

# Interactive Restaurant Density Mapping using and IPython Documentation



**Step 1** - Installing **folium** module. You can do it inside Jupyter Notebook as shown below

```
: !pip install folium

Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: folium in c:\users\deepraj\appdata\roaming\python\python312\site-packages (0.19.5)
Requirement already satisfied: branca>=0.6.0 in c:\users\deepraj\appdata\roaming\python\python312\site-packages (from folium) (0.8.1)
Requirement already satisfied: Jinja2>=2.9 in c:\programdata\anaconda3\lib\site-packages (from folium) (3.1.4)
Requirement already satisfied: numpy in c:\programdata\anaconda3\lib\site-packages (from folium) (1.26.4)
Requirement already satisfied: requests in c:\programdata\anaconda3\lib\site-packages (from folium) (2.32.3)
Requirement already satisfied: xyzservices in c:\programdata\anaconda3\lib\site-packages (from folium) (2022.9.0)
Requirement already satisfied: MarkupSafe>=2.0 in c:\programdata\anaconda3\lib\site-packages (from Jinja2>=2.9->folium) (2.1.3)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\programdata\anaconda3\lib\site-packages (from requests->folium) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\programdata\anaconda3\lib\site-packages (from requests->folium) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\programdata\anaconda3\lib\site-packages (from requests->folium) (2.2.3)
Requirement already satisfied: certifi>=2017.4.17 in c:\programdata\anaconda3\lib\site-packages (from requests->folium) (2024.12.14)
```

**Step 2** - Installing **IPython** module. You can do it inside Jupyter Notebook as shown below

```
!pip install IPython
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: IPython in c:\programdata\anaconda3\lib\site-packages (8.27.0)
Requirement already satisfied: decorator in c:\programdata\anaconda3\lib\site-packages (from IPython) (5.1.1)
Requirement already satisfied: jedi>=0.16 in c:\programdata\anaconda3\lib\site-packages (from IPython) (0.19.1)
Requirement already satisfied: matplotlib-inline in c:\programdata\anaconda3\lib\site-packages (from IPython) (0.1.6)
Requirement already satisfied: prompt-toolkit<3.1.0,>=3.0.41 in c:\programdata\anaconda3\lib\site-packages (from IPython) (3.0.43)
Requirement already satisfied: pygments>=2.4.0 in c:\programdata\anaconda3\lib\site-packages (from IPython) (2.15.1)
Requirement already satisfied: stack-data in c:\programdata\anaconda3\lib\site-packages (from IPython) (0.2.0)
Requirement already satisfied: traitlets>=5.13.0 in c:\programdata\anaconda3\lib\site-packages (from IPython) (5.14.3)
Requirement already satisfied: colorama in c:\programdata\anaconda3\lib\site-packages (from IPython) (0.4.6)
Requirement already satisfied: parso<0.9.0,>=0.8.3 in c:\programdata\anaconda3\lib\site-packages (from jedi>=0.16->IPython) (0.8.3)
Requirement already satisfied: wcwidth in c:\programdata\anaconda3\lib\site-packages (from prompt-toolkit<3.1.0,>=3.0.41->IPython) (0.2.5)
Requirement already satisfied: executing in c:\programdata\anaconda3\lib\site-packages (from stack-data->IPython) (0.8.3)
Requirement already satisfied: asttokens in c:\programdata\anaconda3\lib\site-packages (from stack-data->IPython) (2.0.5)
Requirement already satisfied: pure-eval in c:\programdata\anaconda3\lib\site-packages (from stack-data->IPython) (0.2.2)
Requirement already satisfied: six in c:\programdata\anaconda3\lib\site-packages (from asttokens->stack-data->IPython) (1.16.0)
```

### Step3 - Reading the csv data into a dataframe.

```
: # Importing the Dataset
df = pd.read_csv("zomato_data_final.csv")
```

### Step 3 - Import required library - folium and IPython

```
# Importing Libraries
import pandas as pd
from folium.plugins import MarkerCluster
import folium
from IPython.display import IFrame
```

### Step 4 - Restaurant Density Visualization using Folium

#### 1.Objectivec

The main aim is to visualize restaurant locations on a map to understand their density in a city.

#### 2.Map Initialization

We start by creating a map centered on the city (e.g., Bangalore) using its latitude and longitude.

### 3. Add Marker Clustering

To avoid overlapping markers, we use clustering. This groups nearby restaurants into clusters that can be clicked and zoomed into.

### 4. Plot Restaurant Data

We loop through the restaurant data and add a marker for each one with a valid latitude and longitude.

---

### 5. Show Restaurant Details

Each marker displays a popup with details like:

- City name
- Restaurant rating
- Approximate cost for two people

### 6. Save the Map

The map is saved as an HTML file so it can be opened and viewed in any browser.

```
# Loading data with folium map data performing the resutaurant density

map1 = folium.Map(location=[12.9716, 77.5946], zoom_start=12)
marker_cluster = MarkerCluster().add_to(map1)

for i, row in df.iterrows():
    if not pd.isna(row['Latitude']) and not pd.isna(row['Longitude']):
        folium.Marker([
            location=[row['Latitude'], row['Longitude']],
            popup=f"{row['listed_incity']}<br>Rating: {row['rate']}<br>Cost: ₹{row['approx_costfor_two_people']}",
        ]).add_to(marker_cluster)

map1.save("restaurant_density.html")
```

---

