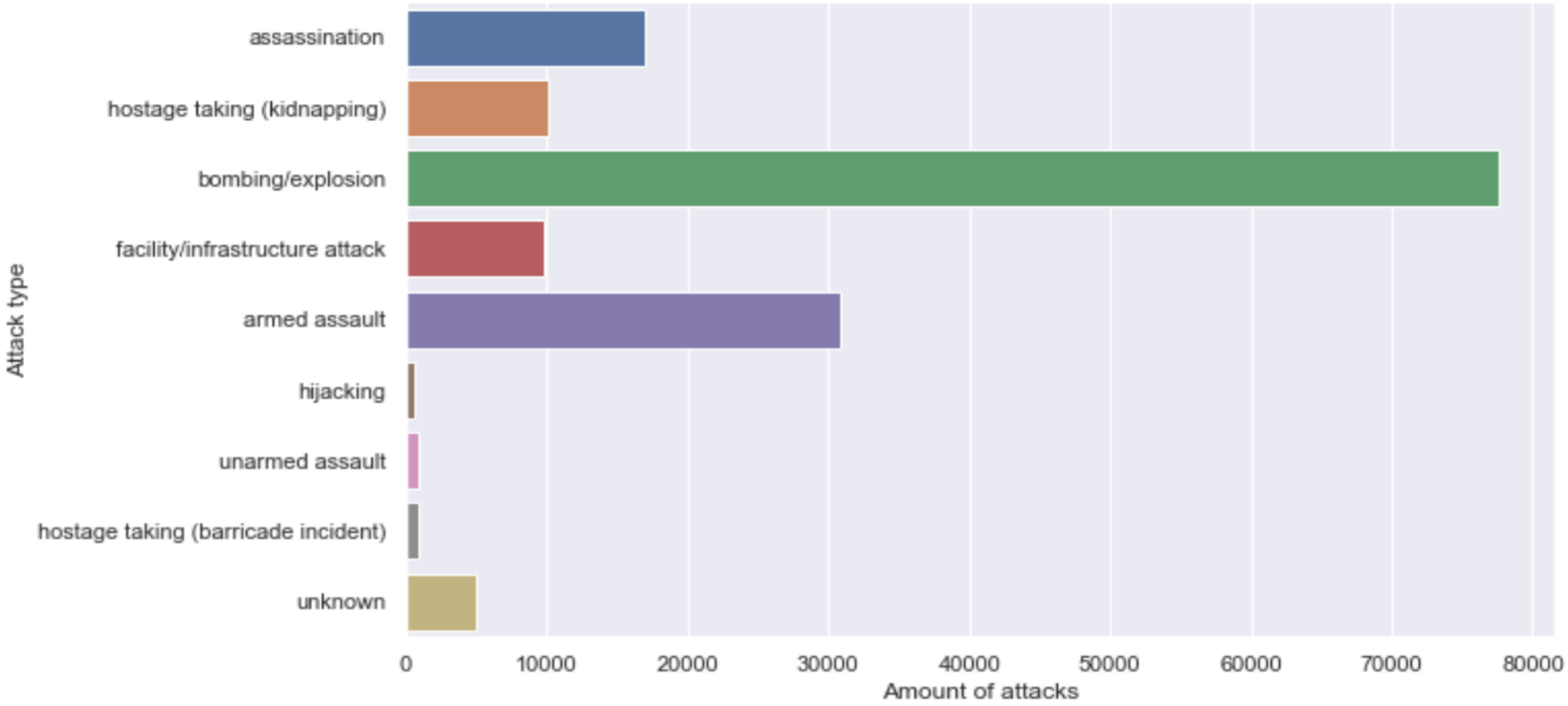


Exploratory Analysis (NLP) (Motives Word Cloud)

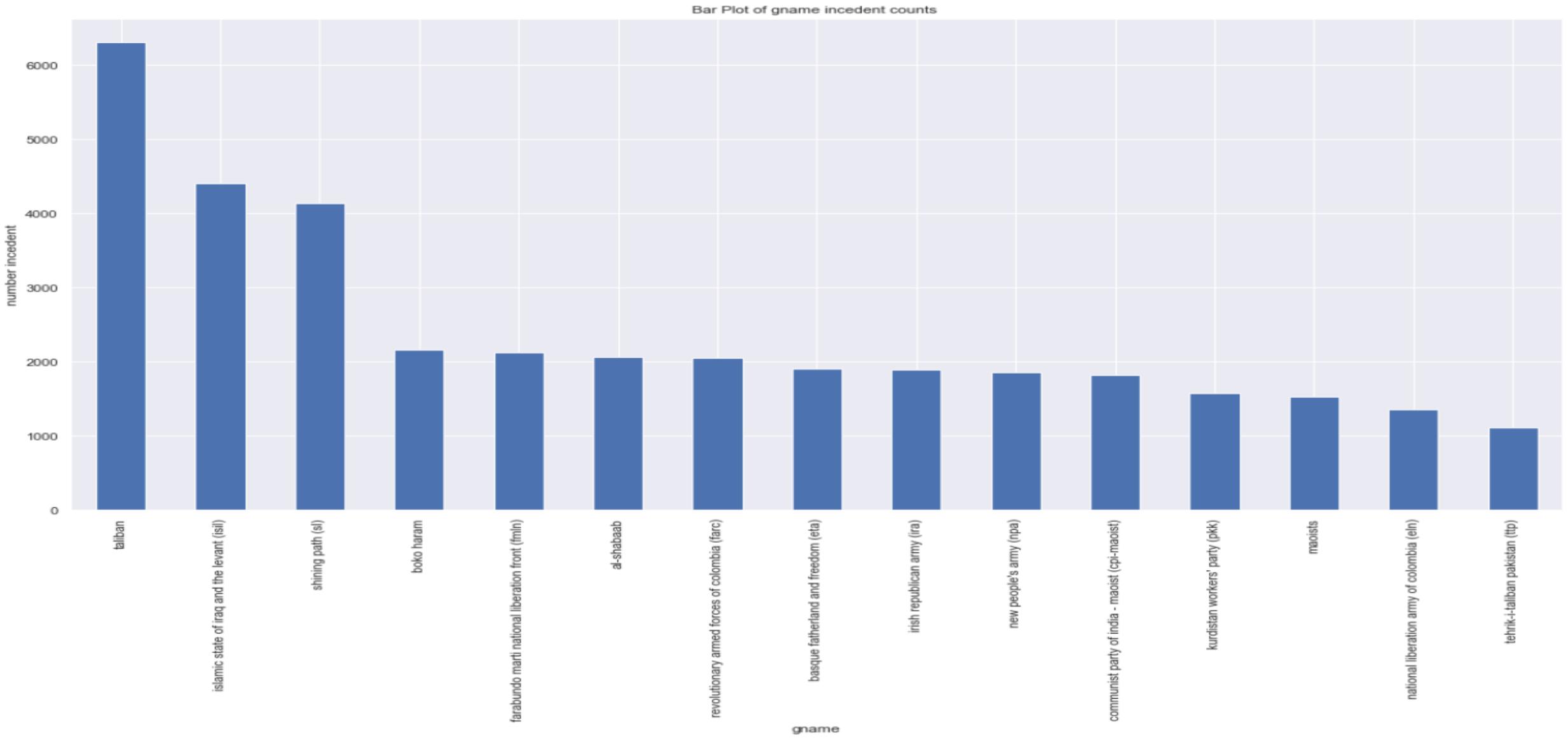
Exploratory Analysis



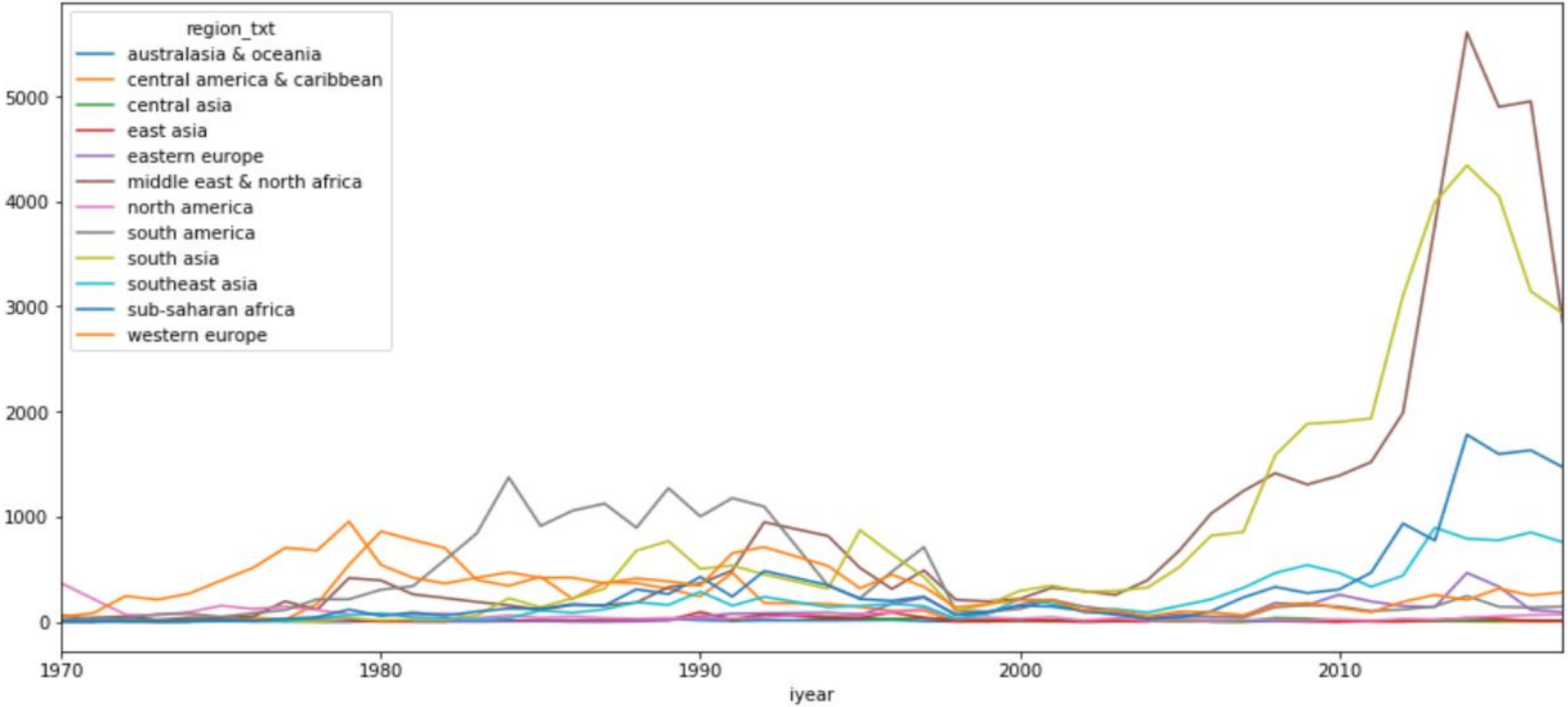
Exploratory Analysis



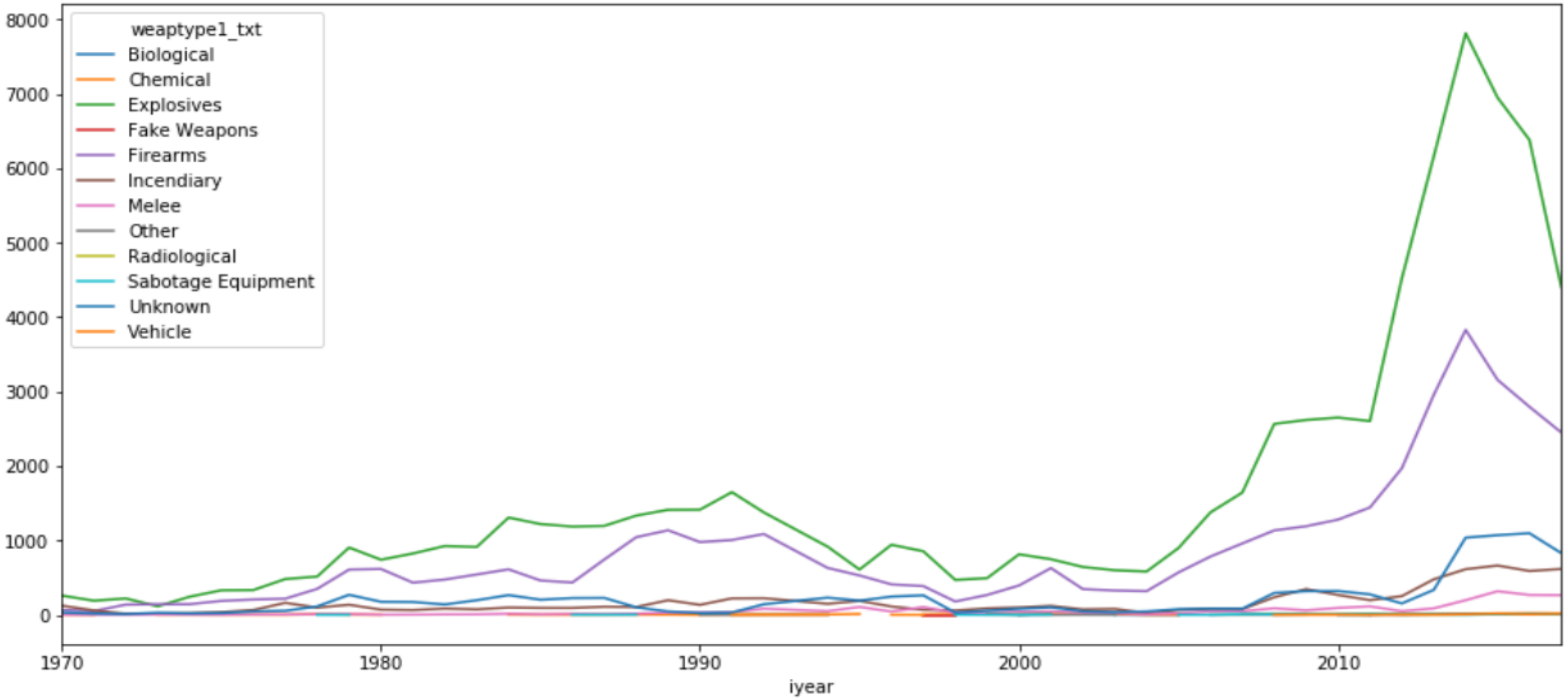
Exploratory Analysis



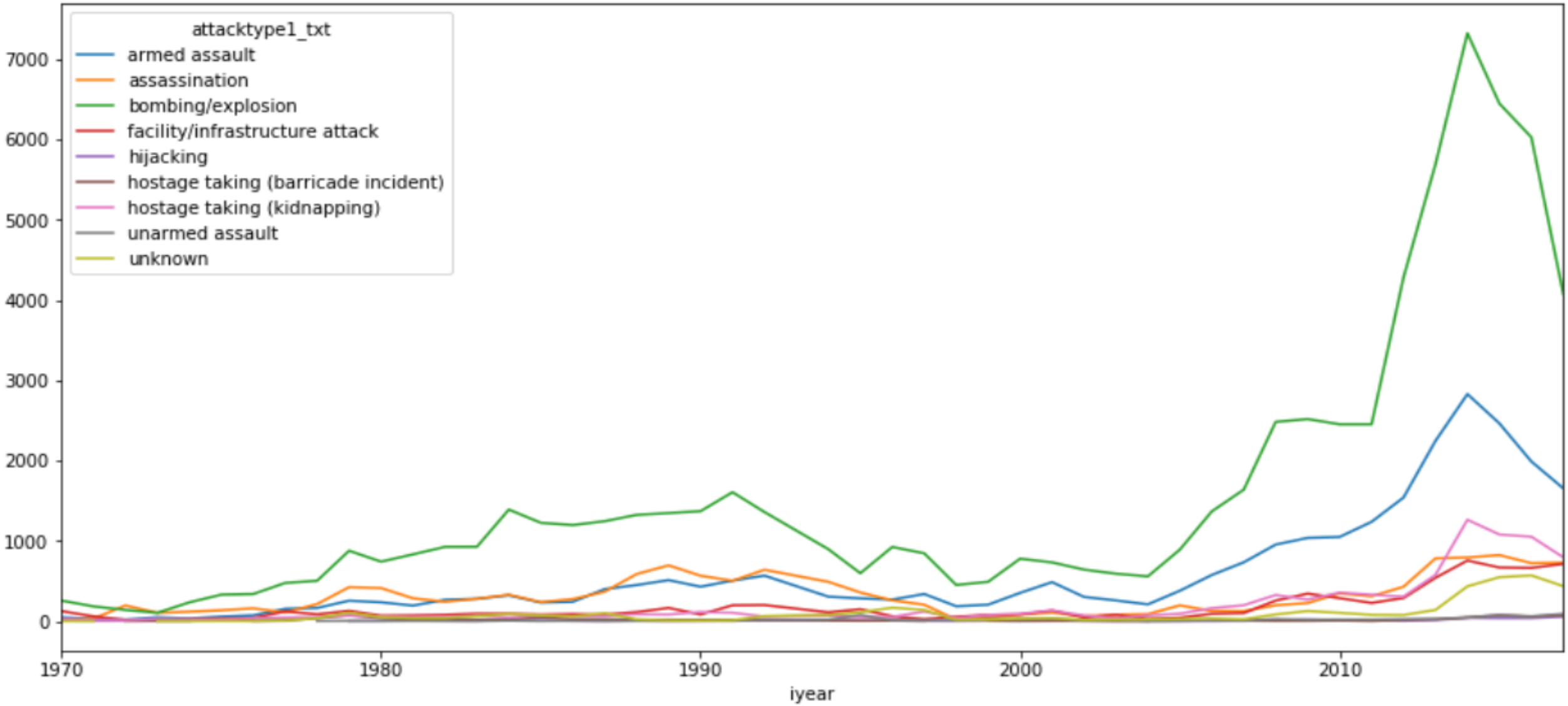
Exploratory Analysis



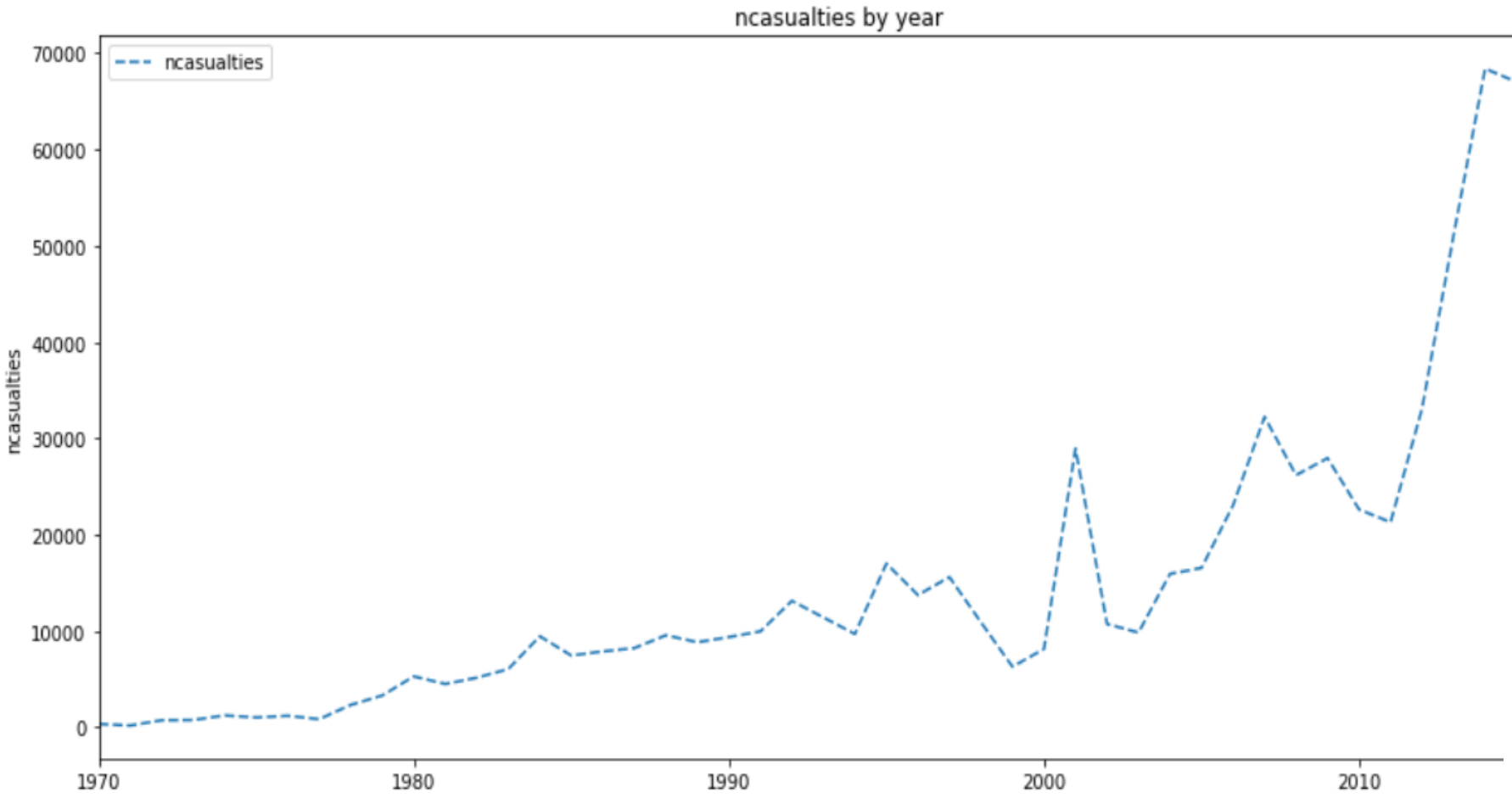
Exploratory Analysis



Exploratory Analysis



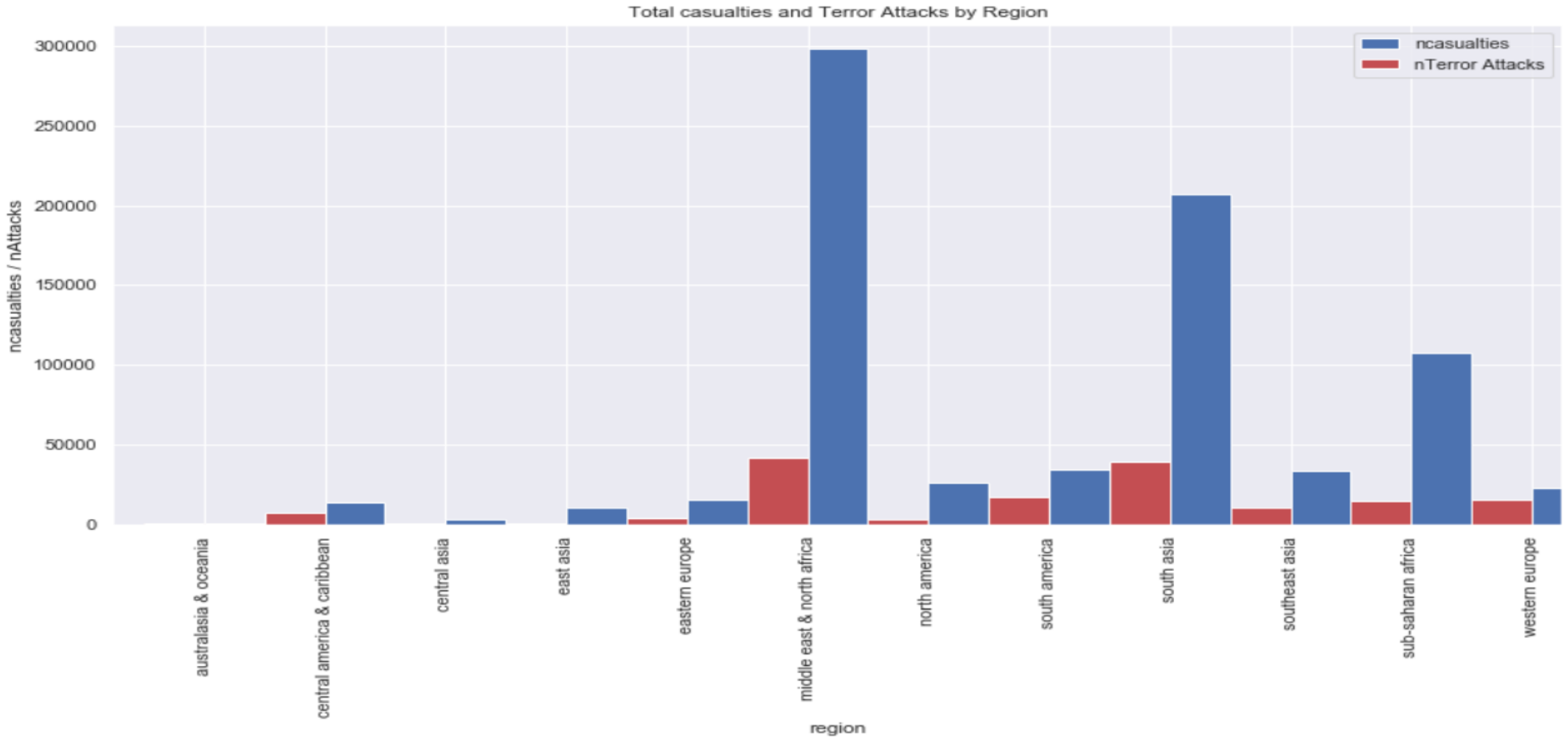
Exploratory Analysis



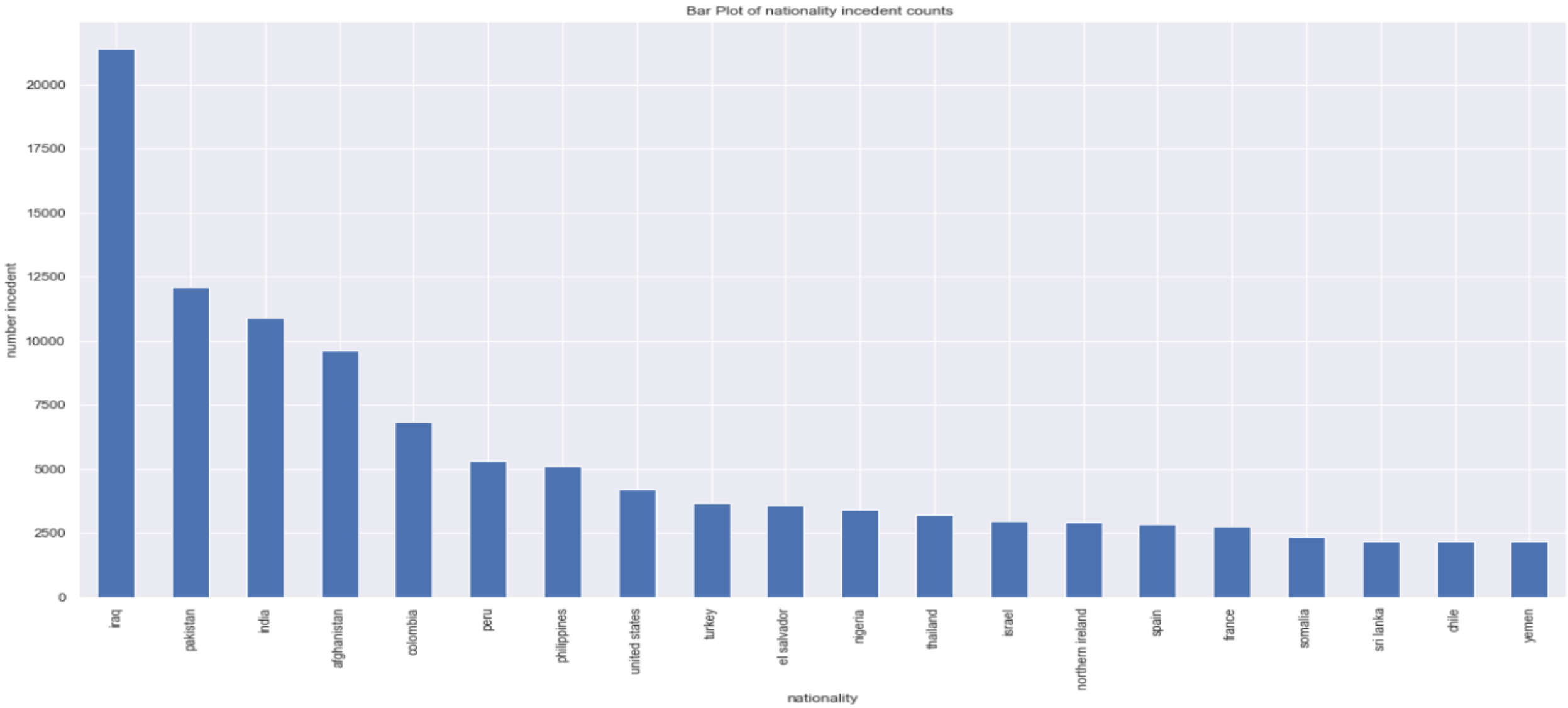
iyear	imonth	iday	extended	summary	scite1	scite2	scite3	dbsource	ncasualties
2001	9	11	0	09/11/2001: this was one of four related attac...	United States Government, The 9/11 Commission ...	Lindsay Kines, □United States on high alert af...	Joe Frolick, □Hijackers Ram Two Airliners Into...	CETIS	9574

iyear	imonth	iday	extended	summary	scite1	scite2	scite3	dbsource	ncasualties
2014	8	3	1	08/03/2014: assailants attacked yizidi civilia...	"Freed From ISIS, Yazidi Women Return in 'Seve...	"Dozens of Yazidi women 'sold into marriage' b...	"WATCH: Yazidi sisters reunite after three yea...	START Primary Collection	953
2014	4	15	0	04/15/2014: assailants attacked a mosque being...	"UN confirms hundreds slaughtered in S.Sudan a...	"S.Sudan Rebels Slaughter 'Hundreds' In Ethnic...	"UN Reveals 353 Killed in South Sudan Attacks,...	START Primary Collection	687
2014	6	10	0	06/10/2014: assailants stormed badush prison i...	"Jihadists seize Iraq's Nineveh province," Age...	"Rebels seize control of Iraq's Nineveh," Al J...	"UN condemns mass executions by Islami...	START Primary Collection	670
2014	8	15	1	08/15/2014: assailants attacked yizidi civilia...	"Freed From ISIS, Yazidi Women Return in 'Seve...	"ISIS strikes Iraq village, kills Yazidis, off...	"Islamic State group kills 80 men," Himalayan ...	START Primary Collection	400
2014	11	28	0	11/28/2014: two suicide bombers and a roadside...	"Nigeria: Kano mosque blasts death toll above ...	"Nigerian mosque attack death toll climbs ...	"At Least 120 Dead In Nigeria Mosque Suicide A...	START Primary Collection	392

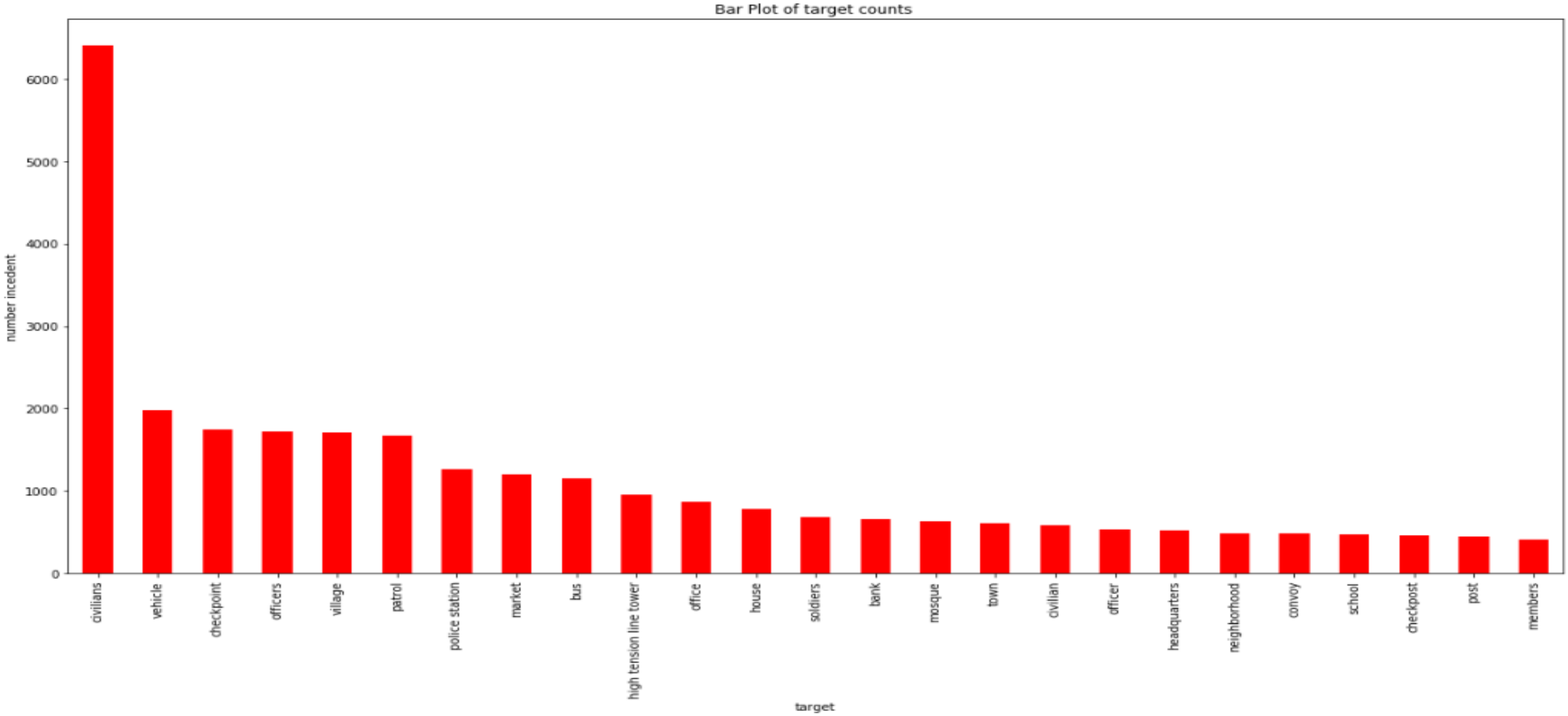
Exploratory Analysis



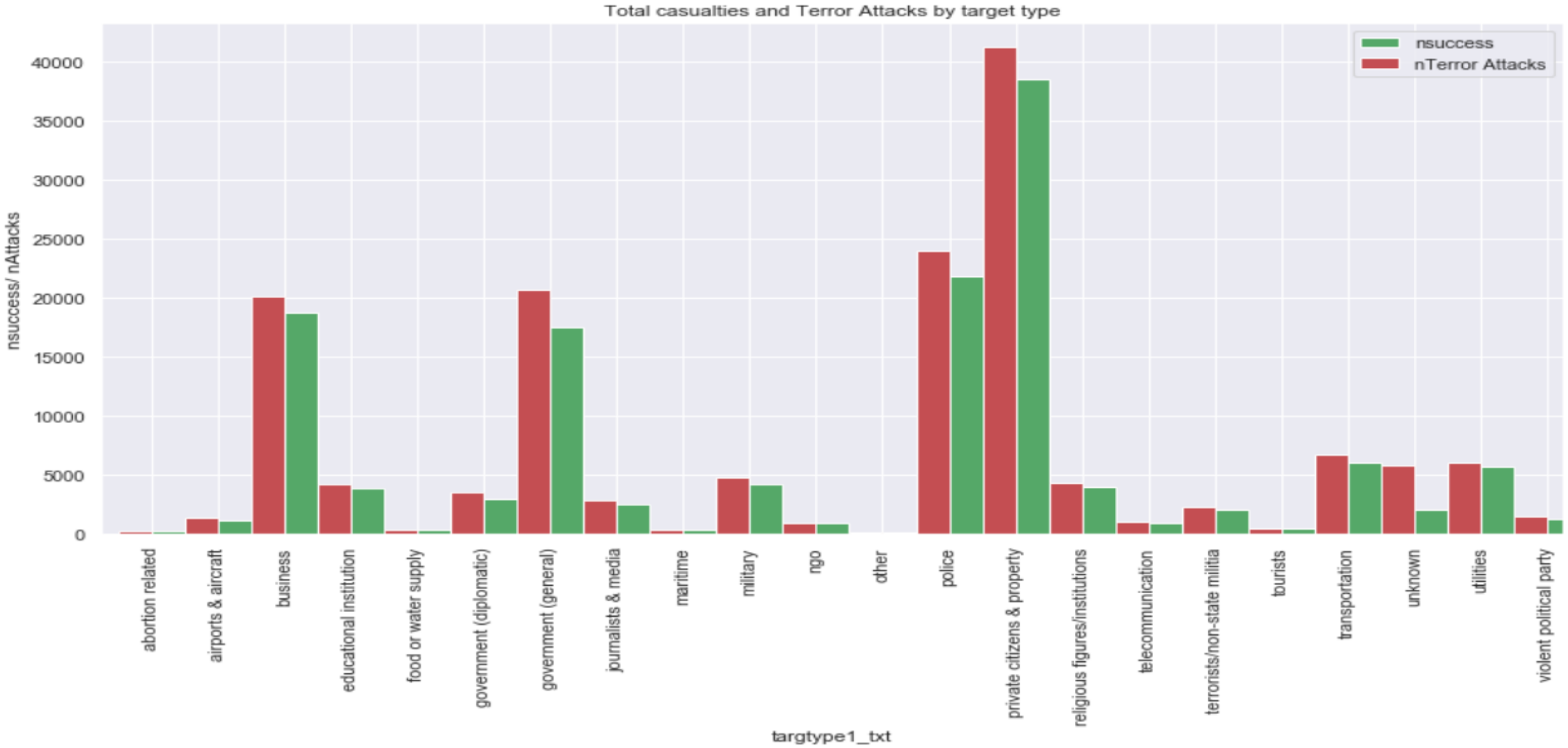
Exploratory Analysis



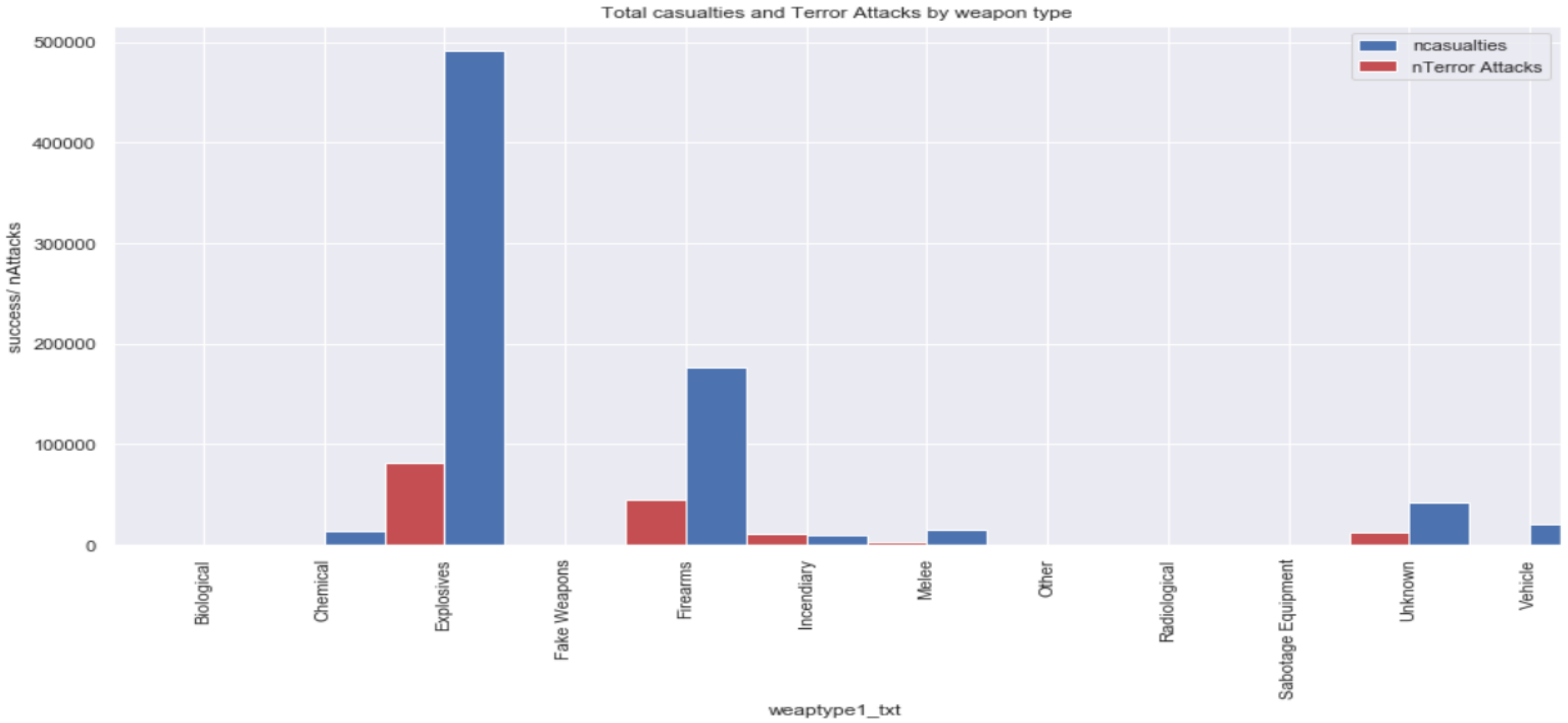
Exploratory Analysis



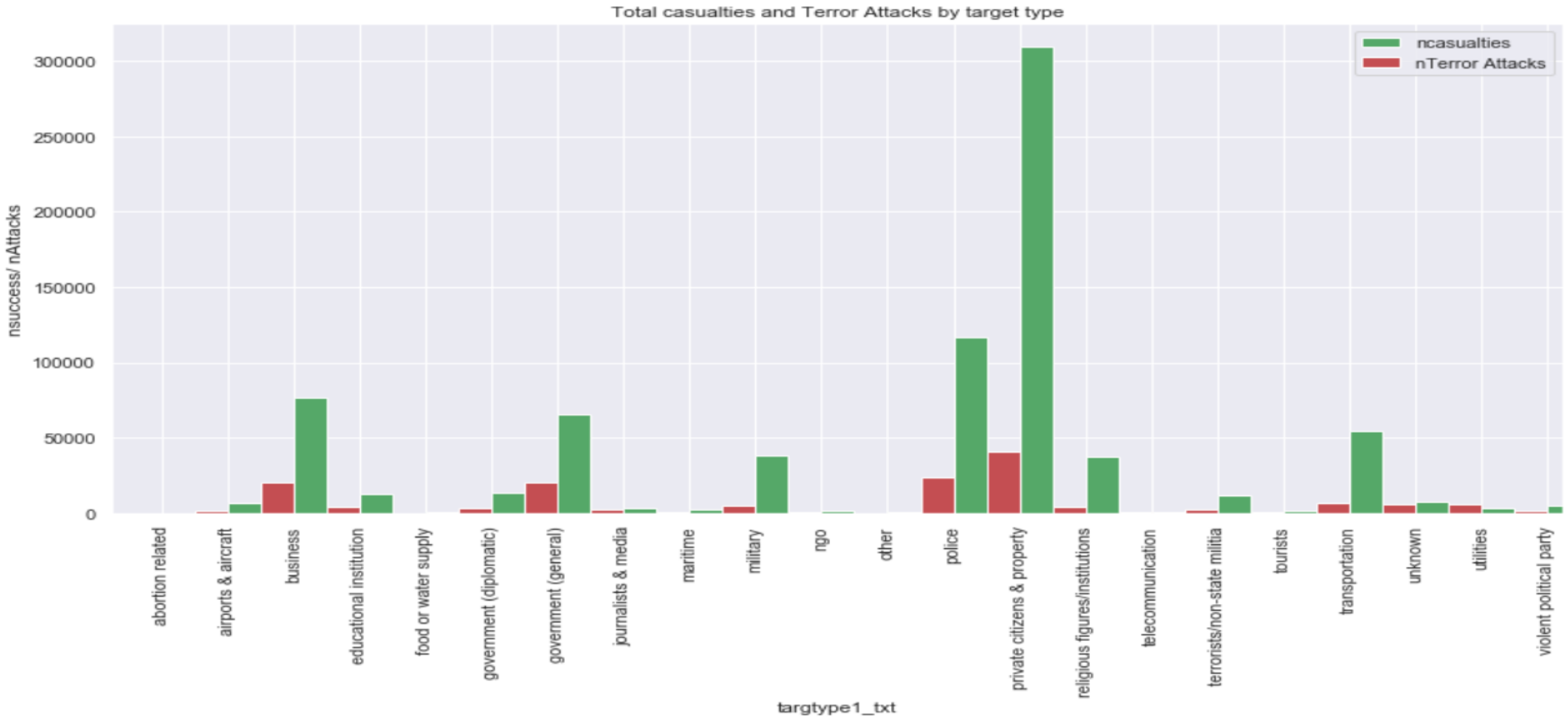
Exploratory Analysis



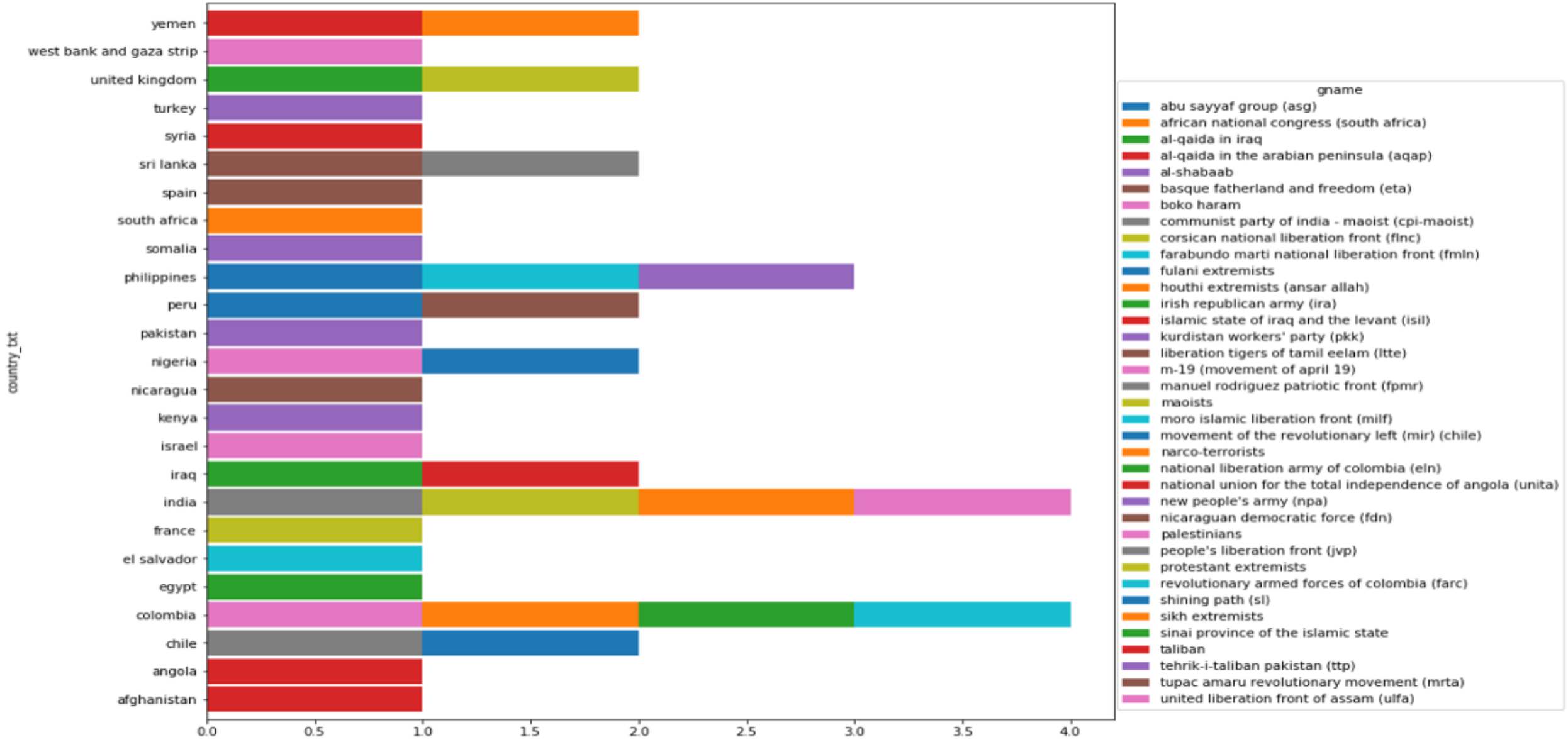
Exploratory Analysis



Exploratory Analysis

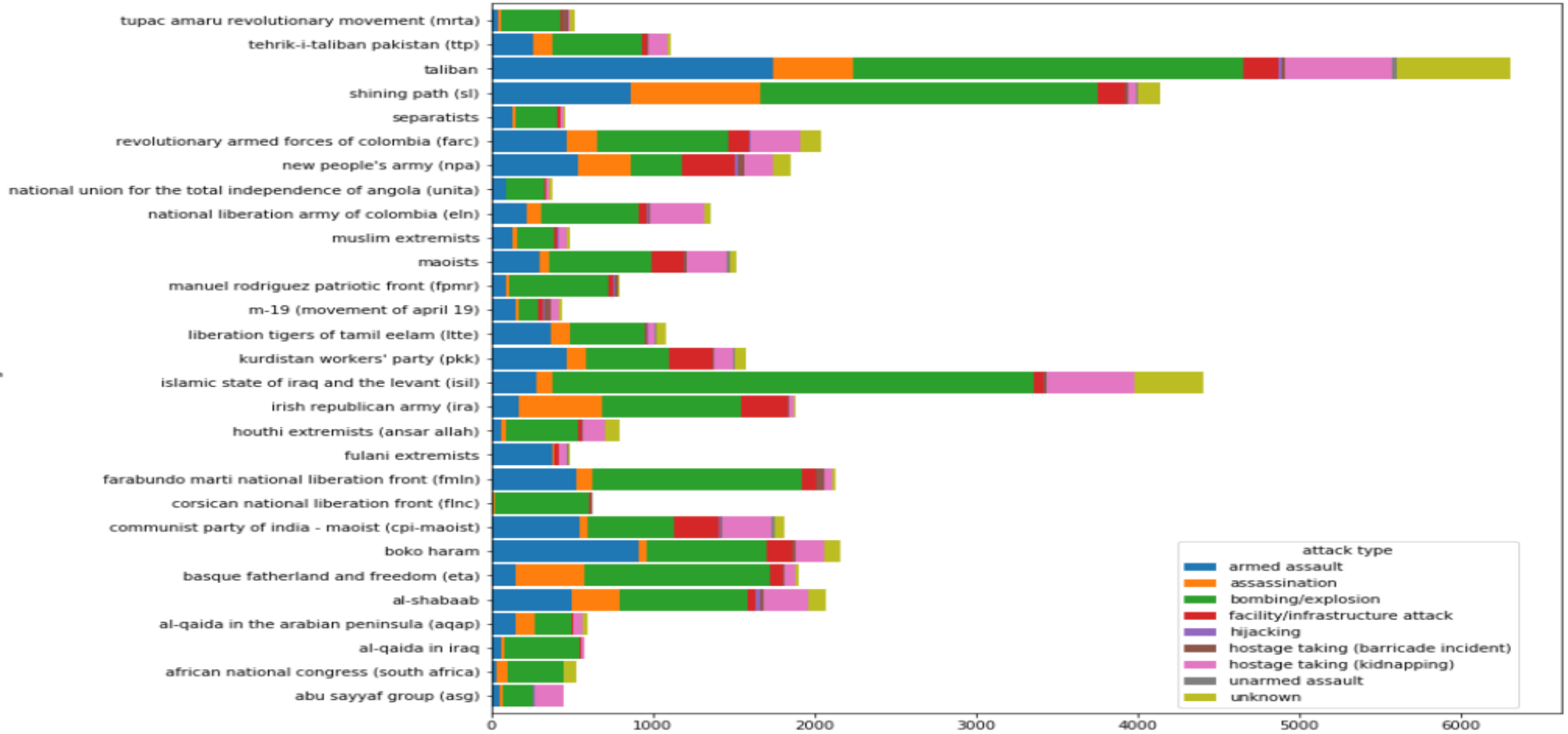


Exploratory Analysis : Terrorism group signature



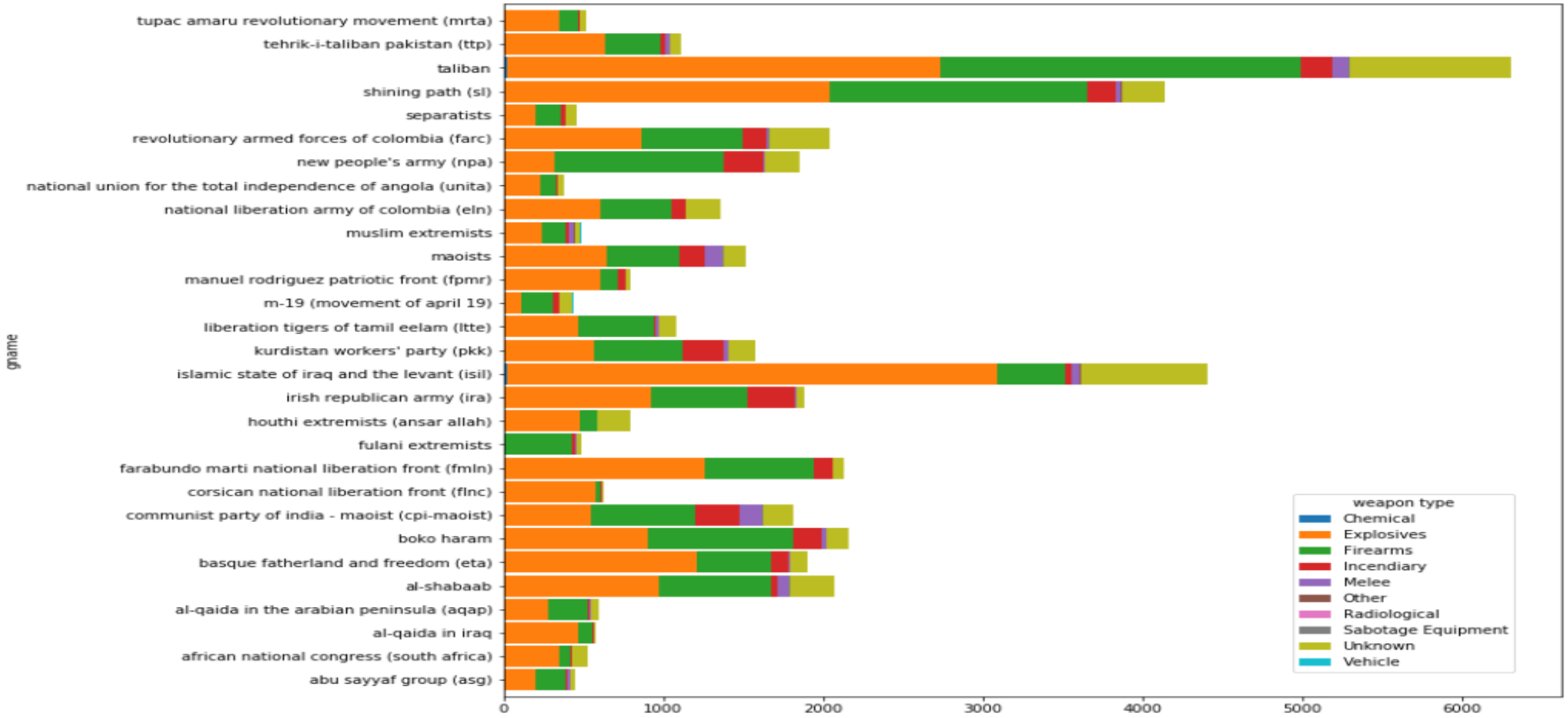
Exploratory Analysis: Terrorism group signature

```
pd.crosstab(mydatatemp1['gname'],mydatatemp1['attacktype1_txt']).plot.barh(stacked=True,figsize=(12,12),width=0.9)
plt.legend(loc=9,bbox_to_anchor=(0.8,0.25),title='attack type')
```



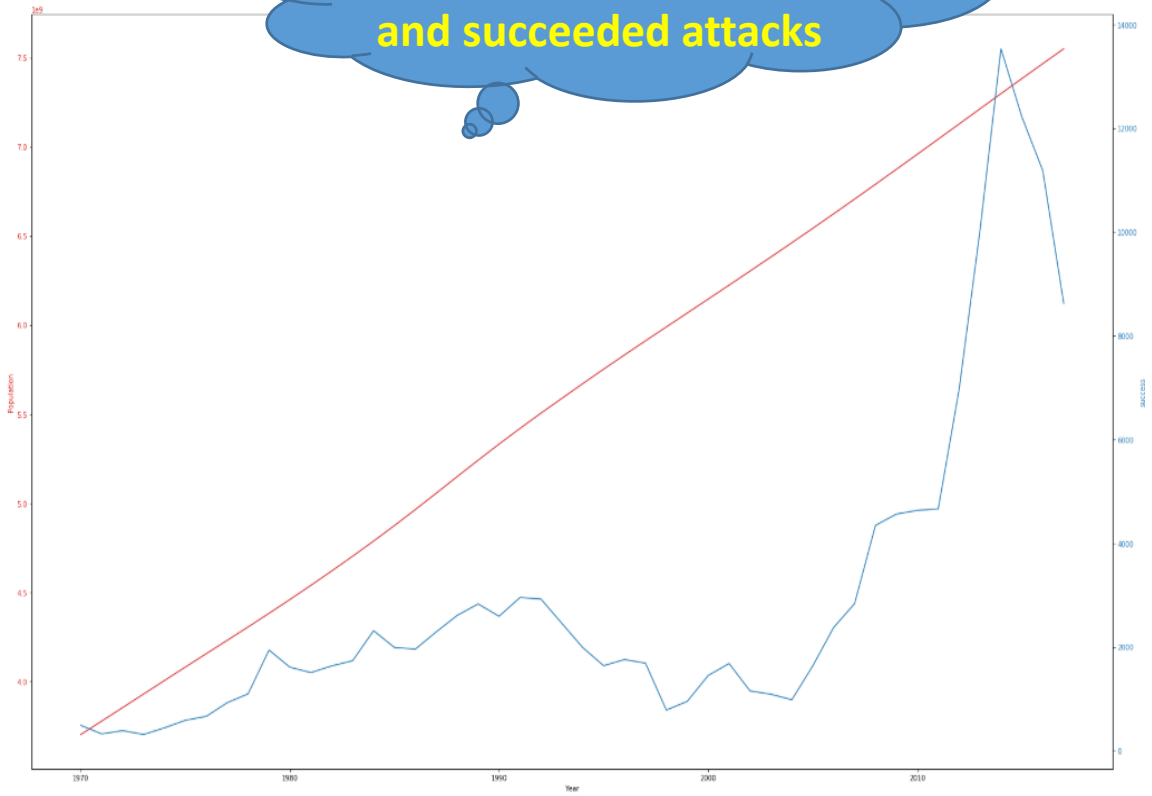
Exploratory Analysis: Terrorism group signature

```
mydatatemp1=mydatatemp[mydatatemp['gname'].isin(mydatatemp['gname'].value_counts()[1:30].index)]
pd.crosstab(mydatatemp1['gname'],mydatatemp1['weaptype1_txt']).plot.barh(stacked=True,figsize=(12,12),width=0.9)
plt.legend(loc=9,bbox_to_anchor=(0.85,0.3) , title='weapon type')
```



Exploratory Analysis: Correlation analysis using external datasets

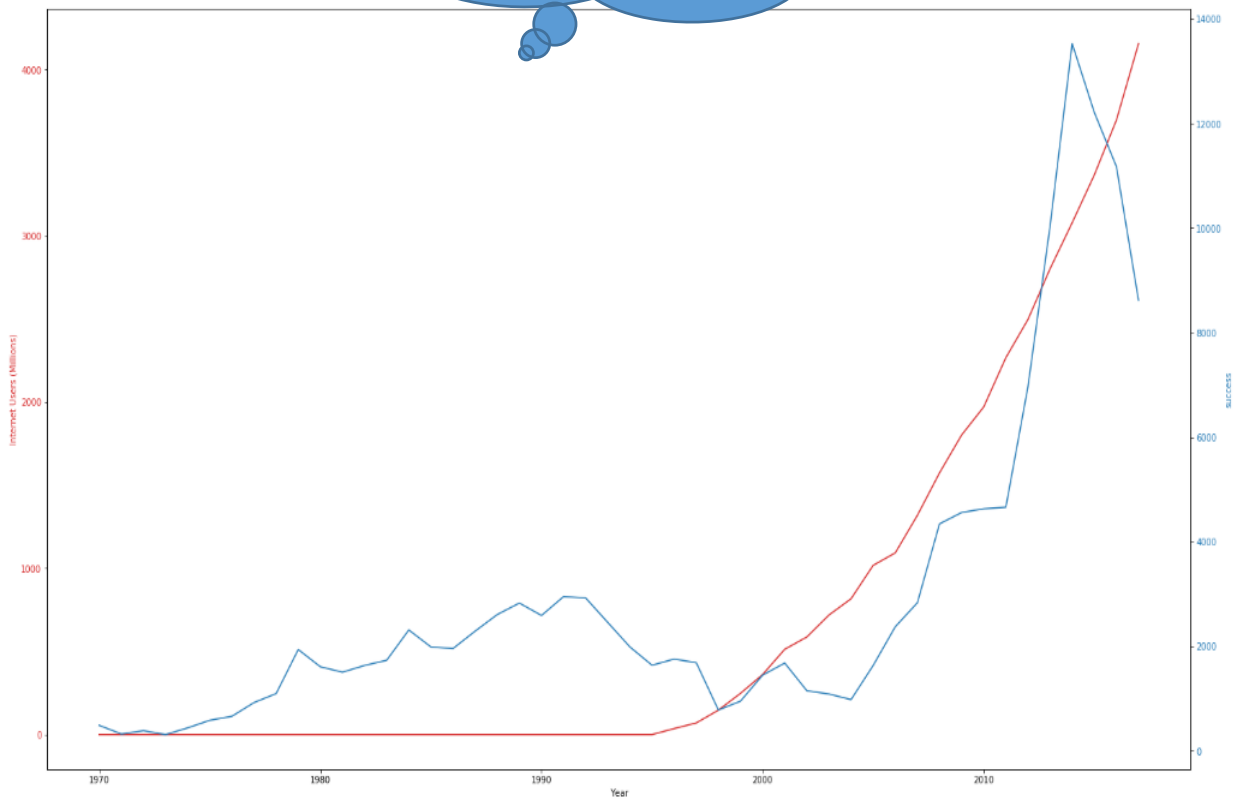
Correlation between population growth and succeeded attacks




```
from scipy.stats import pearsonr
corr, p = pearsonr(pop['World_Population'] , TS['success'])
corr
```

```
0.6996405274523172
```

Correlation between Internet user growth and succeeded attacks




```
from scipy.stats import pearsonr
corr, p = pearsonr(inter['users_millions'] , TS['success'])
corr
```

```
0.8917587261182954
```

Approach:

Dimensionality Reduction

initial Analysis

- **Univariate Analysis** (detailed data dictionary / list all variables and understand what each variable mean and represent)(what is the source of each variable directly measured or calculated based on other variables or does the variable has a time domain associated) - (Decide on the dependent (target) variable) (Assigning the correct data types and appropriate column names)(for numeric attributes check the 5 number summary min 1sQ mean 3rdQ max)(for categorical attributes check and decide on the levels , frequency) (check and deal with data inconsistencies , missing values , errors , duplicates , outliers (boxplots) , numeric signs , upper and lower cases , spaces or special characters in striges) (check distributions of the variables : Normal distribution?)(Low variance filter)(check the imbalance in the dependent variable) (check time variable) (Univariate visualizations)
- **BiVariate Analysis** (pairwise relations) (pairwise visuliazations like scatter plots) (correlation analysis spearman or pearson).
- **Multivariate Analysis** (relations between more than 2 variables)(statistical tools such as one way ANOVA analysis or rank or to compare the means.

Exploratory Analysis

- Normalizing / scaling
- Sub-setting the data
- Decision rules , association rules , n grams
- Clustering such as K-mean
- Hypothesise analysis
- Correlation analysis
- NLP analysis for some text attributes

Dimensionality Reduction

- remove attributes with too many missing values
- remove attributes with zero or very low variance
- remove one of the attributes with high correlations with other - prefer the one with more missing values or lower variance
- Feature selection(decide on the importance of the attribute using statistical measures like information gain or Gini index) (Forward selection and backward elimination)

Experimental Desgin

- Randomizing, Splitting the data into training and test sets to training, validation and testing
- Treatment for imbalance (under-sampling the majority class and over-sampling the minority class)
- Cross validation such as 10 folds

Modeling

- building the Classification Models
- Train the model
- Validate the model

Evaluation

- Classification (Confusion matric , ROC (receiver operating characteristics) , Accuracy , Recall , Precision)
- compare performance between different models (contingency tables , multivariate analysis of variance)

Improving the Model

- deal with very high accuracy(Fixation variables , overfitting)
- deal with very low performance (many need new variables or more observations)
- iterate to improve the model performance

Conclusions

- Inferences , dissuasions
- Threats to validity (Internal , External , construct) and propose the solution that may mitigate these threats
- Next/ future steps

Dimensionality Reduction

Remove low or No Variance , and none relevant attributes

Remove high correlated attributes

```
mydata['crit1'].describe()
```

```
count    152622.0
mean          1.0
std           0.0
min           1.0
25%           1.0
50%           1.0
75%           1.0
max           1.0
Name: crit1, dtype: float64
```

```
mydata['crit2'].describe()
```

```
count    152622.0
mean          1.0
std           0.0
min           1.0
25%           1.0
50%           1.0
75%           1.0
max           1.0
Name: crit2, dtype: float64
```

```
mydata['crit3'].describe()
```

```
count    152622.0
mean          1.0
std           0.0
min           1.0
25%           1.0
50%           1.0
75%           1.0
max           1.0
Name: crit3, dtype: float64
```

```
# remove attributes USA spesefic which will not be used in this study
# 'nkillus' , 'nwoundus' , 'nhostkidus' , 'ransomamtus' , 'ransompaidus'
list1 = ['nkillus' , 'nwoundus' ]
```

```
# remove the additional infromation attributes which will not add value in this study
#'addnotes' , 'INT_LOG' , 'INT_IDEO' , 'INT_MISC', 'INT_ANY' , 'scite1' , 'scite2', 'scite3' ,'dbsource'
list1 = ['INT_LOG' , 'INT_IDEO' , 'INT_MISC', 'INT_ANY' , 'scite1' , 'scite2', 'dbsource']
```

```
# Save the clean dataframe
mydata.to_csv('../code/mydata_clean2.csv' , index= False)
```

```

```

```
#Here start to  Read the clean dataframe
mydata = pd.read_csv('../code/mydata_clean2.csv', encoding='ISO-8859-1')
```

```
mydata.shape
```

```
(152622, 30)
```

Dimensionality Reduction
feature selection / importance

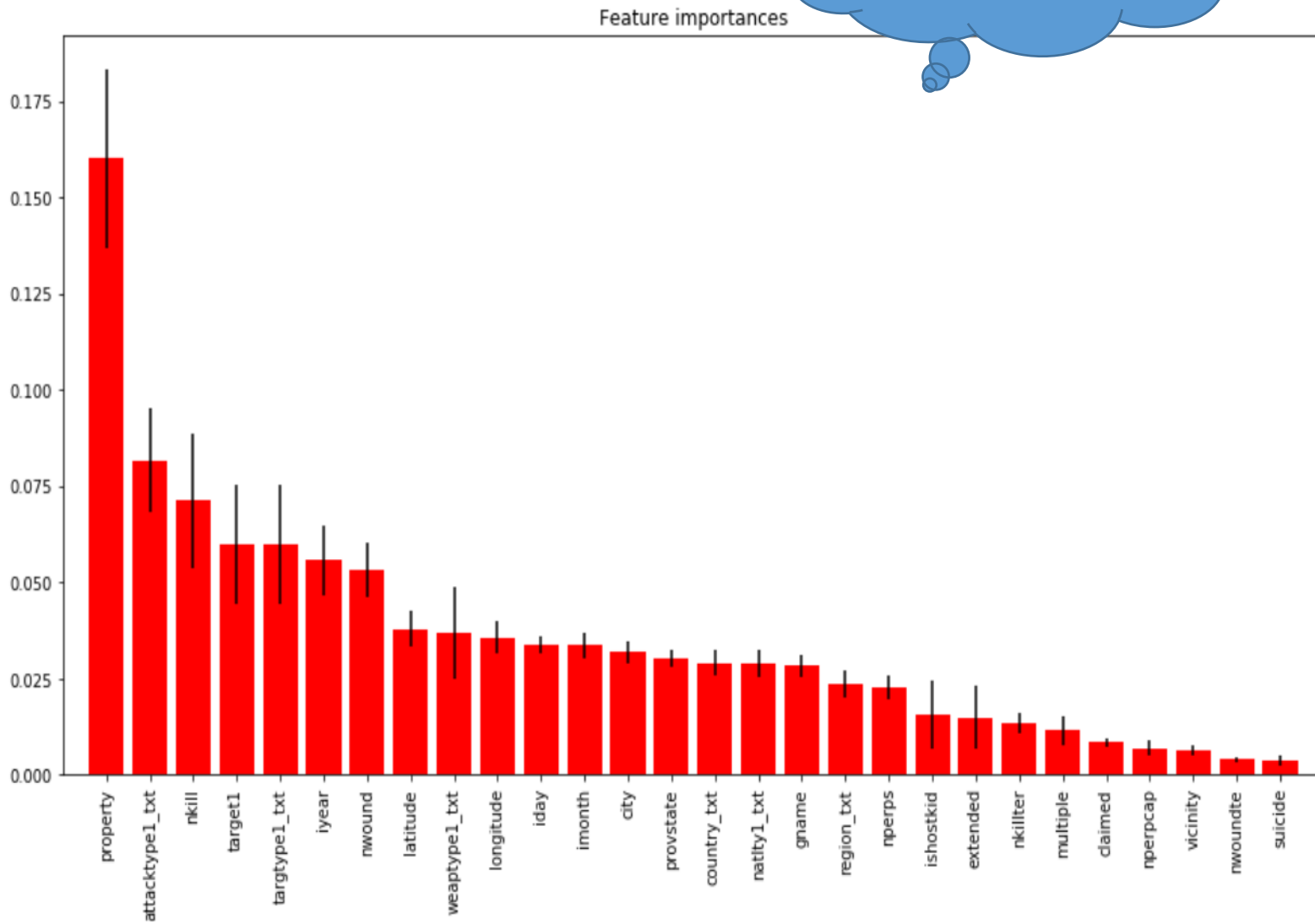
Features importance and Selection

```
lb = LabelEncoder()
```

```
mydata['country_txt'] = lb.fit_transform(mydata['country_txt'])
mydata['region_txt'] = lb.fit_transform(mydata['region_txt'])
mydata['city'] = lb.fit_transform(mydata['city'])
mydata['provstate'] = lb.fit_transform(mydata['provstate'])
mydata['attacktype1_txt'] = lb.fit_transform(mydata['attacktype1_txt'])
mydata['targtype1_txt'] = lb.fit_transform(mydata['targtype1_txt'])
mydata['weaptype1_txt'] = lb.fit_transform(mydata['weaptype1_txt'])
mydata['natlty1_txt'] = lb.fit_transform(mydata['natlty1_txt'])
mydata['gname'] = lb.fit_transform(mydata['gname'])
mydata['target1'] = lb.fit_transform(mydata['target1'])
```

```
importances = forest.feature_importances_
std = np.std([tree.feature_importances_ for tree in for
```

```
feature_cols = [
    'iyear', 'imonth', 'iday',
    #'extended',
    #'multiple',
    'country_txt','region_txt', 'provstate', 'city', #'vicinity',
    'latitude', 'longitude',
    'attacktype1_txt',
    'weaptype1_txt',
    'targtype1_txt', 'target1',
    'nperps', 'nperpcap', 'nkillter', #'nwoundte',
    'claimed',  'gname',
    'nkill',  'nwound', 'natlty1_txt',
    #'suicide',
    'property', #'ishostkid'
]
```



Approach:

experimental design

initial Analysis

- **Univariate Analysis** (detailed data dictionary / list all variables and understand what each variable mean and represent)(what is the source of each variable directly measured or calculated based on other variables or does the variable has a time domain associated) - (Decide on the dependent (target) variable) (Assigning the correct data types and appropriate column names)(for numeric attributes check the 5 number summary min 1sQ mean 3rdQ max)(for categorical attributes check and decide on the levels , frequency) (check and deal with data inconsistencies , missing values , errors , duplicates , outliers (boxplots) , numeric signs , upper and lower cases , spaces or special characters in striges) (check distributions of the variables : Normal distribution?)(Low variance filter)(check the imbalance in the dependent variable) (check time variable) (Univariate visualizations)
- **BiVariate Analysis** (pairwise relations) (pairwise visuliazations like scatter plots) (correlation analysis spearman or pearson).
- **Multivariate Analysis** (relations between more than 2 variables)(statistical tools such as one way ANOVA analysis or rank or to compare the means.

Exploratory Analysis

- Normalizing / scaling
- Sub-setting the data
- Decision rules , association rules , n grams
- Clustering such as K-mean
- Hypothesise analysis
- Correlation analysis
- NLP analysis for some text attributes

Dimensionality Reduction

- remove attributes with too many missing values
- remove attributes with zero or very low variance
- remove one of the attributes with high correlations with other - prefer the one with more missing values or lower variance
- Feature selection(decide on the importance of the attribute using statistical measures like information gain or Gini index) (Forward selection and backward elimination)

Experimental Desgin

- Randomizing, Splitting the data into training and test sets to training, validation and testing
- Treatment for imbalance (under-sampling the majority class and over-sampling the minority class)
- Cross validation such as 10 folds

Modeling

- building the Classification Models
- Train the model
- Validate the model

Evaluation

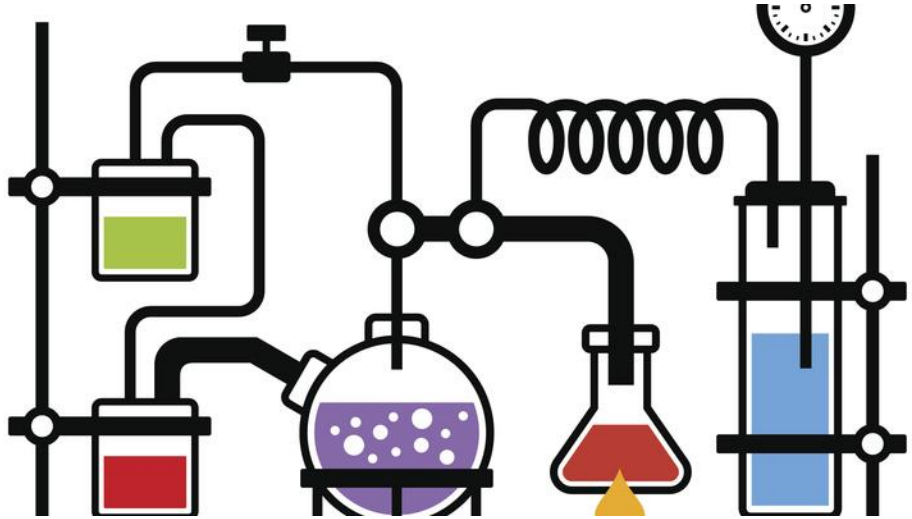
- Classification (Confusion matric , ROC (receiver operating characteristics) , Accuracy , Recall , Precision)
- compare performance between different models (contingency tables , multivariate analysis of variance)

Improving the Model

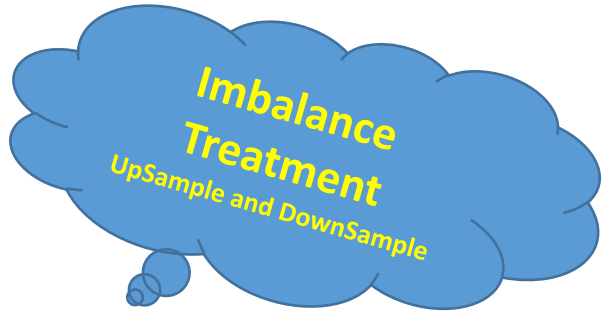
- deal with very high accuracy(Fixation variables , overfitting)
- deal with very low performance (many need new variables or more observations)
- iterate to improve the model performance

Conclusions

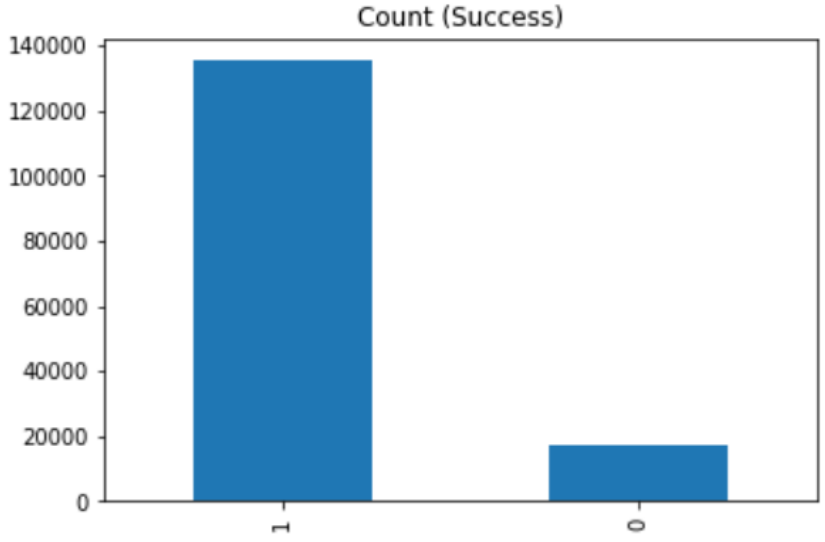
- Inferences , dissuasions
- Threats to validity (Internal , External , construct) and propose the solution that may mitigate these threats
- Next/ future steps



Experimental Design
Class Imbalance treatment



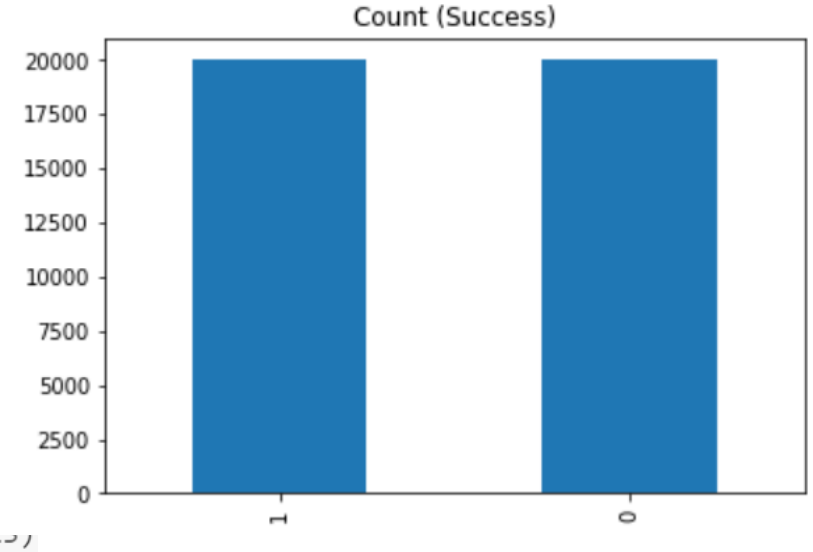

```
Class 0: 17335
Class 1: 135287
Proportion: 0.13  : 1
```




```
# Downsample majority class
df_majority_downsampled = resample(df_majority,
                                   replace=False,
                                   n_samples=20000
                                   random_state=12
```

```
# Upsample minority class
df_minority_upsampled = resample(df_minority,
                                 replace=True,
                                 n_samples=2000
                                 random_state=12
```

```
Class 0: 20000
Class 1: 20000
Proportion: 1.0  : 1
```



Experimental Design
Train Validation test split

Train 60%
Validation 20%
Test 20%

```
target_col =  target_col
```

```
X = mydata[feature_cols].fillna(0)
y = mydata[target_col]
```

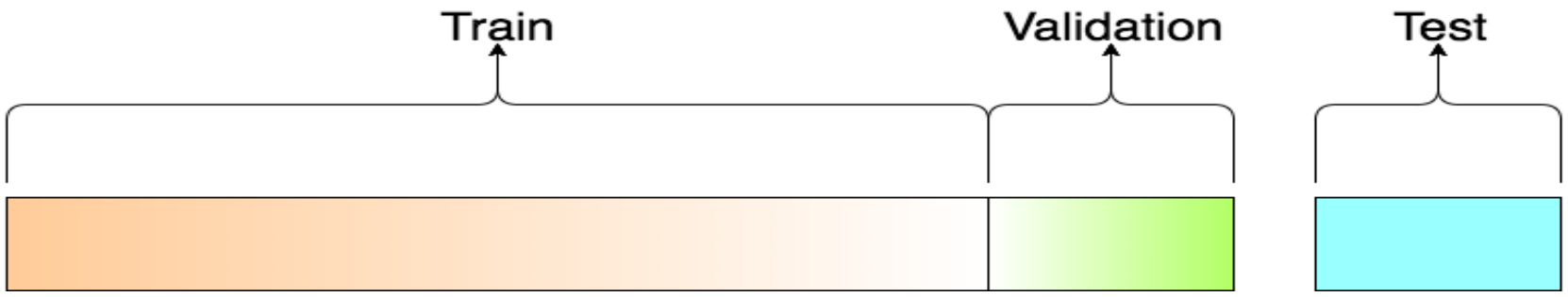
```
X.shape
```

```
(40000, 22)
```

```
y.shape
```

```
(40000,)
```

```
X_train, X_vald_test, y_train, y_vald_test = train_test_split(X, y, test_size=0.4 , shuffle = True)
X_vald, X_test, y_vald, y_test = train_test_split(X_vald_test, y_vald_test, test_size=0.5 , shuffle = True
)
```




```
X_train.shape
```

```
(24000, 22)
```

```
X_vald.shape
```

```
(8000, 22)
```

```
X_test.shape
```

```
(8000, 22)
```

```
y_train.shape
```

```
(24000,)
```

```
y_vald.shape
```

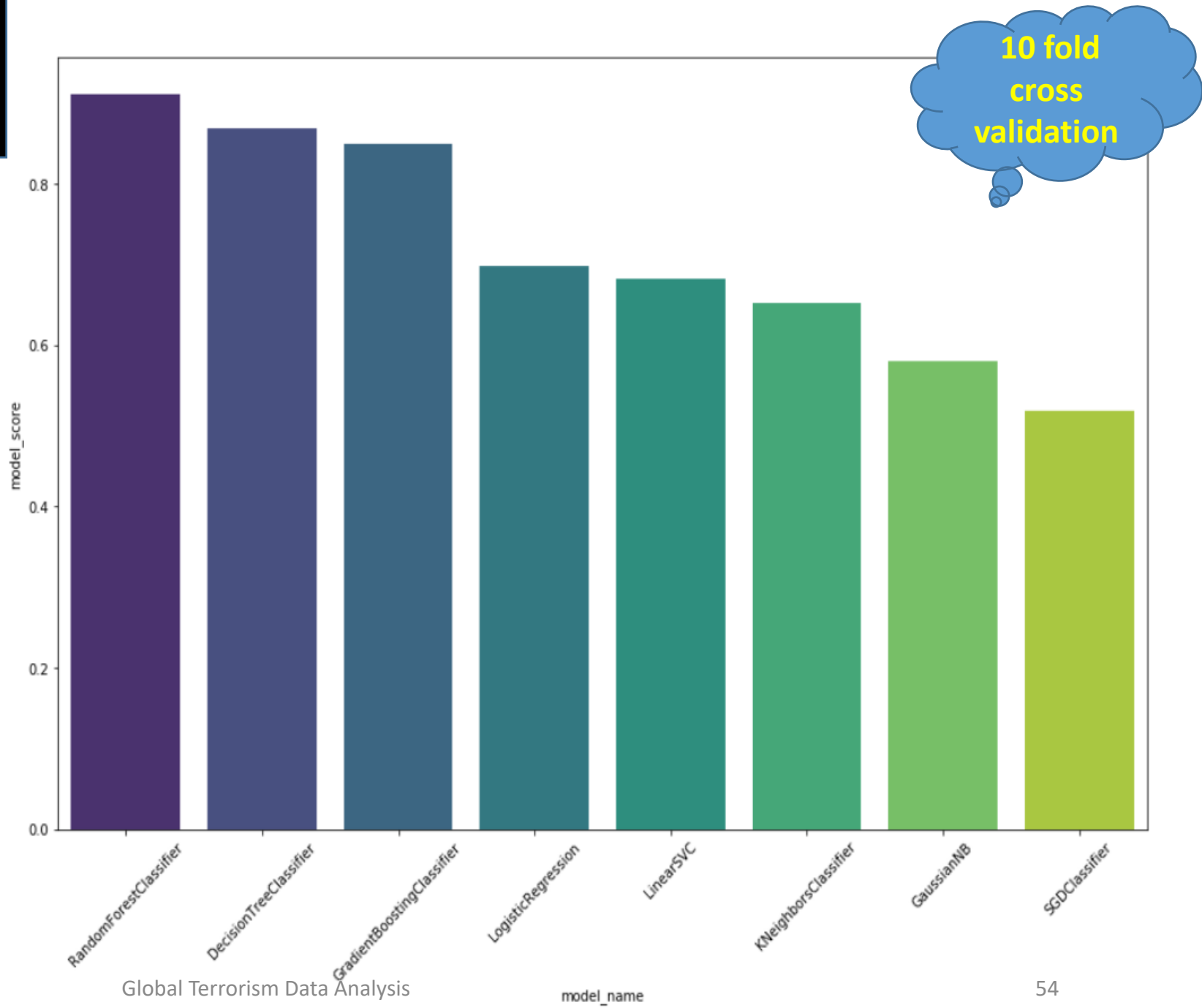
```
(8000,)
```

```
y_test.shape
```

```
(8000,)
```

Experimental Design
Model Selection

model_name	model_score
RandomForestClassifier	0.911375
DecisionTreeClassifier	0.869791
GradientBoostingClassifier	0.849875
LogisticRegression	0.697792
LinearSVC	0.682292
KNeighborsClassifier	0.652416
GaussianNB	0.580125
SGDClassifier	0.518292



Statistical tests
between top 3 models

model_name
RandomForestClassifier
DecisionTreeClassifier
GradientBoostingClassifier

```
# One Way ANOVA
#H0 the null hypothesis: u1=u2=u3 (u is the mean).
#Ha the alternative hypothesis is : at least one u is diffrent.
```

ANOVA

```
1  scores1
```

```
array([0.92958333, 0.91833333, 0.92666667, 0.92458333, 0.92541667,
       0.9275    , 0.92083333, 0.925     , 0.92458333, 0.93041667])
```

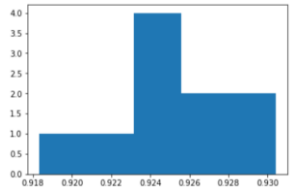
```
1  scores2
```

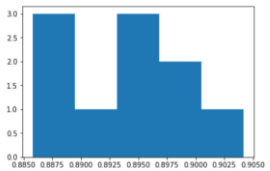
```
array([0.90416667, 0.89      , 0.89375   , 0.88625   , 0.89916667,
       0.89666667, 0.89458333, 0.88916667, 0.88583333, 0.89875   ])
```

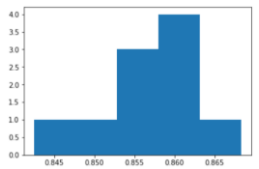
```
1  scores3
```

```
array([0.86041667, 0.8425    , 0.8575    , 0.85125   , 0.85291667,
       0.86041667, 0.85833333, 0.855     , 0.86833333, 0.85958333])
```

check normal distribution?







```
1  #statistic:The computed F-value of the test & pvalue:The associated p-value from the F-dist
2  stats.f_oneway(scores1 , scores2 , scores3)
```

```
F_onewayResult(statistic=366.72655969344885, pvalue=2.6849289583133914e-20)
```

```
1  #statistic:The computed F-value of the test & pvalue:The associated p-value from the F-distribu
2  stats.f_oneway(scores1 , scores2)
```

```
F_onewayResult(statistic=199.33267280568782, pvalue=3.535364163200079e-11)
```

```
1  #statistic:The computed F-value of the test & pvalue:The associated p-value from the F-distribution
2  stats.f_oneway(scores1 , scores3)
```

```
F_onewayResult(statistic=780.1830833067391, pvalue=2.8265598862131117e-16)
```

```
1  #statistic:The computed F-value of the test & pvalue:The associated p-value from the F-distribution
2  stats.f_oneway(scores2 , scores3)
```

```
F_onewayResult(statistic=166.3469926990382, pvalue=1.566720886717172e-10)
```

Reject the null hypothesis H0, U1 != U2 !=U3 there are statistically significant differences between the groups.

Approach:

Modeling & Evaluation

initial Analysis

- **Univariate Analysis** (detailed data dictionary / list all variables and understand what each variable mean and represent)(what is the source of each variable directly measured or calculated based on other variables or does the variable has a time domain associated) - (Decide on the dependent (target) variable) (Assigning the correct data types and appropriate column names)(for numeric attributes check the 5 number summary min 1sQ mean 3rdQ max)(for categorical attributes check and decide on the levels , frequency) (check and deal with data inconsistencies , missing values , errors , duplicates , outliers (boxplots) , numeric signs , upper and lower cases , spaces or special characters in striges) (check distributions of the variables : Normal distribution?)(Low variance filter)(check the imbalance in the dependent variable) (check time variable) (Univariate visualizations)
- **BiVariate Analysis** (pairwise relations) (pairwise visuliazations like scatter plots) (correlation analysis spearman or pearson).
- **Multivariate Analysis** (relations between more than 2 variables)(statistical tools such as one way ANOVA analysis or rank or to compare the means.

Exploratory Analysis

- Normalizing / scaling
- Sub-setting the data
- Decision rules , association rules , n grams
- Clustering such as K-mean
- Hypothesise analysis
- Correlation analysis
- NLP analysis for some text attributes

Dimensionality Reduction

- remove attributes with too many missing values
- remove attributes with zero or very low variance
- remove one of the attributes with high correlations with other - prefer the one with more missing values or lower variance
- Feature selection(decide on the importance of the attribute using statistical measures like information gain or Gini index) (Forward selection and backward elimination)

Experimental Desgin

- Randomizing, Splitting the data into training and test sets to training, validation and testing
- Treatment for imbalance (under-sampling the majority class and over-sampling the minority class)
- Cross validation such as 10 folds

Modeling

- building the Classification Models
- Train the model
- Validate the model

Evaluation

- Classification (Confusion matric , ROC (receiver operating characteristics) , Accuracy , Recall , Precision)
- compare performance between different models (contingency tables , multivariate analysis of variance)

Improving the Model

- deal with very high accuracy(Fixation variables , overfitting)
- deal with very low performance (many need new variables or more observations)
- iterate to improve the model performance

Conclusions

- Inferences , dissuasions
- Threats to validity (Internal , External , construct) and propose the solution that may mitigate these threats
- Next/ future steps

Iterative Process

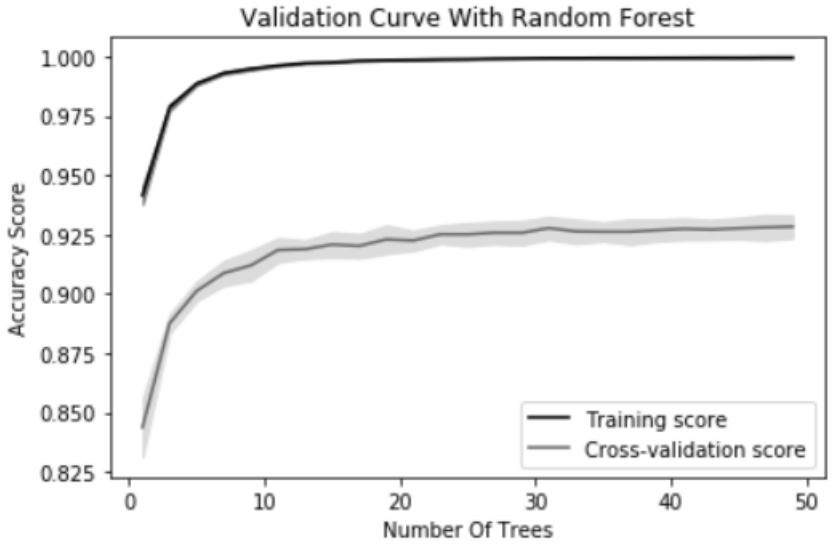
Improving the Models

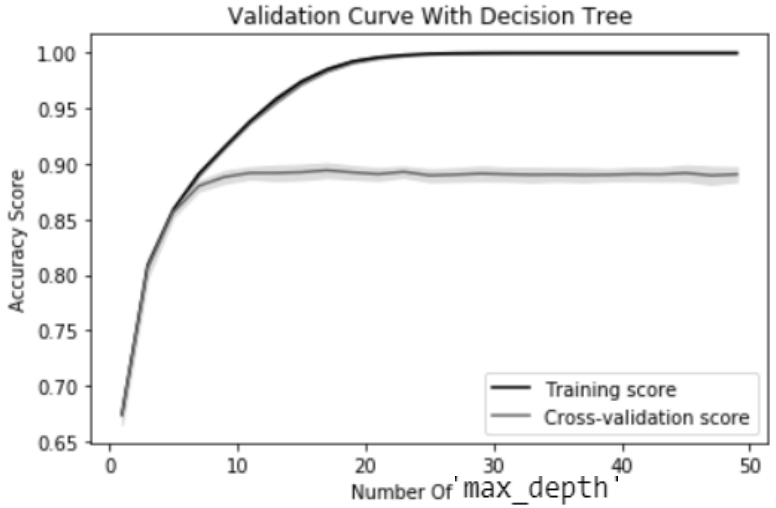
Model Evaluation, calibration and hyperparameters tuning

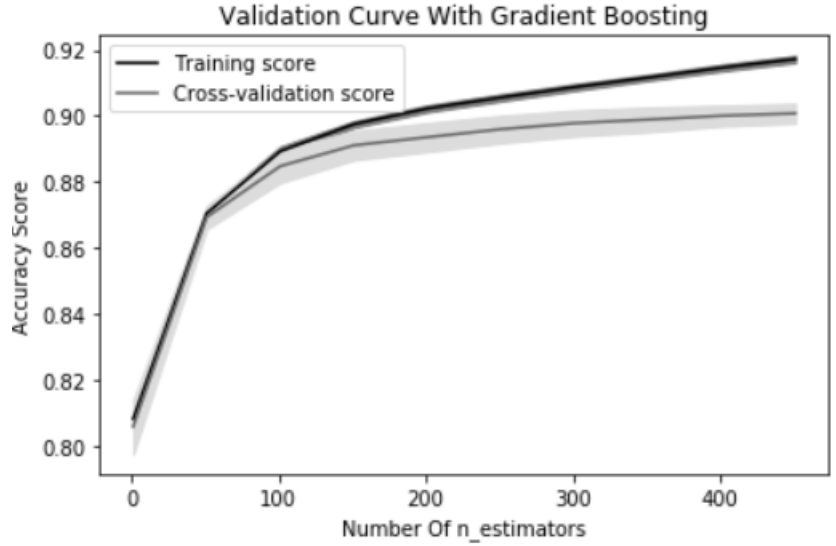
Using GridSeachCV (10 fold cross validation)

```
print (grid.best_score_)
print (grid.best_params_)
print (grid.best_estimator_)
```

```
0.8926379379937673
{'max_depth': 11}
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=11,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=None,
            splitter='best')
```








```
0.928214757894134
{'n_estimators': 27}
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=27, n_jobs=None,
            oob_score=False, random_state=None, verbose=0,
            warm_start=False)
```

```
0.8562625997601467
{'n_estimators': 210}
GradientBoostingClassifier(criterion='friedman_mse', init=None,
              learning_rate=1.0, loss='deviance', max_depth=1,
              max_features=None, max_leaf_nodes=None,
              min_impurity_decrease=0.0, min_impurity_split=None,
              min_samples_leaf=1, min_samples_split=2,
              min_weight_fraction_leaf=0.0, n_estimators=210,
              n_iter_no_change=None, presort='auto', random_state=0,
              subsample=1.0, tol=0.0001, validation_fraction=0.1,
              verbose=0, warm_start=False)
```

Final Model evaluation on validation set

```
#Model Gradient Tree Boosting
from sklearn.ensemble import GradientBoostingClassifier
model3 = GradientBoostingClassifier(n_estimators=210, learning_rate=1.0, max_depth=1, random_state=0)
```

```
model3.fit(X_train, y_train)
```

```
GradientBoostingClassifier(criterion='friedman_mse', init=None,
              learning_rate=1.0, loss='deviance', max_depth=1,
              max_features=None, max_leaf_nodes=None,
              min_impurity_decrease=0.0, min_impurity_split=None,
              min_samples_leaf=1, min_samples_split=2,
              min_weight_fraction_leaf=0.0, n_estimators=210,
              n_iter_no_change=None, presort='auto', random_state=0,
              subsample=1.0, tol=0.0001, validation_fraction=0.1,
              verbose=0, warm_start=False)
```

```
y_pred1 = model1.predict(X_vald)
# Model Accuracy
print(accuracy_score(y_vald, y_pred1))
```

```
0.931625
```

```
#model2 Decision Tree
from sklearn.tree import DecisionTreeClassifier
model2 = DecisionTreeClassifier(max_depth = 11)
```

```
model2.fit(X_train, y_train)
```

```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=11,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=None,
            splitter='best')
```

```
y_pred2 = model2.predict(X_vald)
# Model Accuracy
print( accuracy_score(y_vald, y_pred2))
```

```
0.89925
```

```
#Model Gradient Tree Boosting
from sklearn.ensemble import GradientBoostingClassifier
model3 = GradientBoostingClassifier(n_estimators=96, learning_rate=1.0, max_depth=1, random_state=0)
```

```
model3.fit(X_train, y_train)
```

```
GradientBoostingClassifier(criterion='friedman_mse', init=None,
              learning_rate=1.0, loss='deviance', max_depth=1,
              max_features=None, max_leaf_nodes=None,
              min_impurity_decrease=0.0, min_impurity_split=None,
              min_samples_leaf=1, min_samples_split=2,
              min_weight_fraction_leaf=0.0, n_estimators=96,
              n_iter_no_change=None, presort='auto', random_state=0,
              subsample=1.0, tol=0.0001, validation_fraction=0.1,
              verbose=0, warm_start=False)
```

```
y_pred3 = model3.predict(X_vald)
# Model Accuracy
print( accuracy_score(y_vald, y_pred3))
```

```
0.857625
```

Final Models Evaluation on Test set

---------- RandomForest ----------

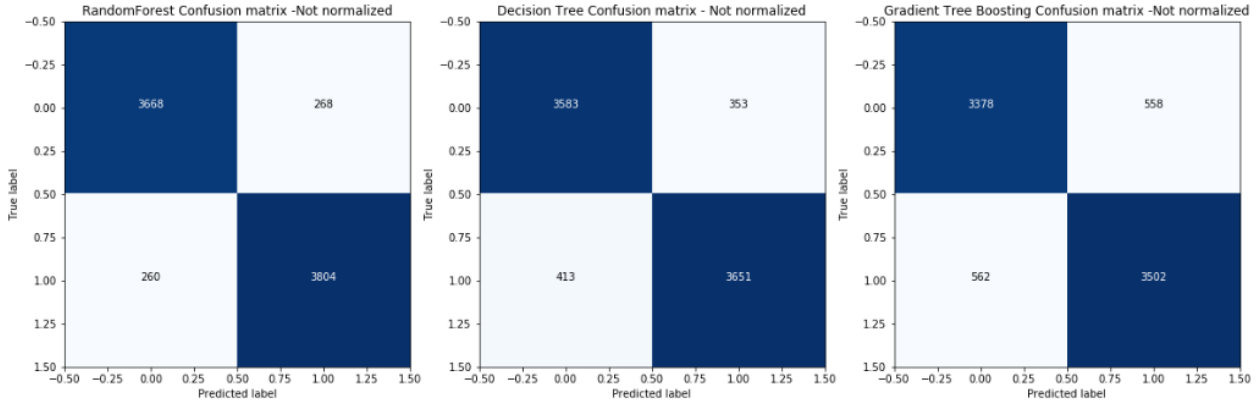
	precision	recall	f1-score	support
0	0.93	0.93	0.93	3936
1	0.93	0.94	0.94	4064
micro avg	0.93	0.93	0.93	8000
macro avg	0.93	0.93	0.93	8000
weighted avg	0.93	0.93	0.93	8000

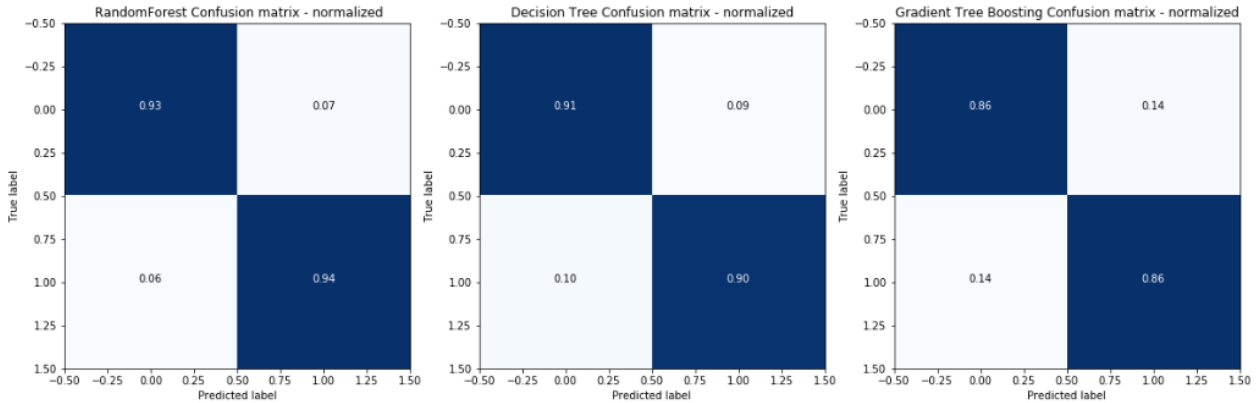
---------- Decision Tree ----------

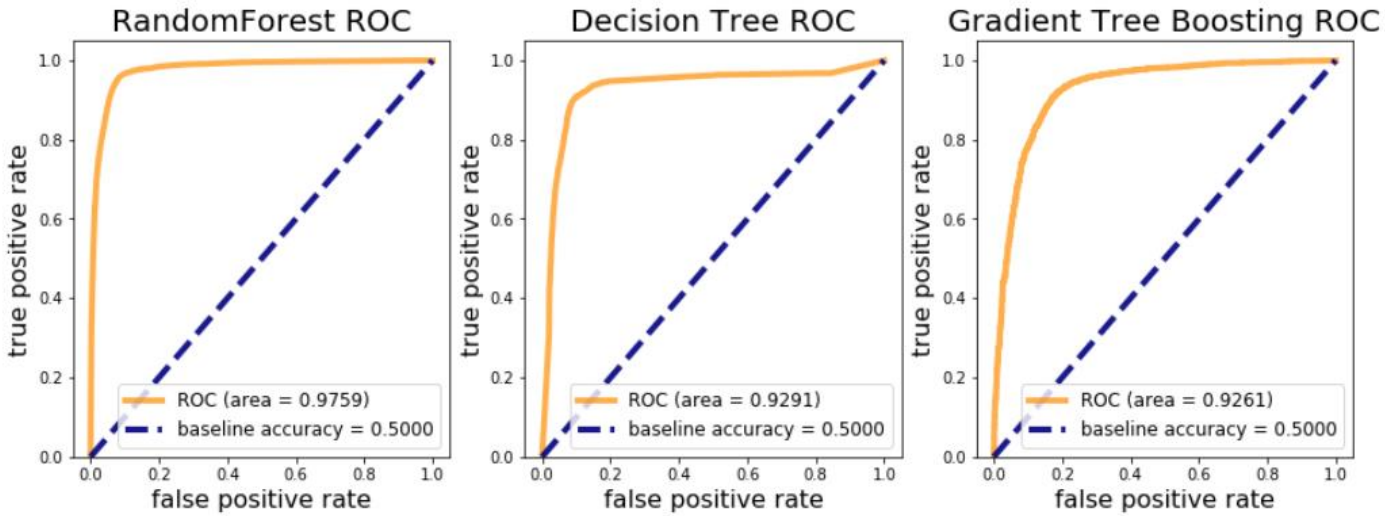
	precision	recall	f1-score	support
0	0.90	0.91	0.90	3936
1	0.91	0.90	0.91	4064
micro avg	0.90	0.90	0.90	8000
macro avg	0.90	0.90	0.90	8000
weighted avg	0.90	0.90	0.90	8000

---------- Gradient Tree Boosting ----------

	precision	recall	f1-score	support
0	0.86	0.86	0.86	3936
1	0.86	0.86	0.86	4064
micro avg	0.86	0.86	0.86	8000
macro avg	0.86	0.86	0.86	8000
weighted avg	0.86	0.86	0.86	8000







Approach:

Conclusion

initial Analysis

- **Univariate Analysis** (detailed data dictionary / list all variables and understand what each variable mean and represent)(what is the source of each variable directly measured or calculated based on other variables or does the variable has a time domain associated) - (Decide on the dependent (target) variable) (Assigning the correct data types and appropriate column names)(for numeric attributes check the 5 number summary min 1sQ mean 3rdQ max)(for categorical attributes check and decide on the levels , frequency) (check and deal with data inconsistencies , missing values , errors , duplicates , outliers (boxplots) , numeric signs , upper and lower cases , spaces or special characters in striges) (check distributions of the variables : Normal distribution?)(Low variance filter)(check the imbalance in the dependent variable) (check time variable) (Univariate visualizations)
- **BiVariate Analysis** (pairwise relations) (pairwise visuliazations like scatter plots) (correlation analysis spearman or pearson).
- **Multivariate Analysis** (relations between more than 2 variables)(statistical tools such as one way ANOVA analysis or rank or to compare the means.

Exploratory Analysis

- Normalizing / scaling
- Sub-setting the data
- Decision rules , association rules , n grams
- Clustering such as K-mean
- Hypothesise analysis
- Correlation analysis
- NLP analysis for some text attributes

Dimensionality Reduction

- remove attributes with too many missing values
- remove attributes with zero or very low variance
- remove one of the attributes with high correlations with other - prefer the one with more missing values or lower variance
- Feature selection(decide on the importance of the attribute using statistical measures like information gain or Gini index) (Forward selection and backward elimination)

Experimental Desgin

- Randomizing, Splitting the data into training and test sets to training, validation and testing
- Treatment for imbalance (under-sampling the majority class and over-sampling the minority class)
- Cross validation such as 10 folds

Modeling

- building the Classification Models
- Train the model
- Validate the model

Evaluation

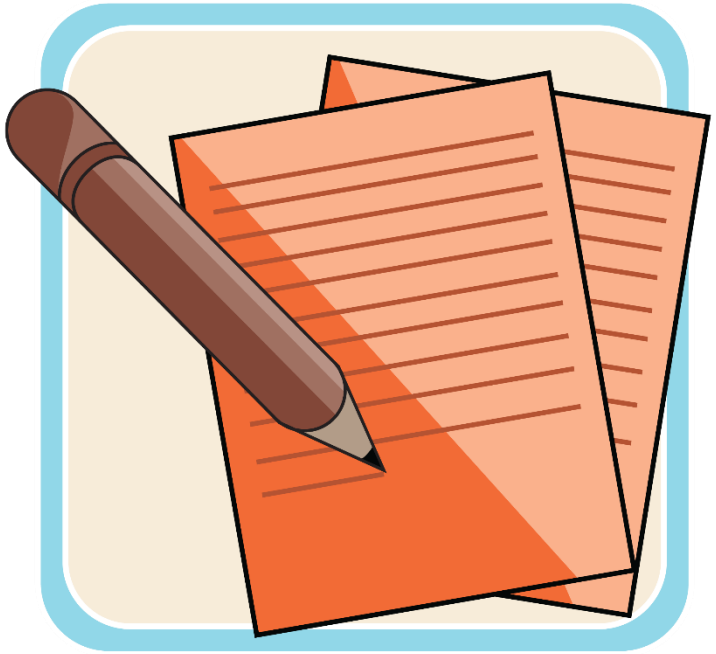
- Classification (Confusion matric , ROC (receiver operating characteristics) , Accuracy , Recall , Precision)
- compare performance between different models (contingency tables , multivariate analysis of variance)

Improving the Model

- deal with very high accuracy(Fixation variables , overfitting)
- deal with very low performance (many need new variables or more observations)
- iterate to improve the model performance

Conclusions

- Inferences , dissuasions
- Threats to validity (Internal , External , construct) and propose the solution that may mitigate these threats
- Next/ future steps



Conclusions

Inferences , discussion:-

- **Post 2005 there are significant increase in the number of the terrorism attacks and majority of them are happening in Middle East and South Asia.**
- **The best classification model was random forest and the model was able to predict the success of an attack with high accuracy > 90% by knowing the attack features. This prediction can help in future to prevent/apprehended or mitigate terrorism attacks.**
- **The study also explored the terrorism groups active in each country and their signature (attack type , weapon type , target type) which can help to mitigate and control future attacks.**

Threats to validity and Solutions to mitigate these threats:

- **The analysis based on long period of time 1970 – 2017 about 47 years and the pattern, countries and terrorism group signatures keep changing, future deeper analysis will have to analysis based on each decade or based on a certain country or city. Also 1993 year data is missing.**
- **Source Data Legacy issues: The GTD now includes incidents of terrorism from 1970 to 2017, however a number of new variables were added to the database beginning with the post-1997 data collection effort. Wherever possible, values for these new variables were retroactively coded for the original incidents .This mean some fields are presently only systematically available with incidents occurring after 1997.**
- **Some additional important attributes were not considered in the original data like political or economical status for the country. For accurate future analysis may need to add more external datasets.**
- **Only one hyper-parameter per model was tuned. Future tuning for more hyper-parameters may improve the model performance.**

Next / Future steps:

- **Deeper analysis on one county or a city over decade periods.**
- **Deeper analysis on one terrorism group to understand their signature in weapon types, attack types and motives.**
- **More statistical tests and analysis using external dataset (mental illness, education, GDP/economic, war status, unemployment ,internet growth / technology , healthcare, weather, virtual currencies such as crypto coin).**
- **Time series analysis to understand the future trend for the terrorism attacks.**

Thank You

Questions?

Project repository:
https://github.com/emilkaram/CKM136XJ0-Global-Terrorism-Data-Analytics-Capstone