

Erindale SS hackathon notebook

February 9, 2020

```
In [1]: import json
```

```
In [2]: with open("erindaledata.json") as json_file:
        data = json.load(json_file)
```

```
In [3]: type(data['data'])
```

```
Out[3]: list
```

```
In [4]: len(data['data'])
```

```
Out[4]: 189
```

```
In [5]: data['data'][0]
```

```
Out[5]: {'temperature': 15,
        'humidity': 45,
        'noise_level': 585,
        'wifiAccessPoints': [{'macAddress': '88:75:56:AF:20:10',
                               'ssid': 'PDSB-ADMIN',
                               'signalStrength': -58},
                              {'macAddress': '88:75:56:AF:20:14',
                               'ssid': 'PDSB-MEDIA',
                               'signalStrength': -58},
                              {'macAddress': '88:75:56:AF:20:11',
                               'ssid': 'PDSB-WiFi',
                               'signalStrength': -58},
                              {'macAddress': '88:75:56:AF:20:15',
                               'ssid': 'PDSB-GUEST',
                               'signalStrength': -58},
                              {'macAddress': '88:75:56:57:0C:94',
                               'ssid': 'PDSB-MEDIA',
                               'signalStrength': -89},
                              {'macAddress': '88:75:56:57:0C:95',
                               'ssid': 'PDSB-GUEST',
                               'signalStrength': -87},
                              {'macAddress': '88:75:56:87:6C:A4',
                               'ssid': 'PDSB-MEDIA',
                               'signalStrength': -83}],
        }
```

```

    {'macAddress': '20:3A:07:48:1A:00',
      'ssid': 'PDSB-ADMIN',
      'signalStrength': -55},
    {'macAddress': '20:3A:07:48:1A:04',
      'ssid': 'PDSB-MEDIA',
      'signalStrength': -54},
    {'macAddress': '88:75:56:87:6C:A0',
      'ssid': 'PDSB-ADMIN',
      'signalStrength': -82},
    {'macAddress': '20:3A:07:48:1A:05',
      'ssid': 'PDSB-GUEST',
      'signalStrength': -56},
    {'macAddress': '88:75:56:87:6C:A5',
      'ssid': 'PDSB-GUEST',
      'signalStrength': -83},
    {'macAddress': '20:3A:07:48:1A:01',
      'ssid': 'PDSB-WiFi',
      'signalStrength': -56},
    {'macAddress': '20:3A:07:6F:40:B5',
      'ssid': 'PDSB-GUEST',
      'signalStrength': -83},
    {'macAddress': '20:3A:07:48:18:21',
      'ssid': 'PDSB-WiFi',
      'signalStrength': -80},
    {'macAddress': '20:3A:07:6F:40:B1',
      'ssid': 'PDSB-WiFi',
      'signalStrength': -85},
    {'macAddress': '20:3A:07:48:18:25',
      'ssid': 'PDSB-GUEST',
      'signalStrength': -81},
    {'macAddress': '20:3A:07:48:18:24',
      'ssid': 'PDSB-MEDIA',
      'signalStrength': -80},
    {'macAddress': '20:3A:07:6F:40:B0',
      'ssid': 'PDSB-ADMIN',
      'signalStrength': -84},
    {'macAddress': '20:3A:07:6F:40:B4',
      'ssid': 'PDSB-MEDIA',
      'signalStrength': -84}],
    'lat': 43.538797,
    'lng': -79.6664817,
    'accuracy': 35,
    'time': '1/20/2020, 7:51:10 AM'}

```

```
In [6]: reduced = []
```

```
In [7]: for d in data['data']:
        temperature=d['temperature']
```

```

humidity = d['humidity']
noise_level=d['noise_level']
lat=d['lat']
lon=d['lng']
accuracy = d['accuracy']
time = d['time']
strength = 0
count=0.
for wap in d['wifiAccessPoints']:
    strength += wap['signalStrength']
    count+=1.

average_strength = strength/count

reduced.append({
    'temperature':temperature,
    'humidity':humidity,
    'noise_level':noise_level,
    'accuracy':accuracy,
    'wifi':strength,
    'time':time,
    'latitude':lat,
    'longitude':lon
})

```

In [8]: `import pandas as pd`

In [9]: `df = pd.DataFrame(reduced)`

In [10]: `df`

```

Out[10]:
   temperature  humidity  noise_level  accuracy  wifi  \
0             15         45          585         35 -1454
1             19         18          582         33 -1351
2             20         17          583         32 -1362
3             20         17          663         22 -1218
4             20         18          668         20  -565
..          ...         ...          ...         ...  ...
184           21         17          677         20 -1091
185           20         18          678         20  -781
186           20         17          676         20  -843
187           20         18          676         20 -1329
188           20         17          674         20 -1232

   time          latitude  longitude
0  1/20/2020, 7:51:10 AM  43.538797 -79.666482
1  1/20/2020, 7:51:24 AM  43.538823 -79.666474
2  1/20/2020, 7:51:33 AM  43.538818 -79.666444

```

```

3      1/20/2020, 7:52:50 AM  43.538894 -79.666039
4      1/20/2020, 7:52:59 AM  43.538940 -79.666016
..      ...
184    1/20/2020, 2:49:43 PM  43.538791 -79.666584
185    1/20/2020, 2:49:52 PM  43.538801 -79.666574
186    1/20/2020, 2:50:00 PM  43.538789 -79.666579
187    1/20/2020, 2:50:09 PM  43.538802 -79.666590
188    1/20/2020, 2:50:29 PM  43.538802 -79.666590

```

[189 rows x 8 columns]

In [11]: `!pip install folium geopandas`

```

Requirement already satisfied: folium in /anaconda3/lib/python3.6/site-packages (0.10.1)
Requirement already satisfied: geopandas in /anaconda3/lib/python3.6/site-packages (0.6.3)
Requirement already satisfied: branca>=0.3.0 in /anaconda3/lib/python3.6/site-packages (from folium) (0.3.0)
Requirement already satisfied: numpy in /anaconda3/lib/python3.6/site-packages (from folium) (1.15.4)
Requirement already satisfied: requests in /anaconda3/lib/python3.6/site-packages (from folium) (2.18.4)
Requirement already satisfied: Jinja2>=2.9 in /anaconda3/lib/python3.6/site-packages (from folium) (2.10.1)
Requirement already satisfied: pyproj in /anaconda3/lib/python3.6/site-packages (from geopandas) (2.1.1)
Requirement already satisfied: shapely in /anaconda3/lib/python3.6/site-packages (from geopandas) (1.7.1)
Requirement already satisfied: Fiona in /anaconda3/lib/python3.6/site-packages (from geopandas) (1.7.0)
Requirement already satisfied: pandas>=0.23.0 in /anaconda3/lib/python3.6/site-packages (from geopandas) (0.23.4)
Requirement already satisfied: six in /anaconda3/lib/python3.6/site-packages (from branca>=0.3.0) (1.11.0)
Requirement already satisfied: urllib3!=1.25.0,!1.25.1,<1.26,>=1.21.1 in /anaconda3/lib/python3.6/site-packages (from requests) (1.24.2)
Requirement already satisfied: certifi>=2017.4.17 in /anaconda3/lib/python3.6/site-packages (from requests) (2018.11.27)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /anaconda3/lib/python3.6/site-packages (from requests) (3.0.4)
Requirement already satisfied: idna<2.9,>=2.5 in /anaconda3/lib/python3.6/site-packages (from requests) (2.8)
Requirement already satisfied: MarkupSafe>=0.23 in /anaconda3/lib/python3.6/site-packages (from Jinja2>=2.9) (1.0)
Requirement already satisfied: cligj>=0.5 in /anaconda3/lib/python3.6/site-packages (from Fiona) (0.5.0)
Requirement already satisfied: attrs>=17 in /anaconda3/lib/python3.6/site-packages (from Fiona) (17.4.0)
Requirement already satisfied: munch in /anaconda3/lib/python3.6/site-packages (from Fiona) (2.5.0)
Requirement already satisfied: click-plugins>=1.0 in /anaconda3/lib/python3.6/site-packages (from click) (1.1.1)
Requirement already satisfied: click<8,>=4.0 in /anaconda3/lib/python3.6/site-packages (from click) (7.0)
Requirement already satisfied: python-dateutil>=2.6.1 in /anaconda3/lib/python3.6/site-packages (from pandas>=0.23.0) (2.6.1)
Requirement already satisfied: pytz>=2017.2 in /anaconda3/lib/python3.6/site-packages (from pandas>=0.23.0) (2018.9)
WARNING: You are using pip version 19.3.1; however, version 20.0.2 is available. You should consider upgrading with the following command:

```

```

In [12]: import folium
         from folium.plugins import HeatMap

```

0.1 Temperature

```

In [13]: hmap1 = folium.Map(location=[43.538797, -79.666482], zoom_start=28., )
         df.temperature = df.temperature.astype(float)
         temperature_max = float(df['temperature'].max())

         hm_temp = HeatMap( list(zip(df.latitude.values, df.longitude.values, df.temperature.values))

```

```

        min_opacity=0.4,
        max_val=temperature_max,
        radius=25., blur=15.,
        max_zoom=35.,
    )
    hmap1.add_child(hm_temp)

```

Out[13]: <folium.folium.Map at 0x113edc908>

0.2 Humidity

```

In [14]: hmap2 = folium.Map(location=[43.538797, -79.666482], zoom_start=28., )
        df.humidity = df.humidity.astype(float)
        humidity_max = float(df['humidity'].max())
        print(humidity_max)
        hm_humid = HeatMap( list(zip(df.latitude.values, df.longitude.values, df.humidity.values))
                            min_opacity=0.4,
                            max_val=humidity_max,
                            radius=30., blur=35.,
                            max_zoom=35.,
                        )

        hmap2.add_child(hm_humid)

```

48.0

Out[14]: <folium.folium.Map at 0x113ede908>

0.3 Noise Level

```

In [15]: hmap3 = folium.Map(location=[43.538797, -79.666482], zoom_start=28., )
        df.noise_level = df.noise_level.astype(float)
        noise_level_max = float(df['noise_level'].max())
        print(noise_level_max)

        hm_noise = HeatMap( list(zip(df.latitude.values, df.longitude.values, df.noise_level.values))
                            min_opacity=0.4,
                            max_val=noise_level_max,
                            radius=30., blur=35.,
                            max_zoom=35.,
                        )

        hmap3.add_child(hm_noise)

```

687.0

Out[15]: <folium.folium.Map at 0x113f211d0>

0.4 Wifi

```
In [16]: hmap4 = folium.Map(location=[43.538797, -79.666482], zoom_start=28., )
df.wifi = df.wifi.astype(float)
wifi_max = float(df['wifi'].max())
print(wifi_max)

hm_wifi = HeatMap( list(zip(df.latitude.values, df.longitude.values, df.wifi.values))
                    min_opacity=0.2,
                    max_val=wifi_max,
                    radius=30., blur=35.,
                    max_zoom=35.,
                    )

hmap4.add_child(hm_wifi)

-199.0
```

```
Out[16]: <folium.folium.Map at 0x113f21dd8>
```

0.5 Best Wi-fi and least noise (A.K.A best place to work)

```
In [17]: normalized_wifi = (-df['wifi'].min() + df['wifi'])
normalized_wifi = normalized_wifi/normalized_wifi.max()

In [18]: normalized_noise = (-df['noise_level'].min() + df['noise_level'])
normalized_noise = normalized_noise/normalized_noise.max()
```

Write down an equation for max wifi strength and minimal noise ...

```
In [19]: composite_score = 5.* normalized_wifi / normalized_noise

In [20]: comp_max = composite_score.max()

In [21]: hmap5 = folium.Map(location=[43.538797, -79.666482], zoom_start=28., )

hm_comp = HeatMap( list(zip(df.latitude.values, df.longitude.values, composite_score.values))
                    min_opacity=0.2,
                    max_val=wifi_max,
                    radius=20., blur=35.,
                    max_zoom=35.,
                    )

hmap5.add_child(hm_comp)
```

```
Out[21]: <folium.folium.Map at 0x1148c6f60>
```

0.6 Head over to the orange/red areas for the best experience in school

.... And also take a poll on Temperature/humidity and set these to the optimum values
:)