GRIP:The Sparks Foundation

Data Science and Business Analytics Intern

Author: Sudeshna Saha

In [1]:

```
#import all necessary libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:

```
#Load the data
df=pd.read_csv("globalterrorismdb_0718dist.csv",encoding="ISO-8859-1",low_memory=False)
```

In [3]:

```
df.head()
```

Out[3]:

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

5 rows × 135 columns

In [4]:

```
#check the basic information of the data
df.info()
```

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

In [5]:

```
df.shape
```

Out[5]:

```
(181691, 135)
```

In [6]:

```
#checking for the null values of the dataset
total=df.isnull().sum().sort_values(ascending=False)
e=df.isnull().count()/100
percentage=pd.DataFrame((total/e).sort_values(ascending=False)).reset_index()
```

In [7]:

```
percentage.columns=["total","percentage"]
```

In [8]:

```
percentage
```

Out[8]:

	total	percentage
0	gsubname3	99.988992
1	weapsubtype4_txt	99.961473
2	weapsubtype4	99.961473
3	weaptype4_txt	99.959822
4	weaptype4	99.959822
...	...	...
130	crit1	0.000000
131	country_txt	0.000000
132	country	0.000000
133	success	0.000000
134	INT_ANY	0.000000

135 rows × 2 columns

In [9]:

```
a=(percentage[percentage["percentage"]>=40])
```

In [10]:

```
a["total"].values
```

Out[10]:

```
array(['gsubname3', 'weapsubtype4_txt', 'weapsubtype4', 'weaptype4_txt',
       'weaptype4', 'claimmode3', 'claimmode3_txt', 'gsubname2', 'claim3',
       'guncertain3', 'divert', 'gname3', 'attacktype3',
       'attacktype3_txt', 'ransomnote', 'ransompaidus', 'ransomamtus',
       'claimmode2', 'claimmode2_txt', 'ransompaid', 'corp3',
       'targsubtype3_txt', 'targsubtype3', 'natlty3_txt', 'natlty3',
       'target3', 'targtype3_txt', 'targtype3', 'ransomamt',
       'weapsubtype3_txt', 'weapsubtype3', 'weaptype3_txt', 'weaptype3',
       'claim2', 'guncertain2', 'gname2', 'resolution', 'kidhijcountry',
       'nhours', 'compclaim', 'gsubname', 'attacktype2_txt',
       'attacktype2', 'ndays', 'approxdate', 'corp2', 'nreleased',
       'targsubtype2_txt', 'targsubtype2', 'natlty2', 'natlty2_txt',
       'hostkidoutcome', 'hostkidoutcome_txt', 'target2', 'targtype2_txt',
       'targtype2', 'weapsubtype2_txt', 'weapsubtype2', 'weaptype2',
       'weaptype2_txt', 'nhostkidus', 'nhostkid', 'claimmode_txt',
       'claimmode', 'related', 'addnotes', 'alternative',
       'alternative_txt', 'propvalue', 'scite3', 'motive', 'location',
       'propcomment', 'propextent_txt', 'propextent', 'scite2', 'ransom'],
      dtype=object)
```

In [11]:

```
col=['gsubname3', 'weapsubtype4_txt', 'weapsubtype4', 'weaptype4_txt',
       'weaptype4', 'claimmode3', 'claimmode3_txt', 'gsubname2', 'claim3',
       'guncertain3', 'divert', 'gname3', 'attacktype3',
       'attacktype3_txt', 'ransomnote', 'ransompaidus', 'ransomamtus',
       'claimmode2', 'claimmode2_txt', 'ransompaid', 'corp3',
       'targsubtype3_txt', 'targsubtype3', 'natlty3_txt', 'natlty3',
       'target3', 'targtype3_txt', 'targtype3', 'ransomamt',
       'weapsubtype3_txt', 'weapsubtype3', 'weaptype3_txt', 'weaptype3',
       'claim2', 'guncertain2', 'gname2', 'resolution', 'kidhijcountry',
       'nhours', 'compclaim', 'gsubname', 'attacktype2_txt',
       'attacktype2', 'ndays', 'approxdate', 'corp2', 'nreleased',
       'targsubtype2_txt', 'targsubtype2', 'natlty2', 'natlty2_txt',
       'hostkidoutcome', 'hostkidoutcome_txt', 'target2', 'targtype2_txt',
       'targtype2', 'weapsubtype2_txt', 'weapsubtype2', 'weaptype2',
       'weaptype2_txt', 'nhostkidus', 'nhostkid', 'claimmode_txt',
       'claimmode', 'related', 'addnotes', 'alternative',
       'alternative_txt', 'propvalue', 'scite3', 'motive', 'location',
       'propcomment', 'propextent_txt', 'propextent', 'scite2', 'ransom']
```

In [12]:

```
#drop the columns which has more than 40% null values
df.drop(columns=col,axis=1,inplace=True)
```

In [15]:

```
df.shape
```

Out[15]:

```
(181691, 58)
```

In [16]:

```
df.columns
```

Out[16]:

```
Index(['eventid', 'iyear', 'imonth', 'iday', 'extended', 'country',
       'country_txt', 'region', 'region_txt', 'provstate', 'city', 'latitud
e',
       'longitude', 'specificity', 'vicinity', 'summary', 'crit1', 'crit2',
       'crit3', 'doubtterr', 'multiple', 'success', 'suicide', 'attacktype
1',
       'attacktype1_txt', 'targtype1', 'targtype1_txt', 'targsubtype1',
       'targsubtype1_txt', 'corp1', 'target1', 'natlty1', 'natlty1_txt',
       'gname', 'guncertain1', 'individual', 'nperps', 'nperpcap', 'claime
d',
       'weaptype1', 'weaptype1_txt', 'weapsubtype1', 'weapsubtype1_txt',
       'weapdetail', 'nkill', 'nkillus', 'nkillter', 'nwound', 'nwoundus',
       'nwoundte', 'property', 'ishostkid', 'scite1', 'dbsource', 'INT_LOG',
       'INT_IDEO', 'INT_MISC', 'INT_ANY'],
      dtype='object')
```

In [17]:

```
col2=["eventid","nwoundte","guncertain1","success","suicide","individual","nperps","claimed
```

In [18]:

```
df.drop(columns=col2,axis=1,inplace=True)
```

In [19]:

```
#Let's check the final shape of the data
df.shape
```

Out[19]:

```
(181691, 19)
```

In [20]:

```
#the final data after dropping the columns
df.head()
```

Out[20]:

	iyear	imonth	iday	country_txt	region_txt	provstate	city	latitude	longitude	
0	1970	7	2	Dominican Republic	Central America & Caribbean	NaN	Santo Domingo	18.456792	-69.951164	
1	1970	0	0	Mexico	North America	Federal	Mexico city	19.371887	-99.086624	
2	1970	1	0	Philippines	Southeast Asia	Tarlac	Unknown	15.478598	120.599741	
3	1970	1	0	Greece	Western Europe	Attica	Athens	37.997490	23.762728	B
4	1970	1	0	Japan	East Asia	Fukouka	Fukouka	33.580412	130.396361	Fac

In [21]:

```
#checking the summary of the data
df.describe()
```

Out[21]:

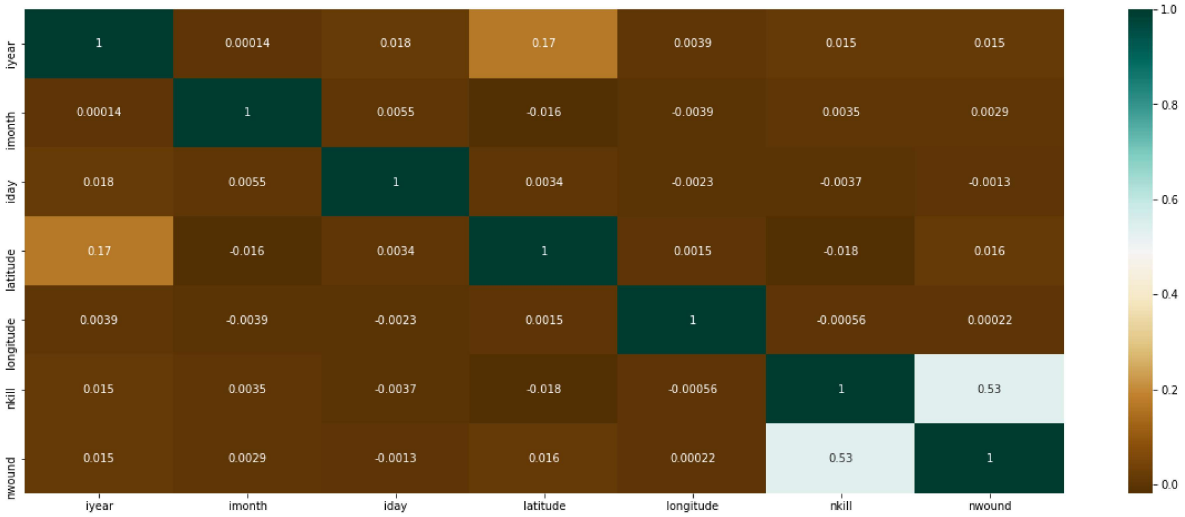
	iyear	imonth	iday	latitude	longitude	nl
count	181691.000000	181691.000000	181691.000000	177135.000000	1.771340e+05	171378.000(
mean	2002.638997	6.467277	15.505644	23.498343	-4.586957e+02	2.4032
std	13.259430	3.388303	8.814045	18.569242	2.047790e+05	11.5457
min	1970.000000	0.000000	0.000000	-53.154613	-8.618590e+07	0.000(
25%	1991.000000	4.000000	8.000000	11.510046	4.545640e+00	0.000(
50%	2009.000000	6.000000	15.000000	31.467463	4.324651e+01	0.000(
75%	2014.000000	9.000000	23.000000	34.685087	6.871033e+01	2.000(
max	2017.000000	12.000000	31.000000	74.633553	1.793667e+02	1570.000(

In [22]:

```
#check the correlations
plt.figure(figsize=(21,8))
sns.heatmap(df.corr(),cmap="BrBG",annot=True)
```

Out[22]:

```
<AxesSubplot:>
```



In [23]:

```
#removing duplicate records
df.drop_duplicates(keep='first',inplace=True)
df.shape
```

Out[23]:

```
(171259, 19)
```

In [24]:

```
#convert the dataframe into excel format
df.to_excel(r'C:\Users\LENOVO\Desktop\jupyter\Terrorist.xlsx',index=False)
```

In [ ]:

```

```