Module 7: Developing Web Maps and Representing information using Plots

Case Study II

1. Display first 100 records from the data and play with parameters - tiles, zoom\_start etc. Map will be saved in BasicWebMap.html – view it in browser

**Solution:**

import pandas as pd

crimeData = pd.read\_csv("Police\_Department\_Incidents\_Year\_2016\_.csv")

crimeData\_100 = crimeData.head(100)

cat = list(crimeData\_100['Category'])

lat = list(crimeData\_100['Y'])

lon = list(crimeData\_100['X'])

import folium

map = folium.Map(location =[37.773972, -122.431297], zoom\_start=50)

fg = folium.FeatureGroup(name="SanFrancisco")

fg.add\_child(folium.Marker(location=[37.773972, -122.431297], popup="SanFrancisco(SFO) City", icon=folium.Icon(color='green')))

map.add\_child(fg)

fgc = folium.FeatureGroup(name="Crime\_Category")

for lt, ln, cy in zip(lat, lon, cat):

fgc.add\_child(folium.Marker(location=[lt, ln], popup=str(cy), icon=folium.Icon(color="red")))

map.add\_child(fgc)

map.add\_child(folium.LayerControl())

map.save("BasicWebMap.html")

2. For latest 7 days of data create WebMap of Crimes which are categorized as ROBBERY

**Solution:**

crimeData['ChangeDate'] = pd.to\_datetime(crimeData['Date'])

crimeData\_Ind = crimeData.set\_index('ChangeDate')

df = crimeData\_Ind.sort\_index()

crimeData\_Lat7 = df[df.last\_valid\_index()-pd.DateOffset(7, 'D'):].reset\_index()

selectColumn = crimeData\_Lat7.loc[:,["ChangeDate", "Category", "X", "Y"]]

df\_robbery = selectColumn[selectColumn["Category"]=="ROBBERY"]

lat = list(df\_robbery['Y'])

lon = list(df\_robbery['X'])

import folium

map = folium.Map(location =[37.773972, -122.431297], zoom\_start=100,)

fgr = folium.FeatureGroup(name="Latest 7 days Robbery")

for lt, ln in zip(lat, lon):

fgr.add\_child(folium.Marker(location=[lt, ln], popup="ROBBERY", icon=folium.Icon(color="red")))

map.add\_child(fgr)

map.add\_child(folium.LayerControl())

3. For latest 15 days of data create One WebMap of Crimes which are categorized as FRAUD and GAMBLING. Change the icon to font awesome icon (http://fontawesome.io/icons/)

**Solution:**

import pandas as pd

crimeData = pd.read\_csv("Police\_Department\_Incidents\_Year\_2016\_.csv")

crimeData['ChangeDate'] = pd.to\_datetime(crimeData['Date'])

crimeData\_Ind = crimeData.set\_index('ChangeDate')

df = crimeData\_Ind.sort\_index()

crimeData\_Lat15 = df[df.last\_valid\_index()-pd.DateOffset(15, 'D'):].reset\_index()

selectColumn = crimeData\_Lat15.loc[:,["ChangeDate", "Category", "X", "Y"]]

filter\_list = ["FRAUD", "GAMBLING"]

df = selectColumn[selectColumn.Category.isin(filter\_list)]

lat = list(df['Y'])

lon = list(df['X'])

cat = list(df['Category'])

import folium

map = folium.Map(location =[37.773972, -122.431297], zoom\_start=100,)

fgb = folium.FeatureGroup(name="Latest 15 days Fraud and No Gambling")

for lt, ln, cy in zip(lat, lon, cat):

fgb.add\_child(folium.Marker(location=[lt, ln], popup=str(cy), icon=folium.Icon(color="green", prefix='fab', icon='fa-cloudversify')))

map.add\_child(fgb)

map.add\_child(folium.LayerControl())

4. BONUS ASSIGNMENT -- Display heatmap for Divvy Bikes. Divvy Bikes runs bike rental service in Chicago and their bike station geolocation data is shared. Refer to BikesHeatMap.py and Divvy\_Stations\_2016\_Q3.csv, Divvy\_Stations\_2016\_Q4.csv

**Solution:**

import folium

from folium import plugins

import pandas as pd

divvyStations\_q3 = pd.read\_csv('Divvy\_Stations\_2016\_Q3.csv')

divvyStations\_q4 = pd.read\_csv('Divvy\_Stations\_2016\_Q4.csv')

divvyStations = pd.concat([divvyStations\_q3, divvyStations\_q4], axis=0).drop\_duplicates(subset=['id'])

CHICAGO\_COORD = [41.8781, -87.6298]

map\_heat = folium.Map(CHICAGO\_COORD,

zoom\_start=11)

stationArr = [[row['latitude'],row['longitude']] for index, row in divvyStations.iterrows()]

map\_heat.add\_child(plugins.HeatMap(stationArr, radius=15))

map\_heat.save("BikeStationHeatMap.html")

Enhancements for code

You can try these enhancements in code

1. Lot of visual options are available to change the map. You can refer to https://python-visualization.github.io/folium/modules.html for options

**Solution:**

map\_test = folium.Map(location =[37.773972, -122.431297], zoom\_start=50, tiles=" OpenStreetMap")

2. Change the code to create webmap for 'BURGALRY ' for last 7 days

**Solution:**

import pandas as pd

crimeData = pd.read\_csv("Police\_Department\_Incidents\_Year\_2016\_.csv")

crimeData['ChangeDate'] = pd.to\_datetime(crimeData['Date'])

crimeData\_Ind = crimeData.set\_index('ChangeDate')

df = crimeData\_Ind.sort\_index()

crimeData\_Lat7 = df[df.last\_valid\_index()-pd.DateOffset(7, 'D'):].reset\_index()

selectColumn = crimeData\_Lat7.loc[:,["ChangeDate", "Category", "X", "Y"]]

df\_burglary = selectColumn[selectColumn["Category"]=="BURGLARY"]

lat = list(df\_burglary['Y'])

lon = list(df\_burglary['X'])

import folium

map = folium.Map(location =[37.773972, -122.431297], zoom\_start=100,)

fgr = folium.FeatureGroup(name="Last 7 days Burglary")

for lt, ln in zip(lat, lon):

fgr.add\_child(folium.Marker(location=[lt, ln], popup="Burglary", icon=folium.Icon(color="red")))

map.add\_child(fgr)

map.add\_child(folium.LayerControl())

3. BONUS ASSIGNMENT – Uncomment the code see the changes

**Solution:**

for index, row in divvyStations.iterrows():

folium.CircleMarker([row['latitude'], row['longitude']],

radius=15,

popup=row['name'],

fill\_color="#3db7e4", # divvy color

).add\_to(map\_heat)

map\_heat