ACLED Middle East Geomapping

Moses A. Boudourides

```
import pandas as pd
import folium
from folium import plugins
import warnings
warnings.filterwarnings("ignore", category=RuntimeWarning)
warnings.filterwarnings("ignore", category=UserWarning)
warnings.simplefilter('ignore')
```

```
df = pd.read_csv("MiddleEast_2017-2018_June12.csv")
df.head(4)
```

	ISO	EVENT_ID_CNTY	EVENT_ID_NO_CNTY	EVENT_DATE	YEAR	TIME_PRECISION	
0	48	BHR2	2.0	01-January-2017	2017	1	
1	48	BHR1	1.0	01-January-2017	2017	1	
2	48	BHR3	3.0	01-January-2017	2017	1	
3	48	BHR4	4.0	02-January-2017	2017	1	

4 rows × 28 columns

```
print(len(df),'\n')
df['EVENT_TYPE'].value_counts()
```

```
68630



Remote violence                    40053
Battle-No change of territory      14656
Riots/Protests                      4659
```

```
Strategic development                           2755
Battle-Government regains territory             2582
Violence against civilians                      2057
Battle-Non-state actor overtakes territory      1397
Non-violent transfer of territory                431
Headquarters or base established                  30
Battle-no change of territory                      6
Battles-No change of territory                     4
Name: EVENT_TYPE, dtype: int64
```

```
df['COUNTRY'].value_counts()
```

```
Syria                   31374
Yemen                   20715
Iraq                     6672
Saudi Arabia             3274
Palestine                2103
Turkey                   1847
Iran                     1310
Israel                    445
Lebanon                   442
Bahrain                   307
Jordan                    130
Kuwait                      6
Oman                        2
United Arab Emirates        2
Qatar                       1
Name: COUNTRY, dtype: int64
```

```
df['YEAR'].value_counts()
```

```
2017    39904
2018    19012
2016     9714
Name: YEAR, dtype: int64
```

```
dfc=df.copy()
dfs=dfc.loc[dfc['COUNTRY'].isin(['Syria','Palestine','Israel','Jordan','Lebanon'])]
print(len(dfs),'\n')
dfs['EVENT_TYPE'].value_counts()
```

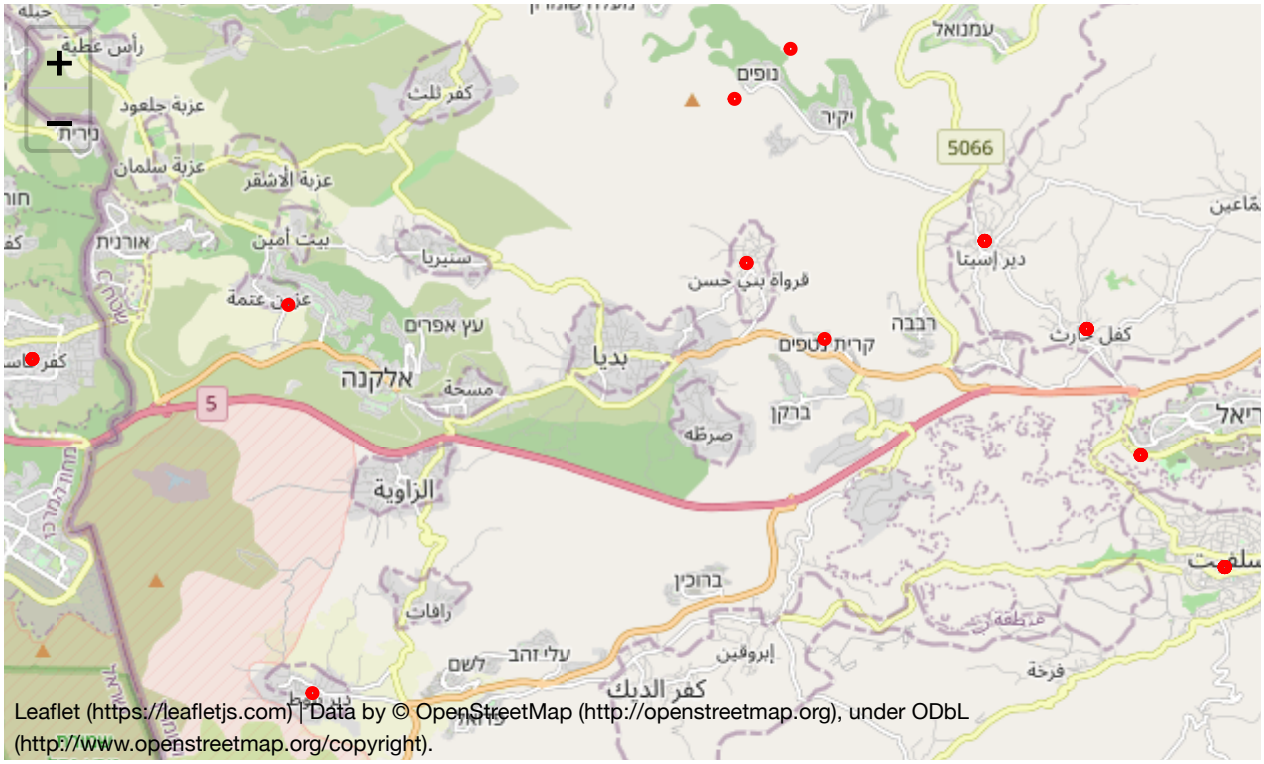
```
34494



Remote violence                           19577
Battle-No change of territory             7261
Riots/Protests                             2212
Strategic development                      1541
Battle-Non-state actor overtakes territory 1267
Battle-Government regains territory        1221
Violence against civilians                 1059
Non-violent transfer of territory           324
Headquarters or base established             26
```

```
Battle-no change of territory                    6
Name: EVENT_TYPE, dtype: int64
```

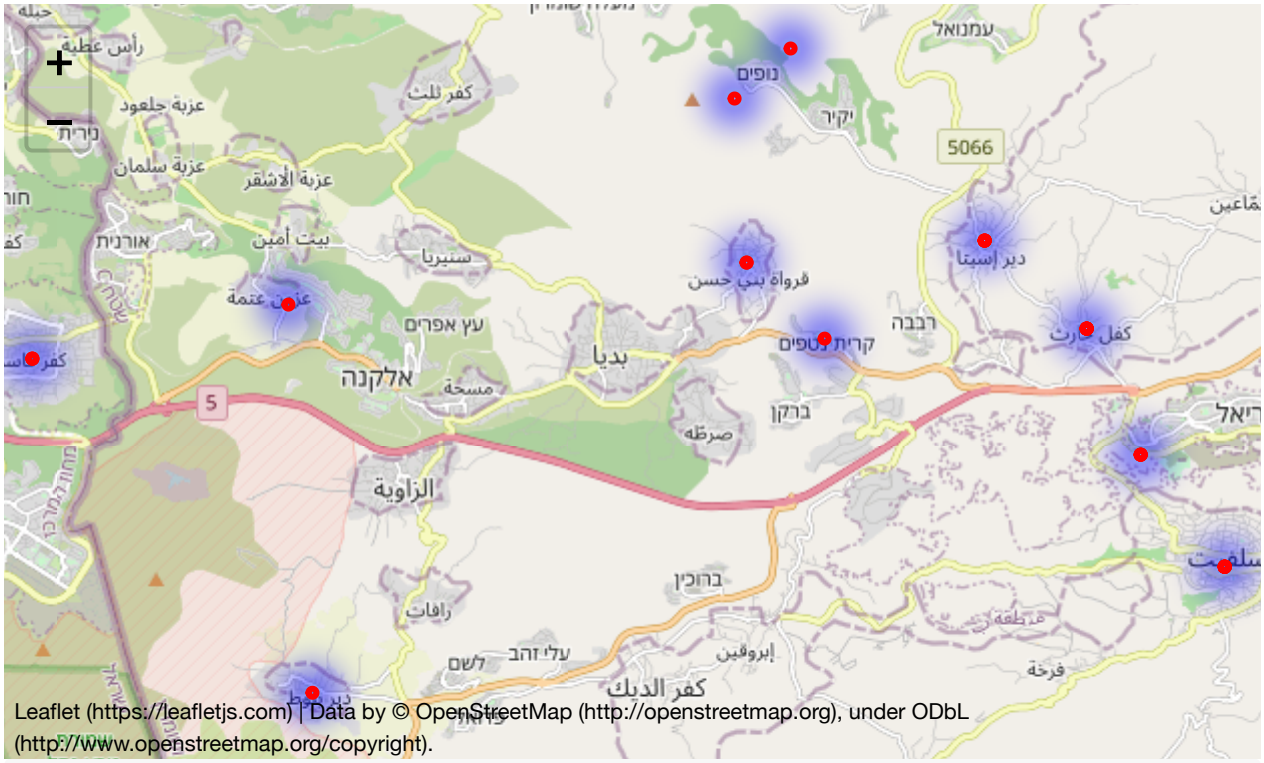
```
dfc=df.copy()
dfs=dfc.loc[dfc['COUNTRY'].isin(['Palestine','Israel','Jordan','Lebanon'])]
print(len(dfs),'\n')
dfs['EVENT_TYPE'].value_counts()
```

```
3120



Riots/Protests                                1942
Strategic development                          395
Violence against civilians                     287
Remote violence                                273
Battle-No change of territory                  190
Battle-Non-state actor overtakes territory      10
Battle-Government regains territory              9
Non-violent transfer of territory                8
Battle-no change of territory                    6
Name: EVENT_TYPE, dtype: int64
```

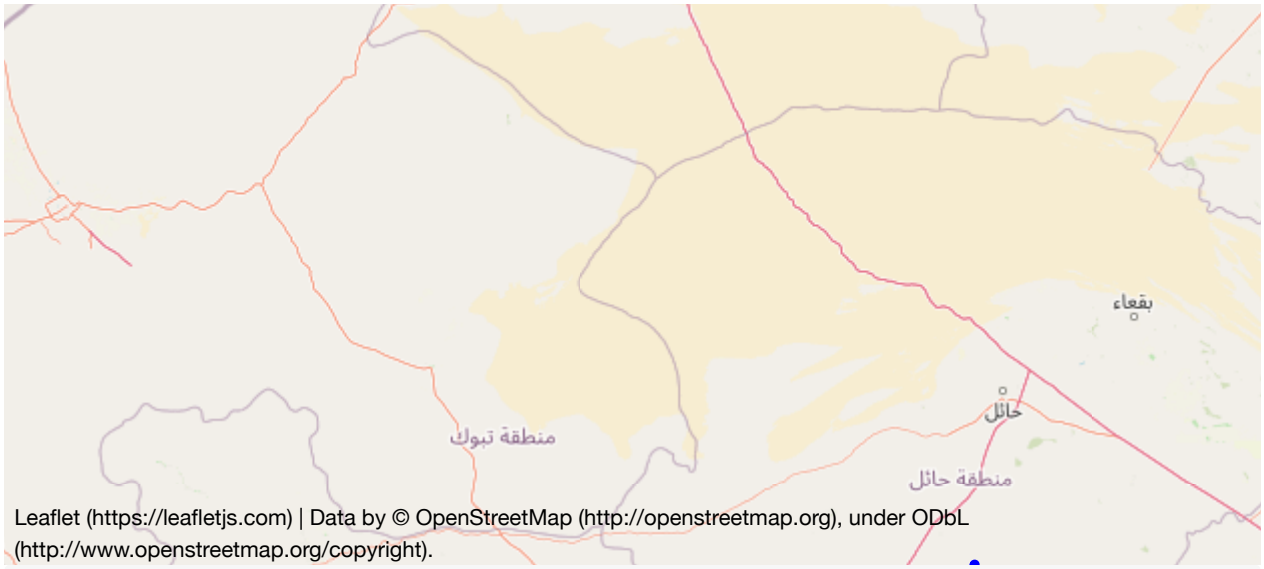
```
map_center = [dfs["LATITUDE"].mean(), dfs["LONGITUDE"].mean()]
map_3 = folium.Map(location=map_center, zoom_start=12)
for i, row in dfs[["LATITUDE", "LONGITUDE"]].dropna().iterrows():
    position = (row["LATITUDE"], row["LONGITUDE"])
    folium.CircleMarker(position,radius=2,color="red").add_to(map_3)
map_3.save('html_map_output/ACLED_ME_pns.html')
map_3
```



```
hm=dfs[["LATITUDE", "LONGITUDE"]].as_matrix()
hm_c = [x for x in hm if str(x[0]) != 'nan' and str(x[1]) != 'nan']
hm_c
map_3.add_child(plugins.HeatMap(hm_c,radius=15))
map_3.save('html_map_output/ACLED_ME_pns_heatmap.html')
map_3
```




```
dfc=df.copy()
dfs=dfc

def color_producer(EVENT_TYPE):
    if EVENT_TYPE=="Remote violence": #":
        return 'red'
    elif EVENT_TYPE=="Battle-No change of territory": #":
        return 'blue'
    elif EVENT_TYPE=="Riots/Protests": #Violence against civilians":
        return 'green'
    elif EVENT_TYPE=="Strategic development":
        return 'cyan'
    elif EVENT_TYPE=="Battle-Government regains territory": #":
        return 'orange'
    elif EVENT_TYPE=="Violence against civilians":
        return 'magenta'
    else:
        return 'yellow'
map_center = [dfs["LATITUDE"].mean(), dfs["LONGITUDE"].mean()]
map_4 = folium.Map(location=map_center, zoom_start=7)
for i, row in dfs[["EVENT_TYPE","LATITUDE", "LONGITUDE"]].dropna().iterrows():
    position = (row["LATITUDE"], row["LONGITUDE"])
    folium.CircleMarker(position,radius=1,color=color_producer(row['EVENT_TYPE'])).add_to(map_4) #row['marker
map_4.save('html_map_output/ACLED_ME_pns_Events.html')
map_4
```


بقعاء
حائل
منطقة تبوك
منطقة حائل
Leaflet (https://leafletjs.com) | Data by © OpenStreetMap (http://openstreetmap.org), under ODbL (http://www.openstreetmap.org/copyright).