Data Visualization: Final Project Review

GLOBAL TERRORISM ANALYSIS

Team-8 Members:

Shashwat Kumar-20BCE2818
Dhruv Sethi - 20BCE2240
Ikshit Samanta - 20BCE2821
Neha Baggan - 20BCE2030

Slot-D1
Faculty Name: Lydia Jane
CSE 3020

[Link to the Project Presentation]

OBJECTIVE AND MOTIVATION

Use global terrorism database and perform a comprehensive analysis in order to gain insights into the terrorism.
Present data and insights in a user friendly and **interactive manner**.
Used graphical plots such as scatter plots and maps for an intuitive and better understanding.
Used libraries like - matplotlib, pandas, numpy, seaborn, plotly to present data.
Proposed Graphs - Scatter-plot, Histograms, Bar Graphs, Maps, Pie chart,

We are going to be using the data set from the following link :
https://www.start.umd.edu/gtd/access/

ABOUT DATASET

DB - Maintained by researchers at the National Consortium for the Study of Terrorism and Responses to Terrorism **(START)**
Released - July, 2018
Information on 180,000+ attacks.
Geography - Worldwide
Time period - 1970 - 2019, except 1993
Unit of Analysis - Attack
100+ variables on location, tactics, perpetrators, targets, and outcomes
Sources - Unclassified media articles

EXPLORING DATASET & PRE-PROCESSING DATA

Importing necessary libraries.
Loading and exploring the dataset.
Started decompressing, cleaning and formatting data in accordance with our further analysis.
Visualisation using Histogram.
The dataset is semi-structured so it requires a lot of cleaning and preprocessing tasks in order to make it accessible for the entire project.

IMPORTING ALL REQUIRED MODULES

jupyter Data Viz (GTA) Last Checkpoint: an hour ago (autosaved)

GLOBAL TERRORISM ANALYSIS

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
import sklearn
warnings.filterwarnings("ignore")
```

In [2]:
```
import plotly
import plotly.express as px
import plotly.io as pio
import chardet
from collections import Counter
from bokeh.plotting import figure
from bokeh.io import show, output_notebook
from bokeh.layouts import column
from bokeh.models import Select, Slider, CustomJS, HoverTool, ColumnDataSource
from bokeh.palettes import Turbo11 as palette
import itertools
```

jupyter Data Viz (GTA) Last Checkpoint: an hour ago (autosaved)

In [9]:
```
df.isnull().sum()
```

Out[9]:
```
Year                      0
Month                     0
Day                       0
Country_no                0
Country                   0
Region                    0
City                    434
success                   0
Attack_type               0
no_of_kills           10313
property_value       142702
Target_type               0
latitude               4556
group_name                0
longitude              4557
targsubtype1_txt      10373
Target                  636
weaptype1_txt             0
Weapon_detail         67670
dtype: int64
```

In [10]:
```
df1 = pd.read_csv('globalterrorismdb_0718dist (1).csv', encoding = 'latin1', low_memory=False)
```

jupyter Data Viz (GTA) Last Checkpoint: an hour ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Graphs

In [16]:
```
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
import seaborn as sns
import numpy as np
plt.style.use('fivethirtyeight')
import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
import plotly.tools as tls
from matplotlib import animation,rc
import io
import base64
from IPython.display import HTML, display
import warnings
warnings.filterwarnings('ignore')
import codecs
from subprocess import check_output
```

jupyter Data Viz (GTA) Last Checkpoint: an hour ago (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

```
import codecs
from subprocess import check_output
```

In [17]:
```
terror=pd.read_csv('globalterrorismdb_0718dist (1).csv',encoding='ISO-8859-1')
terror.rename(columns={'iyear':'Year','imonth':'Month','iday':'Day','country_txt':'Country','region_txt
terror=terror[['Year','Month','Day','Country','Region','city','latitude','longitude','AttackType','Kill
terror['casualities']=terror['Killed']+terror['Wounded']
terror.head(3)
```

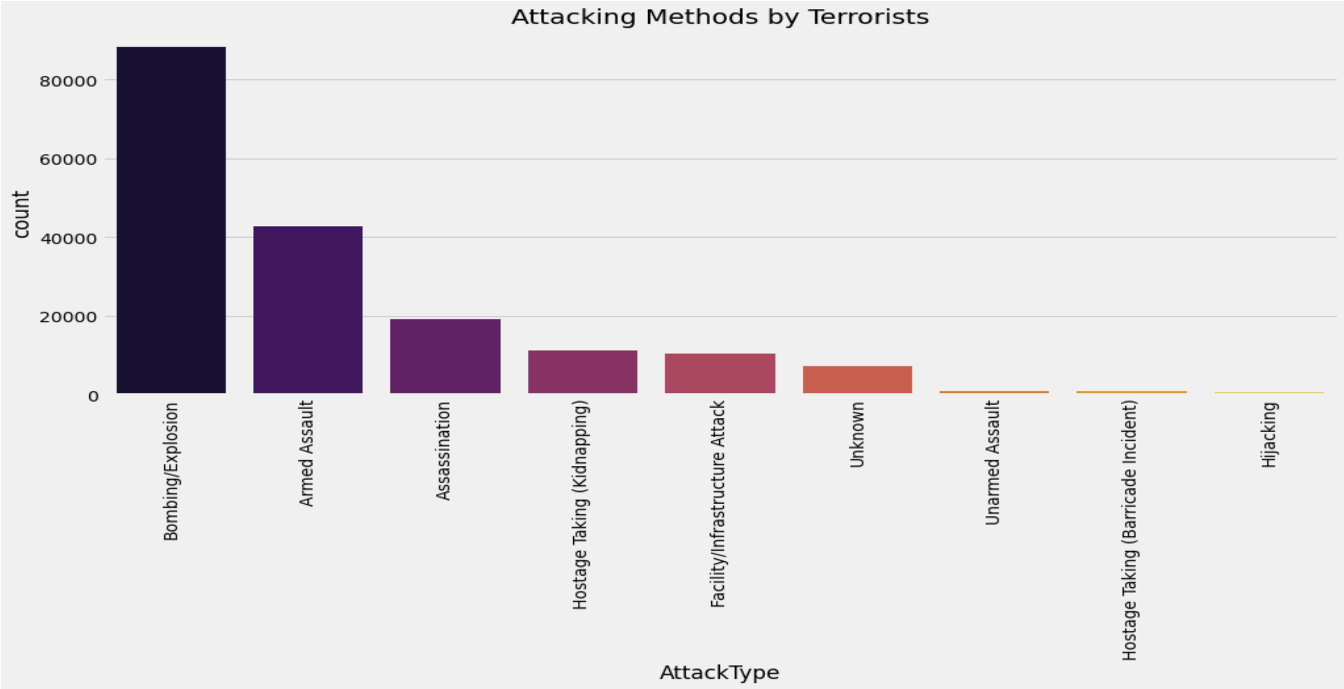
Out[17]:

	Year	Month	Day	Country	Region	city	latitude	longitude	AttackType	Killed	Wounded	Target	Summ
0	1970	7	2	Dominican Republic	Central America & Caribbean	Santo Domingo	18.456792	-69.951164	Assassination	1.0	0.0	Julio Guzman	N
1	1970	0	0	Mexico	North America	Mexico city	19.371887	-99.086624	Hostage Taking (Kidnapping)	0.0	0.0	Nadine Chaval, daughter	N
2	1970	1	0	Philippines	Southeast Asia	Unknown	15.478598	120.599741	Assassination	1.0	0.0	Employee	N

Non-Interactive Plots:

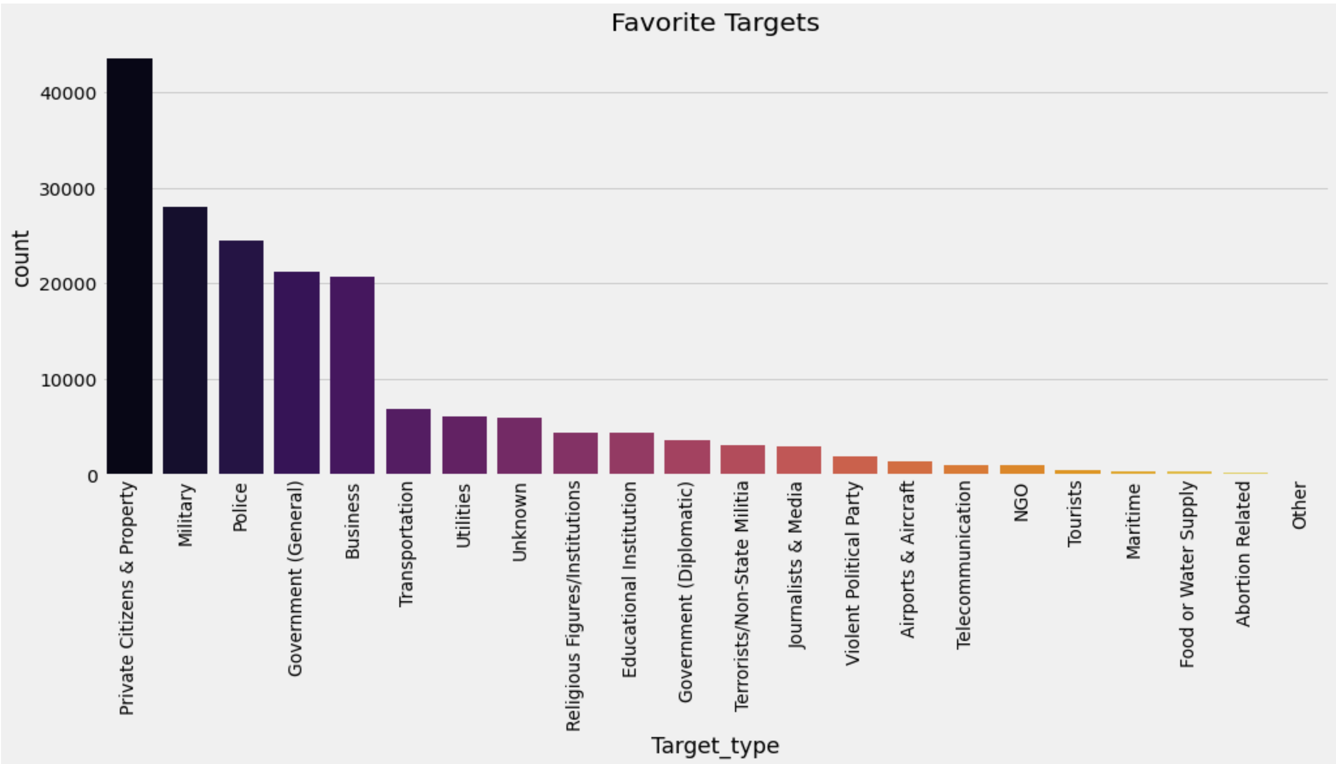
1. Attacking Methods by Terrorists

plt.subplots(figsize=(15,6))
sns.countplot('AttackType',data=terror,palette='inferno',order=terror['AttackType'].value_counts().index)
plt.xticks(rotation=90)
plt.title('Attacking Methods by Terrorists')
plt.show()



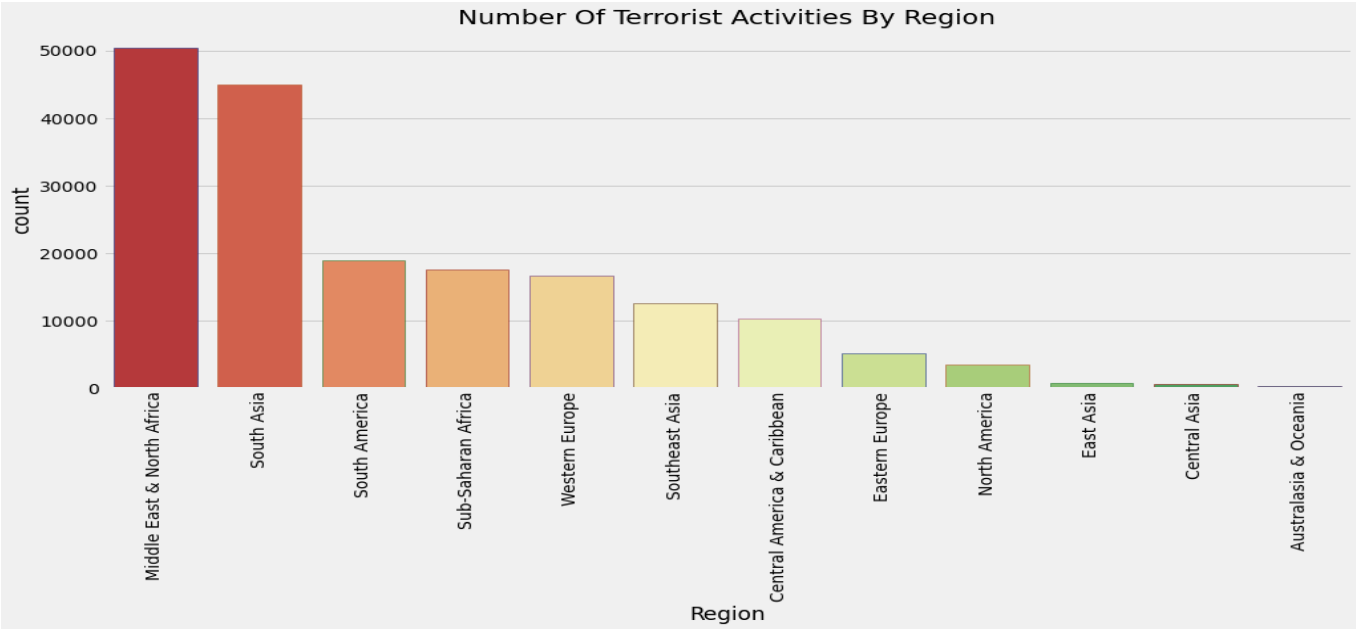
2. Favourite Targets

```
plt.subplots(figsize=(15,6))
sns.countplot(terror['Target_type'],
palette='inferno',order=terror['Target_type'].value_counts().index)
plt.xticks(rotation=90)
plt.title('Favorite Targets')
plt.show()
```



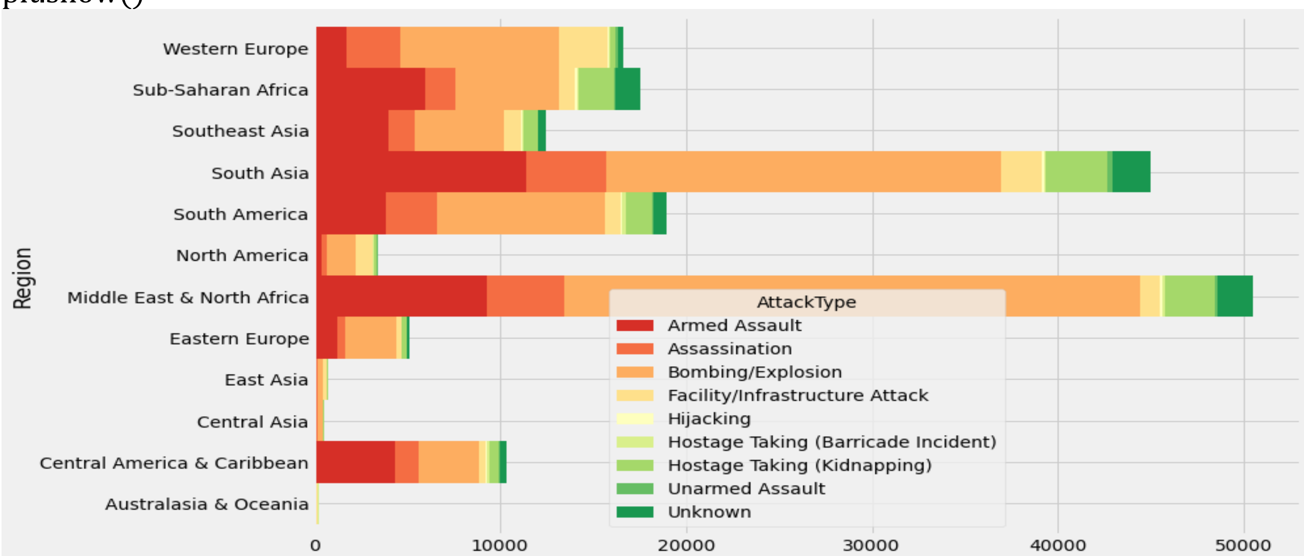
3. Number of Terrorist Activities by Region

```
plt.subplots(figsize=(15,6))
sns.countplot('Region',data=terror,
palette='RdYlGn',edgecolor=sns.color_palette('dark',7),order=terror['Region'].value_c
ounts().index)
plt.xticks(rotation=90)
plt.title('Number Of Terrorist Activities By Region')
plt.show()
```



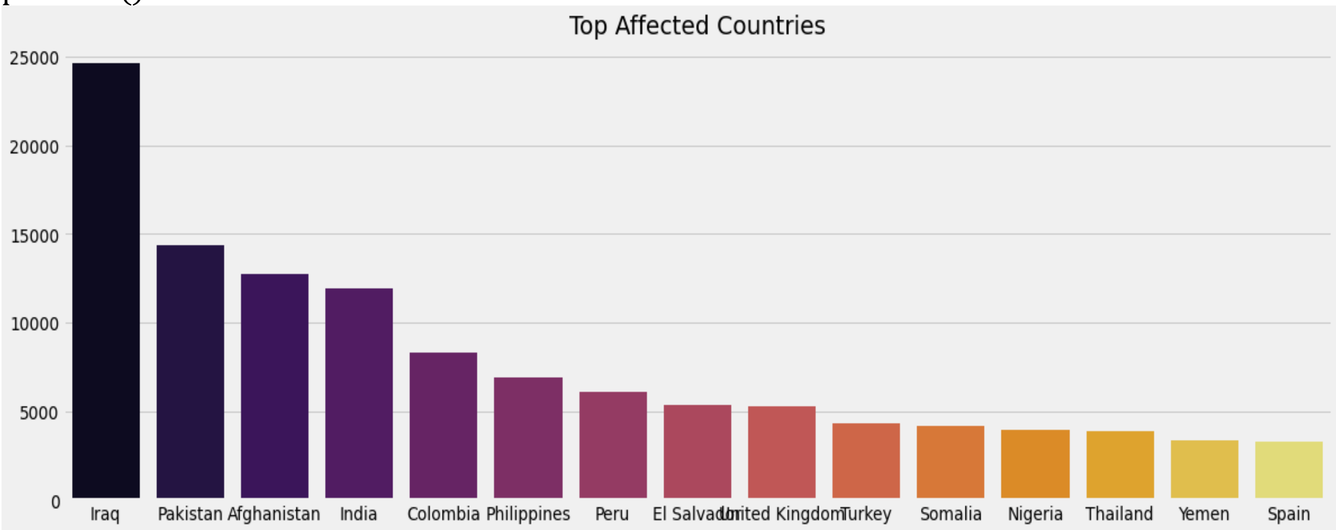
4. Attack Type By Region

```
pd.crosstab(terror.Region,terror.AttackType).plot.barh(stacked=True,width=1,color=s
ns.color_palette('RdYlGn',9))
fig=plt.gcf()
fig.set_size_inches(12,8)
plt.show()
```



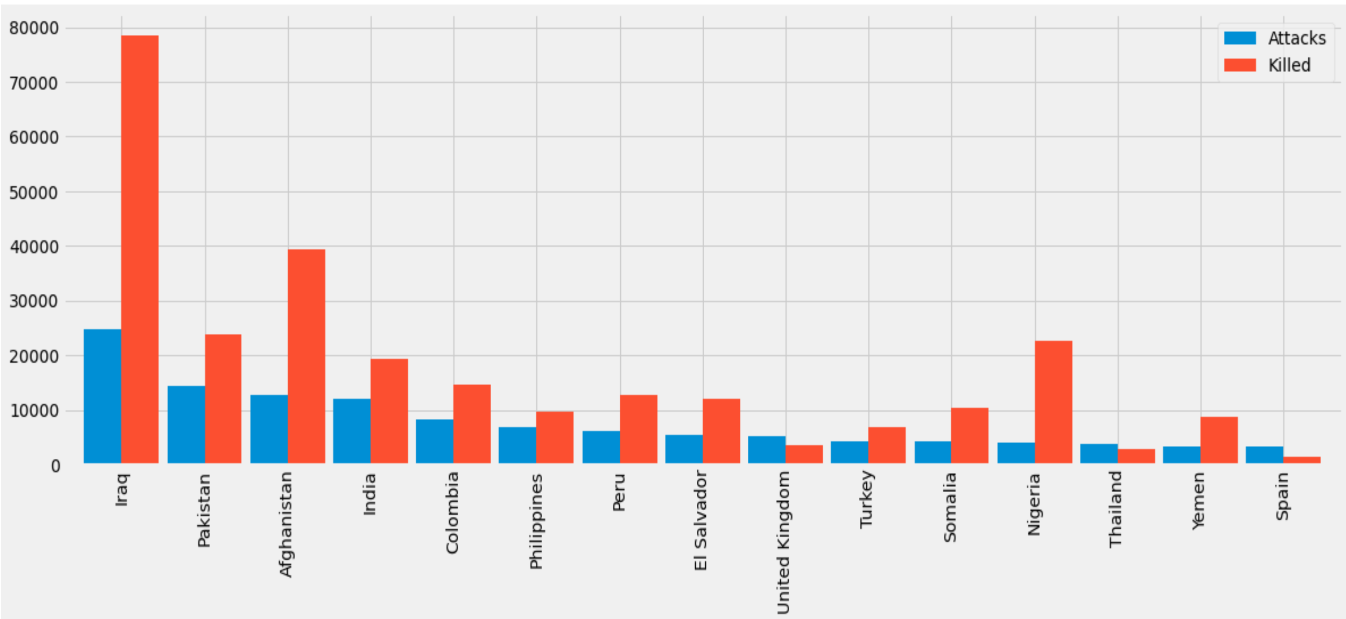
5. Top Affected Countries

```
plt.subplots(figsize=(18,6))
sns.barplot(terror['Country'].value_counts()[:15].index,terror['Country'].value_counts()
[:15].values,palette='inferno')
plt.title('Top Affected Countries')
plt.show()
```



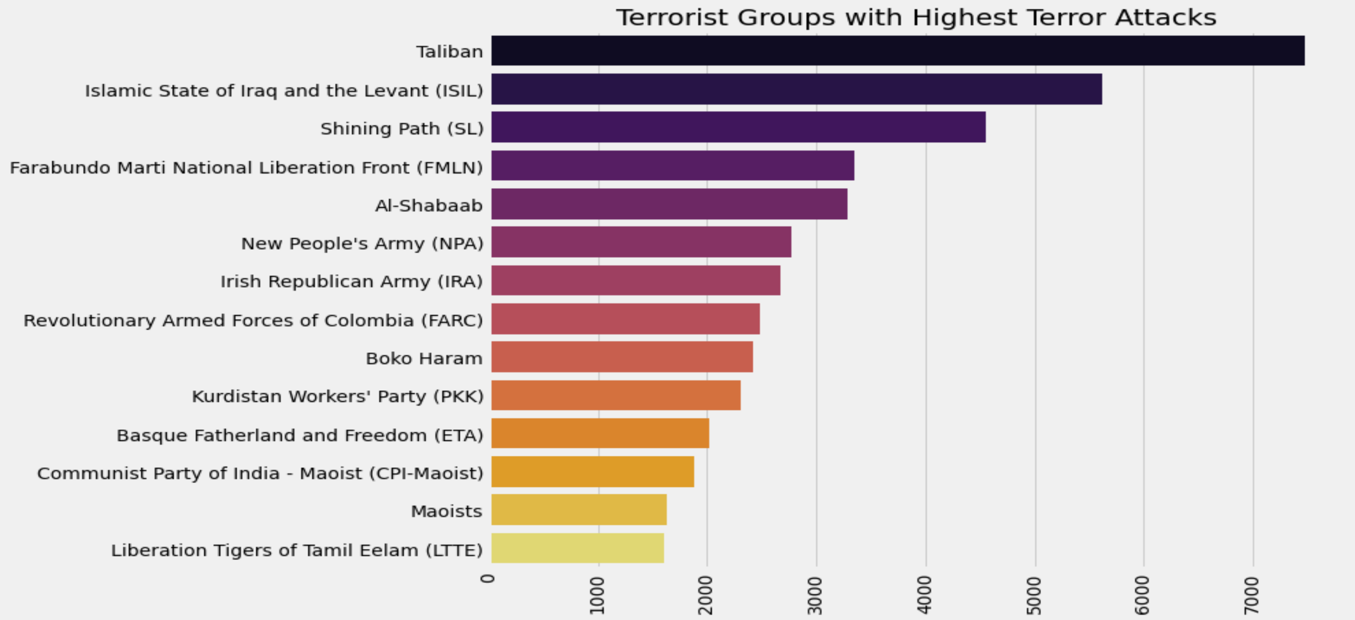
6. Attacks vs Killed by Country

```
coun_terror=terror['Country'].value_counts()[:15].to_frame()
coun_terror.columns=['Attacks']
coun_kill=terror.groupby('Country')['Killed'].sum().to_frame()
coun_terror.merge(coun_kill,left_index=True,right_index=True,how='left').plot.bar(wi
dth=0.9)
fig=plt.gcf()
fig.set_size_inches(18,6)
plt.show()
```



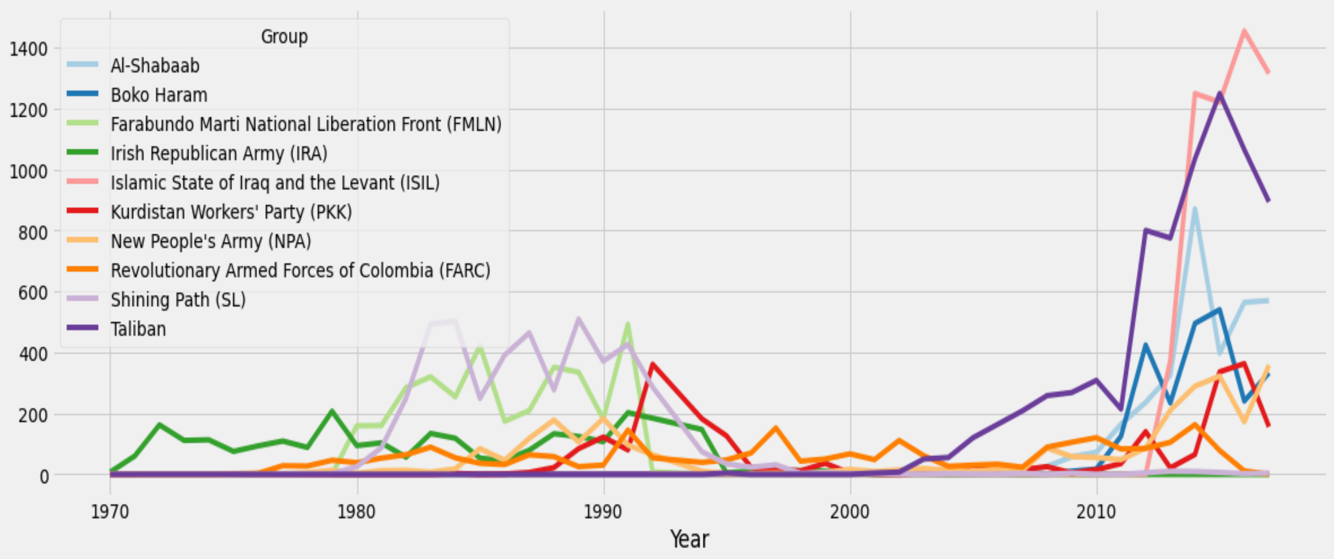
7. **Terrorist Groups with Highest Terror Attacks**

```
sns.barplot(terror['Group'].value_counts()[1:15].values,terror['Group'].value_counts()[
1:15].index,palette=('inferno'))
plt.xticks(rotation=90)
fig=plt.gcf()
fig.set_size_inches(10,8)
plt.title('Terrorist Groups with Highest Terror Attacks')
plt.show()
```



8. Groupwise Attacks

```
top_groups10=terror[terror['Group'].isin(terror['Group'].value_counts()[1:11].index)]
pd.crosstab(top_groups10.Year,top_groups10.Group).plot(color=sns.color_palette('Pair
ed',10))
fig=plt.gcf()
fig.set_size_inches(18,6)
plt.show()
```



In [28]:

```
df["affected"]=df.no_of_kills

df.head()
```

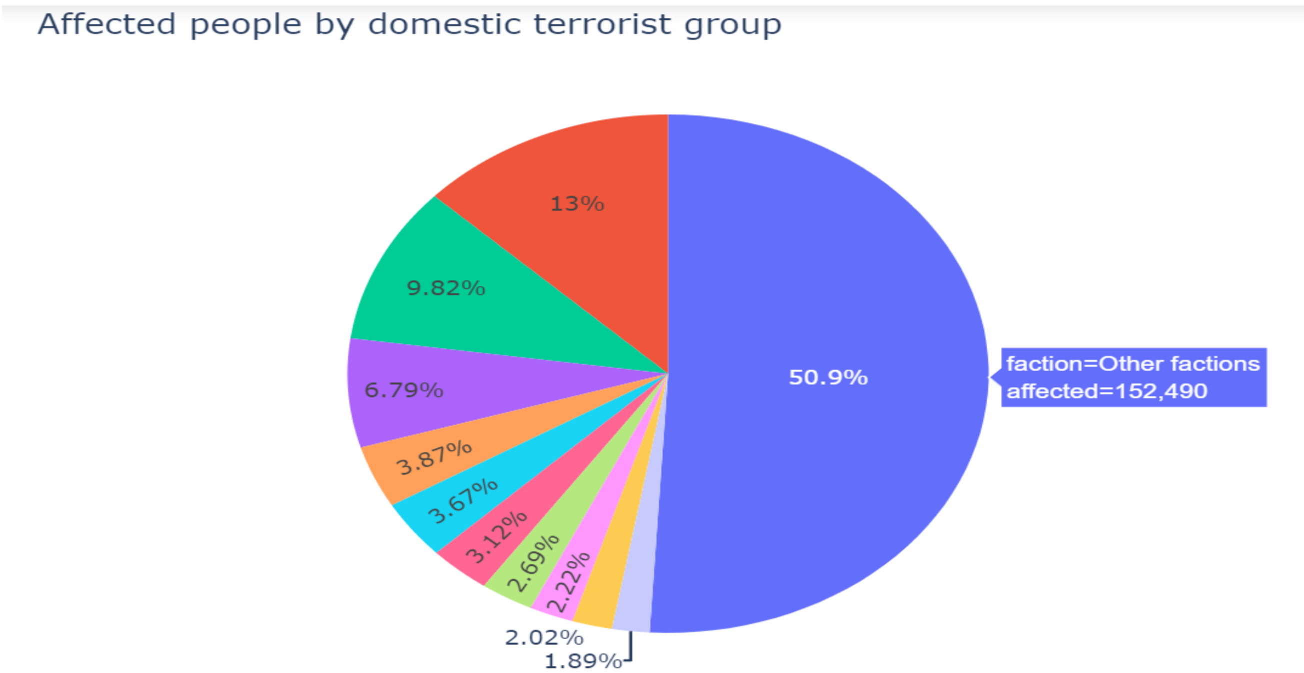
Out[28]:

	Year	Month	Day	Country_no	Country	Region	City	success	Attack_type	no_of_kills	property_value	Target_type	latitude	group_nam
0	1970	7	2	58	Dominican Republic	Central America & Caribbean	Santo Domingo	1	Assassination	1.0	NaN	Private Citizens & Property	18.456792	MANO-
1	1970	0	0	130	Mexico	North America	Mexico city	1	Hostage Taking (Kidnapping)	0.0	NaN	Government (Diplomatic)	19.371887	23rd Septemb Communi Leagu
2	1970	1	0	160	Philippines	Southeast Asia	Unknown	1	Assassination	1.0	NaN	Journalists & Media	15.478598	Unknow
3	1970	1	0	78	Greece	Western Europe	Athens	1	Bombing/Explosion	NaN	NaN	Government (Diplomatic)	37.997490	Unknow
4	1970	1	0	101	Japan	East Asia	Fukouka	1	Facility/Infrastructure Attack	NaN	NaN	Government (Diplomatic)	33.580412	Unknow

Interactive Plots

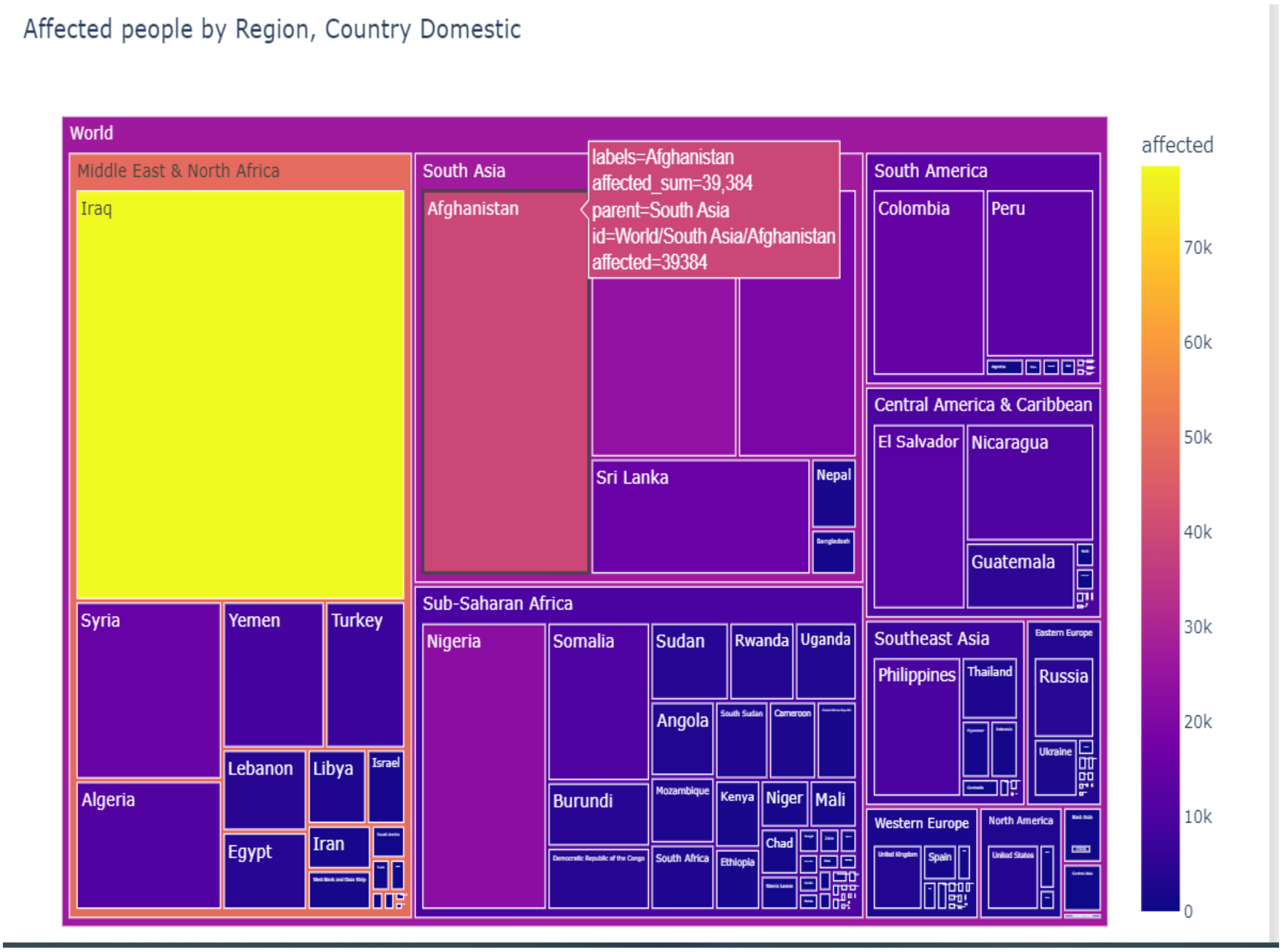
1. Affected people by domestic terrorist group

```
aff_gn = []
for name in df.group_name.unique():
aff_gn1 = df.affected[df.group_name.isin([name])].sum()
aff_gn.append(aff_gn1)
df_aff_gn=pd.DataFrame(list(zip(aff_gn,df.group_name.unique())),columns=['aff
ected','faction'])
df_aff_gn.loc[df_aff_gn[:]['affected']<5000,'faction']='Other factions'
df_aff_gn = df_aff_gn[df_aff_gn.faction != 'Unknown']
figx = px.pie(df_aff_gn, values='affected',names='faction', title='Affected people
by domestic terrorist group',width=700,height=500)
figx.update_layout(showlegend=False)
figx.show()
```



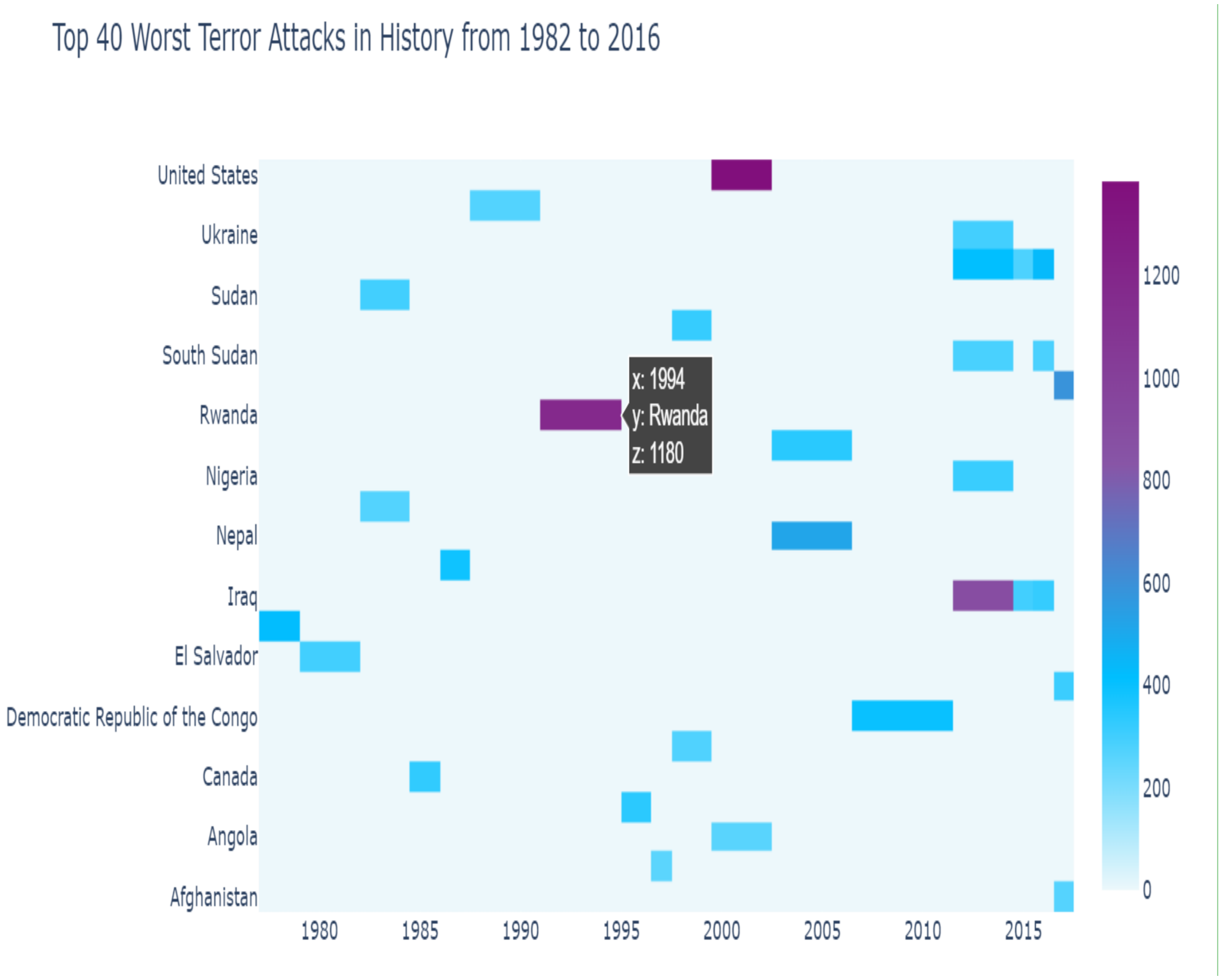
2. Affected people by Region, Country Domestic

```
aff_c = []
reg_c = []
for name in df.Country.unique():
   aff_c1 = df.affected[df.Country.isin([name])].sum()
   reg_c1 = df.Region[df.Country.isin([name])].values[0]
   aff_c.append(aff_c1)
   reg_c.append(reg_c1)
df_aff_rc=pd.DataFrame(list(zip(aff_c,df.Country.unique(),reg_c)),columns
=['affected','Country','Region'])
#df_aff_rc.loc[df_aff_rc[:]['affected']=0,'attacktype']='Other'
df_aff_rc = df_aff_rc[df_aff_rc.affected != 0]
figx = px.treemap(df_aff_rc,
path=[px.Constant('World'),'Region','Country'], values='affected',
              title='Affected people by Region, Country
Domestic',color='affected',
             width=1000,height=700)
figx.show()
```



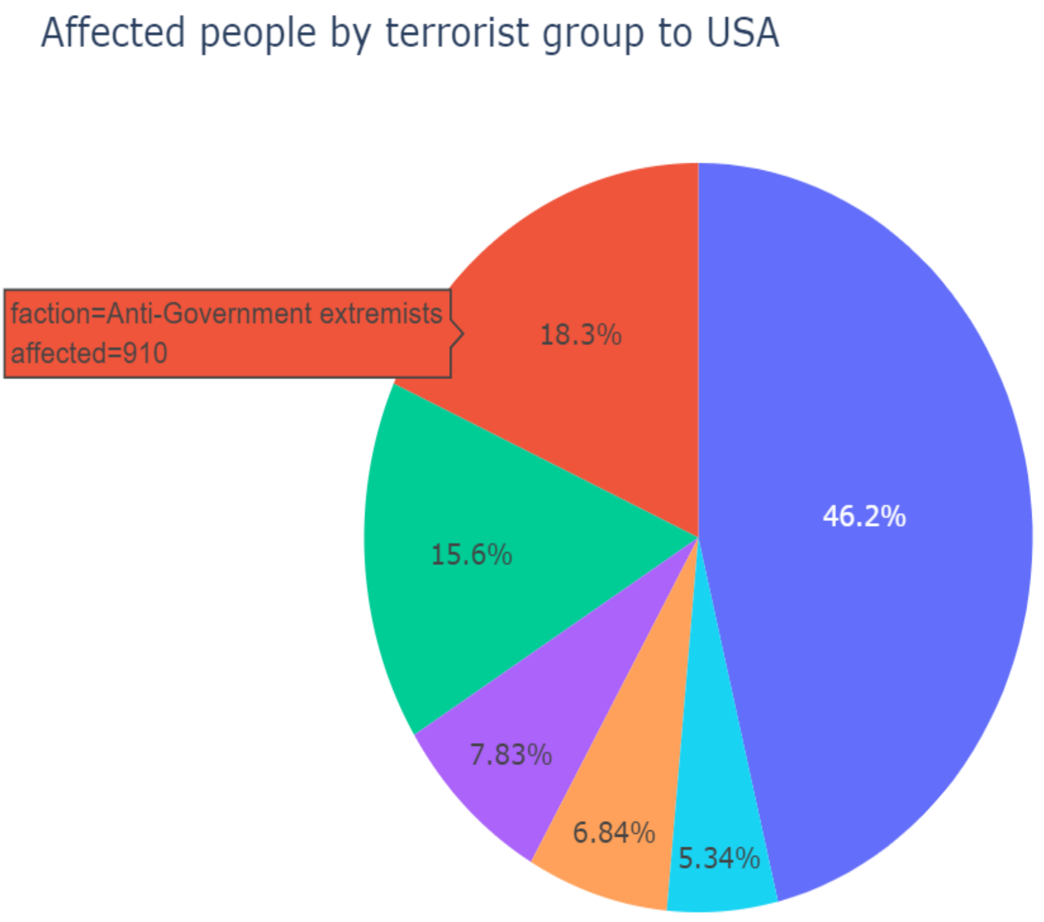
3. Top 40 Worst Terror Attacks in History from 1982 to 2016

```
import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
terror1 = df.sort_values(by='no_of_kills',ascending=False)[:40]
heat=terror1.pivot_table(index='Country',columns='Year',values='no_of_kills')
heat.fillna(0,inplace=True)
colorscale = [[0, '#edf8fb'], [.3, '#00BFFF'],  [.6, '#8856a7'],  [1, '#810f7c']]
heatmap = go.Heatmap(z=heat.values, x=heat.columns, y=heat.index,
colorscale=colorscale)
data = [heatmap]
layout = go.Layout(
   title='Top 40 Worst Terror Attacks in History from 1982 to 2016',
   xaxis = dict(ticks='', nticks=20),
   yaxis = dict(ticks='')
)
fig = go.Figure(data=data, layout=layout)
py.iplot(fig, filename='heatmap',show_link=False)
```



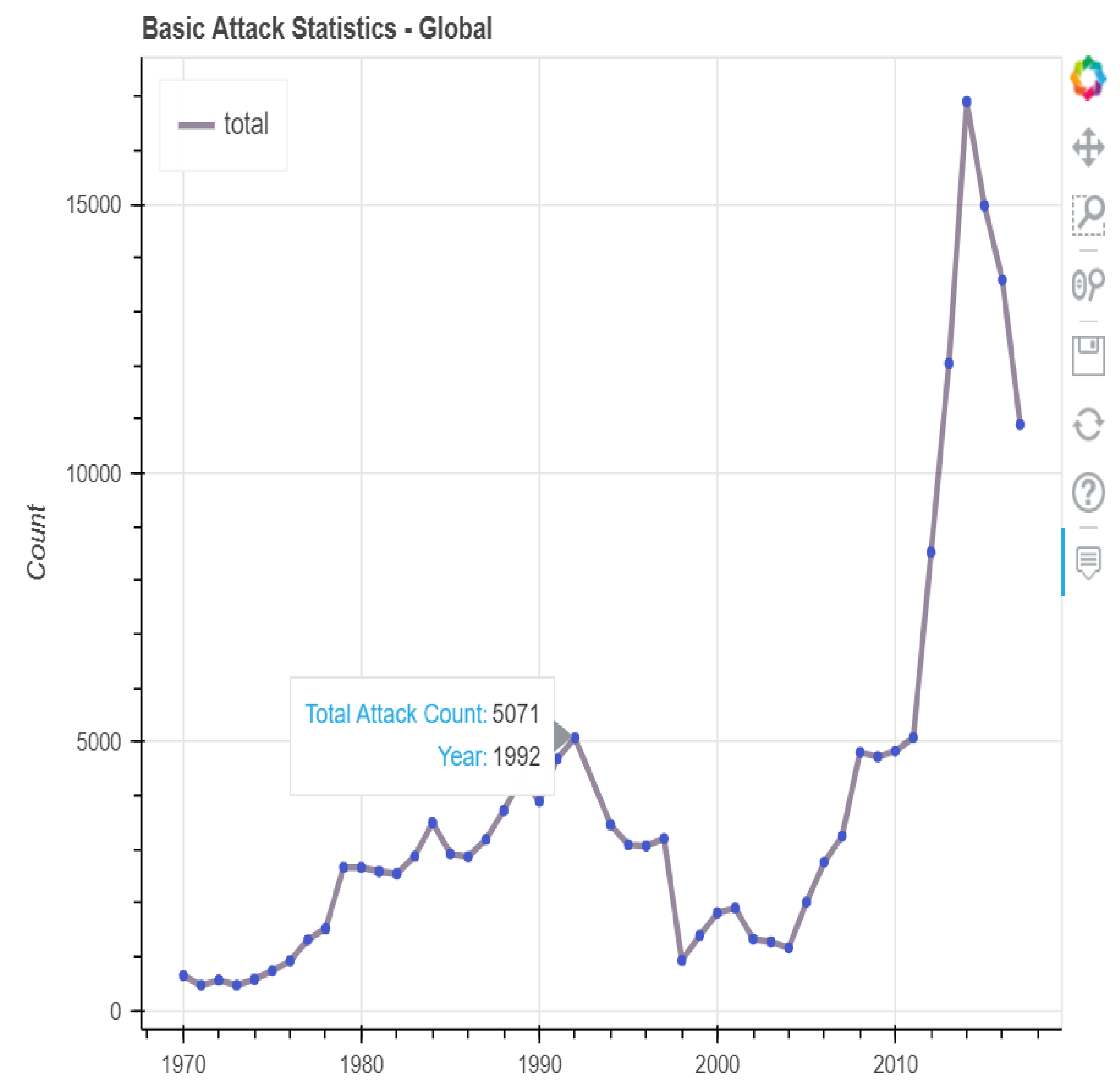
4. Affected people by terrorist group to USA

```
df1 = pd.read_csv('globalterrorismdb_0718dist (1).csv', encoding = 'latin1',
low_memory=False)
df1["affectedus"]=df1.nkillus+df1.nwoundus
df1["affectedter"]=df1.nkillter+df1.nwoundte
aff_gnus = []
for name in df1.gname.unique():
   aff_gnus1 = df1.affectedus[df1.gname.isin([name])].sum()
   aff_gnus.append(aff_gnus1)

df_aff_gn_us=pd.DataFrame(list(zip(aff_gnus,df1.gname.unique())),columns=['affected'
,'faction'])
df_aff_gn_us.loc[df_aff_gn_us[:]['affected']<200,'faction']='Other factions'
df_aff_gn_us = df_aff_gn_us[df_aff_gn_us.faction != 'Unknown']
figx = px.pie(df_aff_gn_us, values='affected',names='faction', title='Affected people by
terrorist group to USA',
          width=700,height=500)
figx.update_layout(showlegend=False)
figx.show()
```



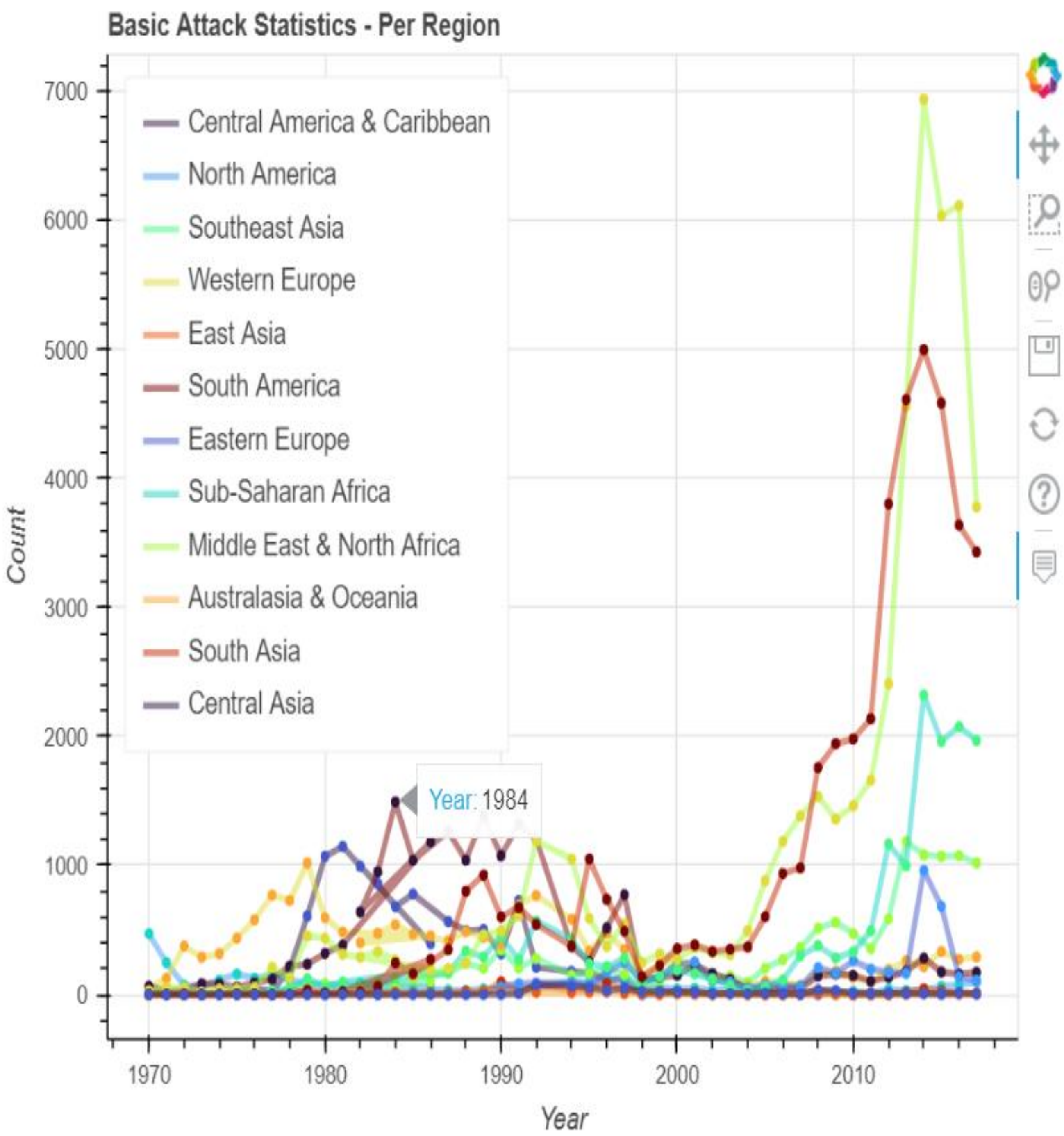
5. Basic Attack Statistics-Global

```
output_notebook()
cat = 'global'
f_list, f_init, f_title, f_title_plot= selectFilter(cat)
temp_df = createTempDf(cat, f_init)
cds_1 = ColumnDataSource(temp_df)

hover = createHover([('Total Attack Count', '@total'),('Year', '@x')])
createLinePlot(cds_1, f_title_plot[0], ['total'], hover, 'Year', 'Count')
```



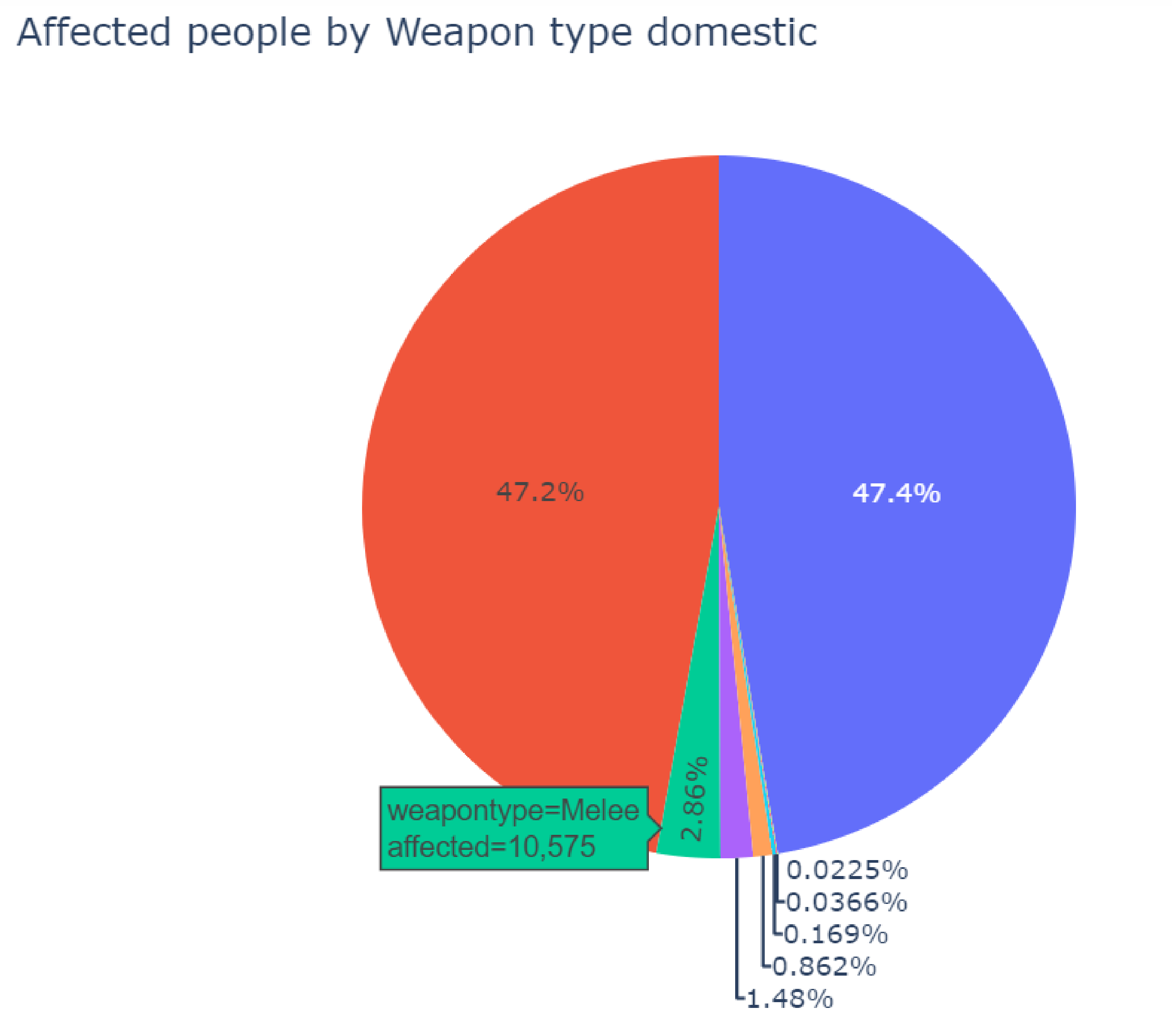
6. Basic Attack Statistics-Per Region

```
temp_df = createTempDf(cat, f_list[1])
cds_2 = ColumnDataSource(temp_df)

x=cds_2.data['index']; del cds_2.data['index']; cds_2.data['x'] = x
hover = createHover([('Year', '@x')])
createLinePlot(cds_2, f_title_plot[1], df1.region_txt.unique(), hover, 'Year',
'Count')
```



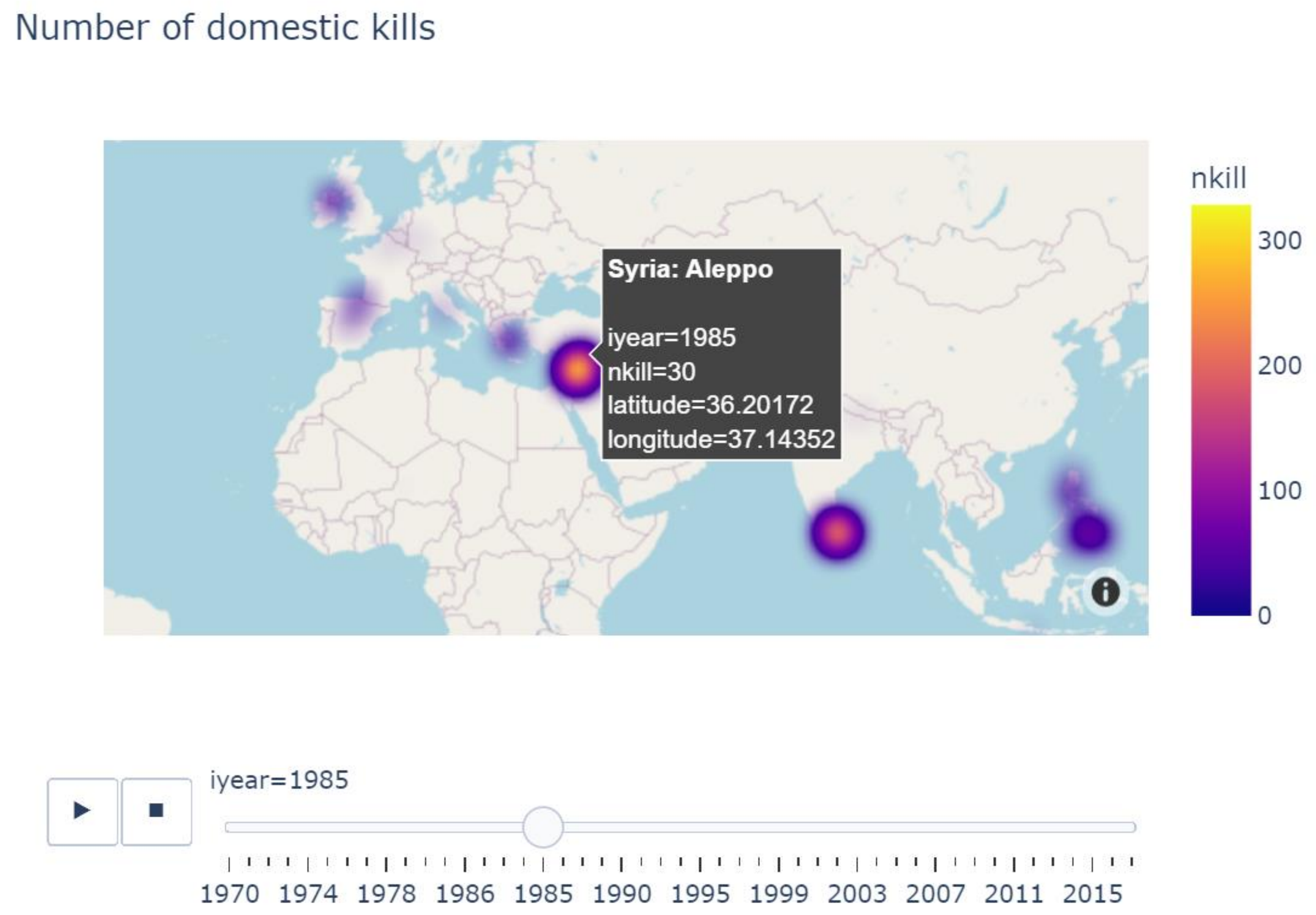
7. Affected people by Weapon type domestic

```
aff_weap = []
for name in df.weaptype1_txt.unique():
    aff_weap1 = df.affected[df.weaptype1_txt.isin([name])].sum()
    aff_weap.append(aff_weap1)

df_aff_weap=pd.DataFrame(list(zip(aff_weap,df.weaptype1_txt.unique())),columns=['a
ffected','weapontype'])
df_aff_weap.loc[df_aff_weap[:]['affected']<10,'weapontype']='Other'
df_aff_weap = df_aff_weap[df_aff_weap.weapontype != 'Unknown']
figx = px.pie(df_aff_weap, values='affected',names='weapontype', title='Affected
people by Weapon type domestic',
             width=700,height=500)
figx.update_layout(showlegend=False)
figx.show()
```



8. Number of Domestic kills

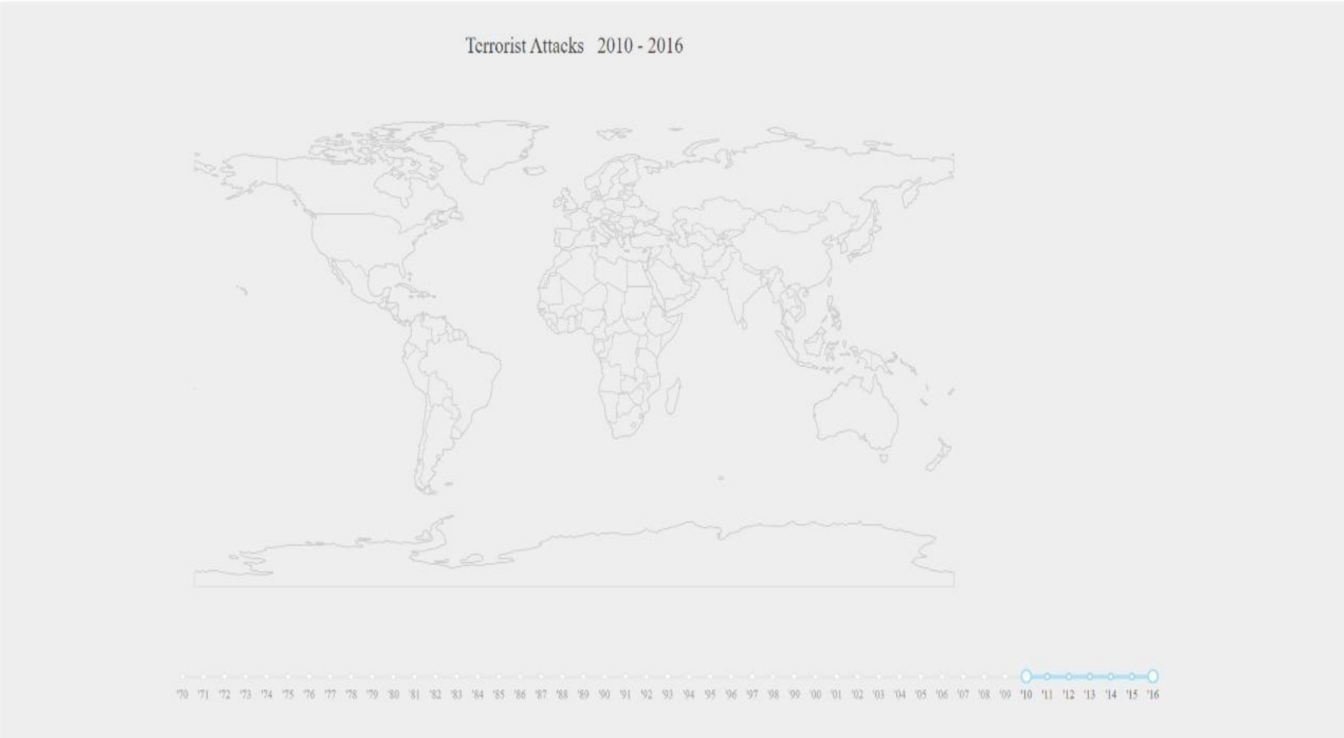
```
import plotly.express as px
center_point = dict(lon=0, lat=0)
figx = px.density_mapbox(df1, lat='latitude', lon='longitude', z="nkill",
center = center_point, hover_name='country_prov',zoom=0,
radius=20,mapbox_style= 'open-street-map', title='Number of domestic
kills',animation_frame='iyear',width=700, height=500)
figx.update(layout_coloraxis_showscale=True)
figx.show()
```



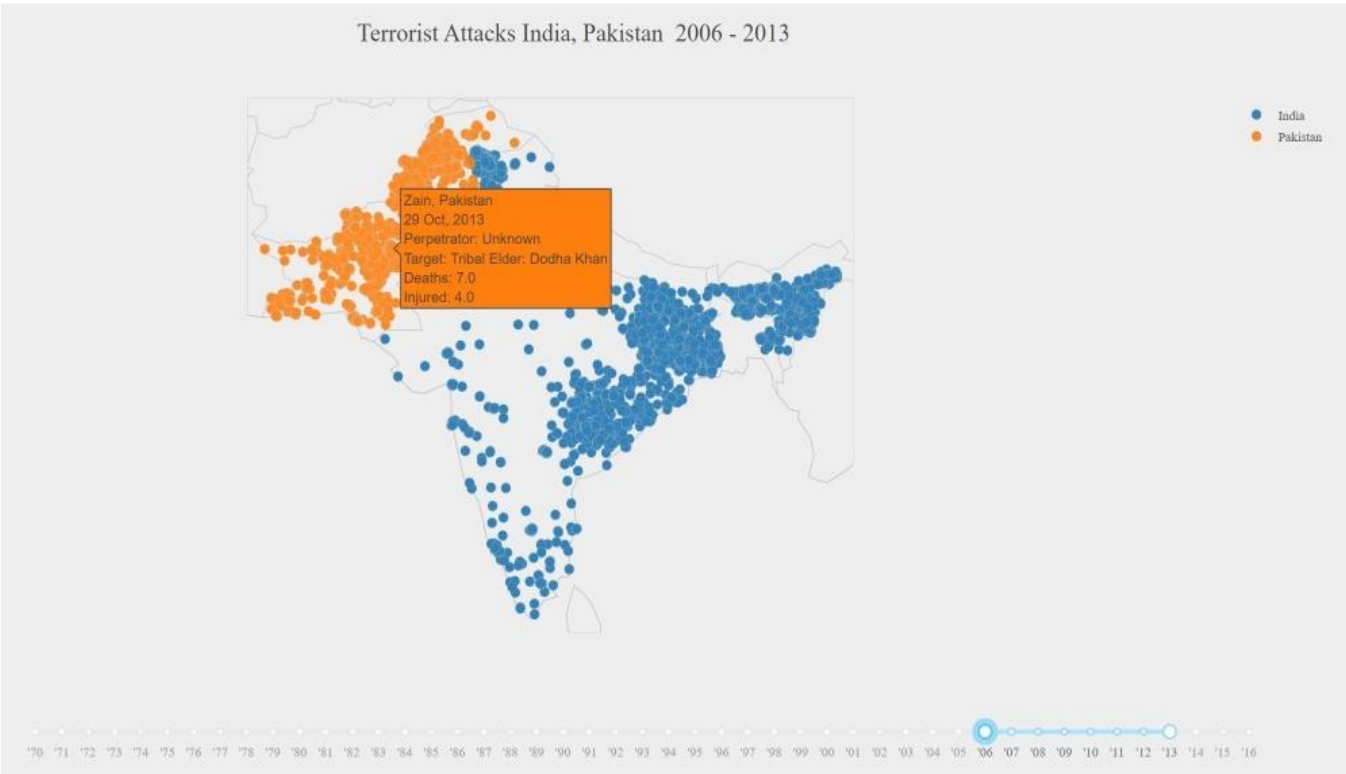
EXPLORING DATASET & PRE-PROCESSING DATA

There are no missing columns with entire data missing in the dataset. We intended to drop columns with missing values more than a given threshold (say 0.5, or about 5000 examples for now), because they won't assist in providing much sensitive data. Since we didn't build any pipelines for model selection and training, and only performed EDA, we didn't need to deal with missing values as for now. Hence for the time being, we have done our visualization on the already existing features.

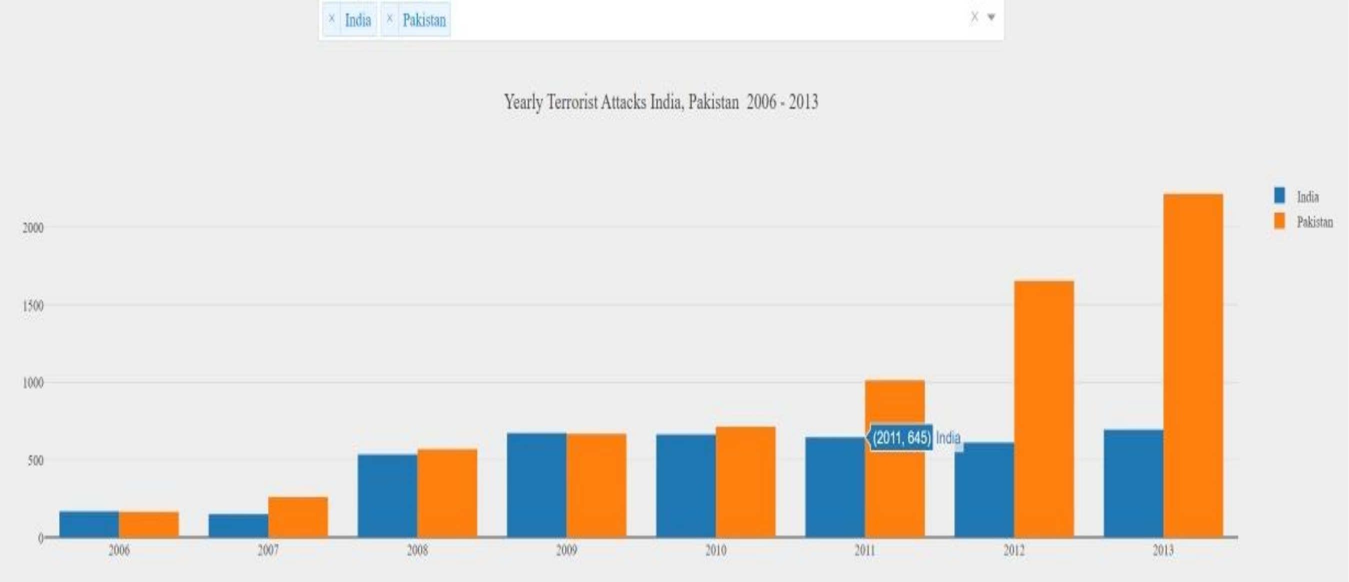
DASHBOARD



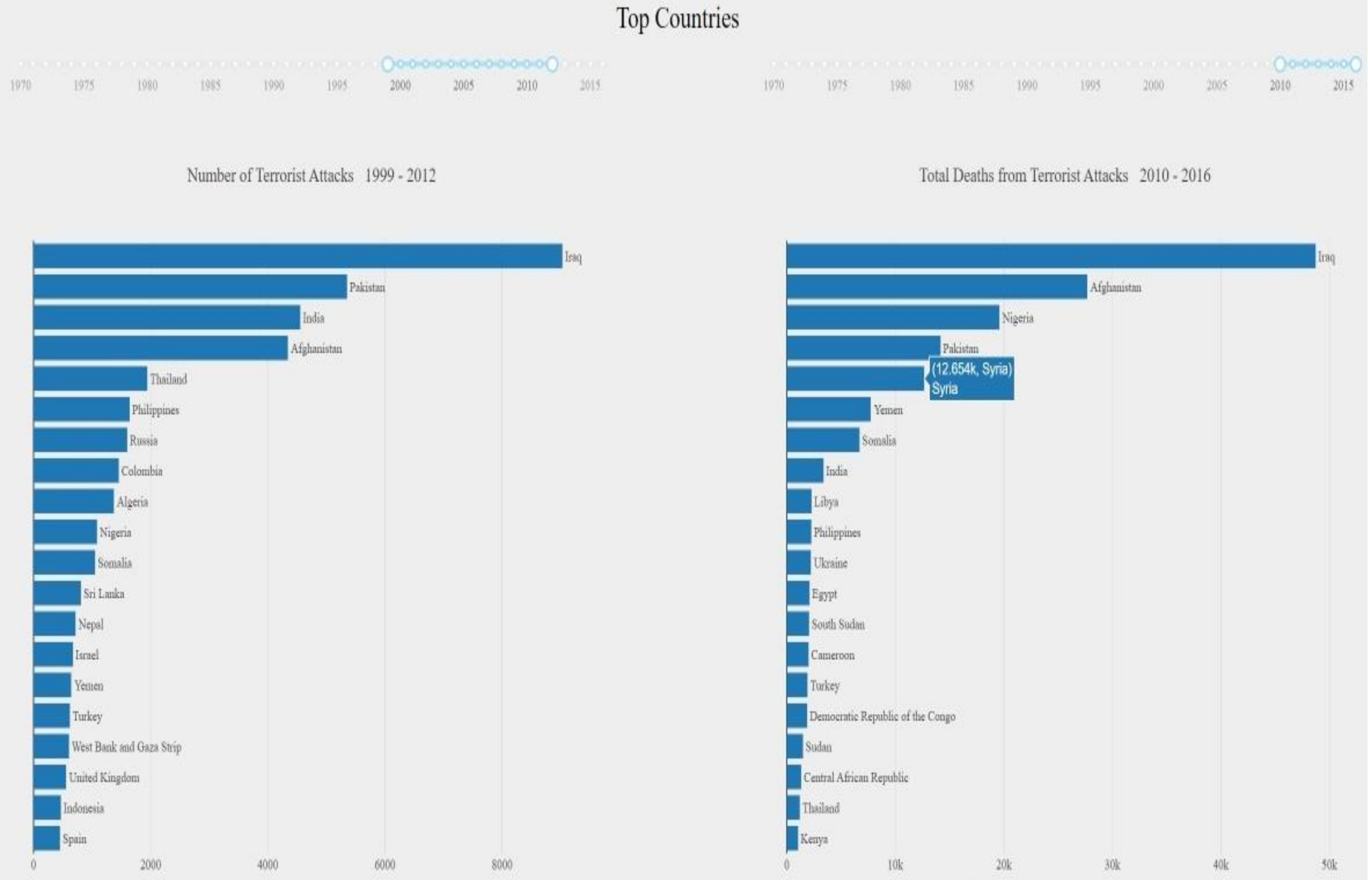
Our dashboard starts with the world map, we can select a particular country or multiple countries from a dropdown menu. This will display the country's map on which we can see the terror attacks in various regions of that country along with the details of the attack.



There is a range slider from which we can select the range of years in which the terror attacks took place.
Below the geospatial graph, we can view the same data in the form of a bar plot; this plot also includes the terror attacks in those given years.



The dashboard also presents two horizontal bar plots which are displaying the number of terror attacks, and total deaths from those attacks in the selected range of years.



CONCLUSION

Our analysis on this Global terrorism database has given us important insights into the major trends related to terrorism. It was crystal clear from these visualizations that terrorism is most prevalent in middle east and Saharan Africa regions. The main reason behind this is the presence of extremist groups in middle eastern countries and local gangs in the African region. Moreover, developed nations are more effective in thwarting(preventing) terror attacks while Developing nations and third world countries face attacks at greater frequency. The analysis of this data is critical to understand where a country's security is lacking and which areas need more attention by the country's security services.