Easy Steps To Plot Geographic Data on a Map — Python

Ahmed Qassim Follow
Aug 16, 2019 · 3 min read ★

Assume that you are working in a startup and you need to conduct spatial data analysis and prediction to users' geographical data. Or your company runs a lot of delivery operations and your job again to analyze, visualize and maybe predict the drivers or users' geographical data. So, visualizing your data (predicted ones maybe) on a map will be very necessary.

In this article, I will go through easy steps of how to plot geographic data on any map using Python. The thing that I found it very useful and helpful in my previous projects using the same language: Python- check my article: Spatial Data Analysis for Traffic Management.

Off course, when we mention geographical data it crosses to our mind the coordinates of a data point which are: Longitude and Latitude. This is true, they are just the X and Y coordinates for a specific point on the map. However, there are other types of geographic data, such as polygon, line data. The main focus here will be on how to visualize points data on a map. shall we begin?

Loading Libraries and Dataset

First, let us start by Loading the libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

Loading the Spatial Dataset

```
df = pd.read_csv('C:/.. …/SpatialDataSet.txt')
```

Taking a look at the dataset

```
df.head()
```

	longitude	latitude
0	46.659107	24.768269
1	46.702409	24.680454
2	46.650310	24.767690
3	46.748908	24.715848
4	46.758220	24.712114

My dataset is a simple one (1444 rows × 2 columns). I have collected random geographical data points in Riyadh City for this demonstration.

Define the Bounding Box

Now, we have to define the Bounding Box. Bounding Box is the area defined by two longitudes and two latitudes that will include all spatial points.

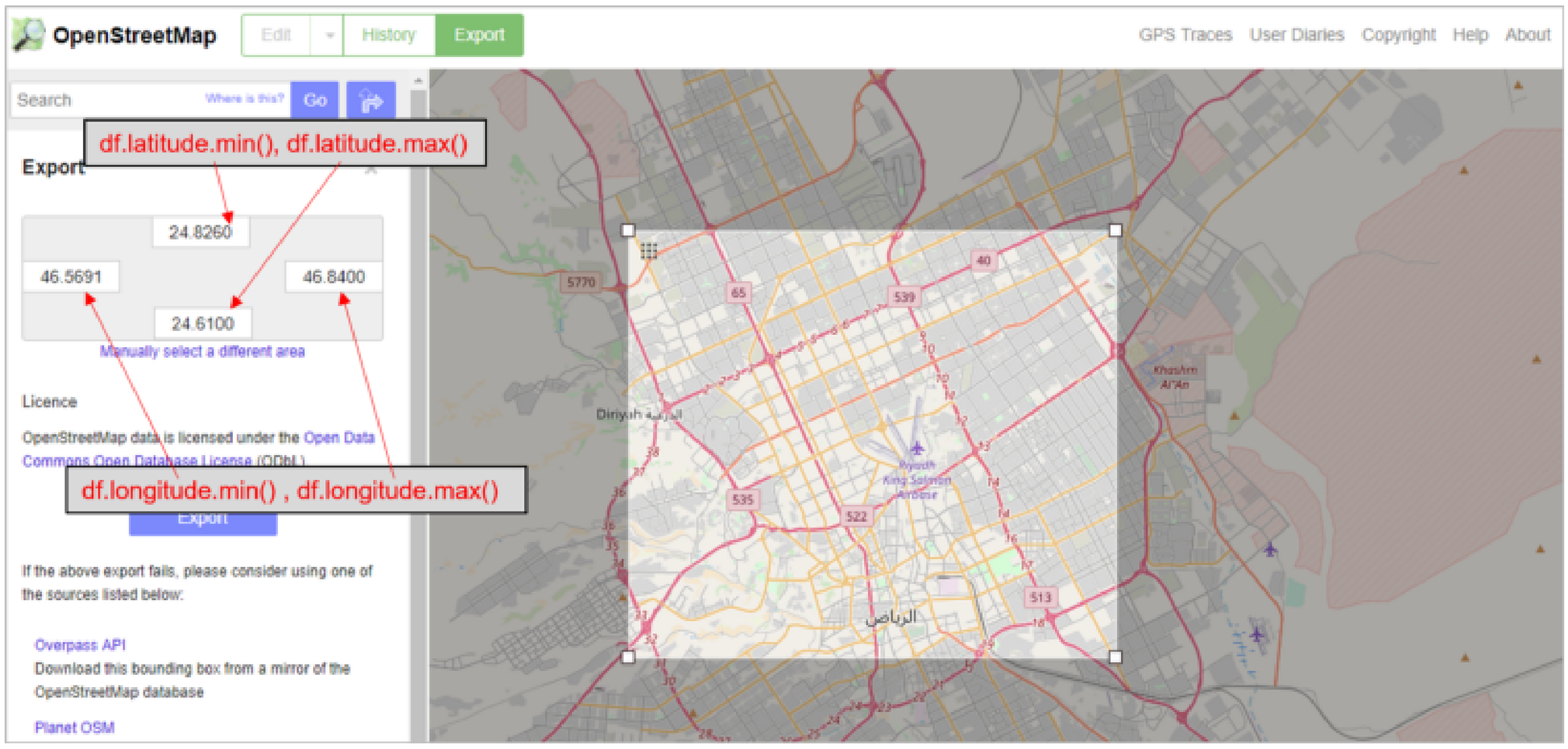
```
BBox = ((df.longitude.min(),   df.longitude.max(),
         df.latitude.min(), df.latitude.max())

> (46.5691,46.8398, 24.6128, 24.8256)
```

Get Your Map

Go to opestreetmap.org website and export the desired map as an image by first entering the bounding box data. I did the same as explained in the

below image (more details are mentioned here as well: steps to export a map image).



Back to the coding environment and load the map image:

```
ruh_m = plt.imread('C:/.. … /Riyadh_map.png')
```

Final Step: Plotting

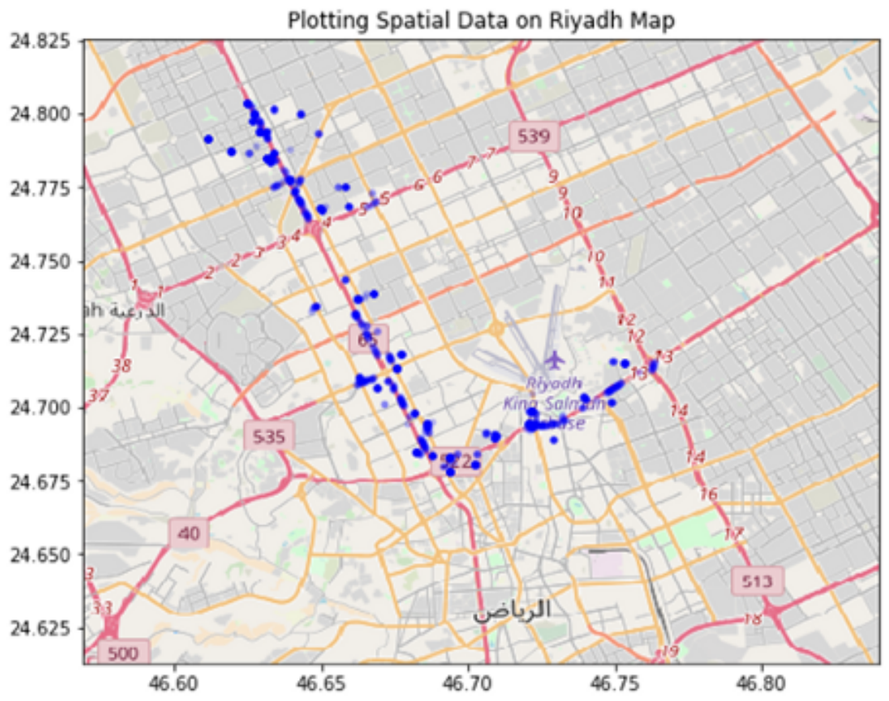
Finally, plot the '*df.longitude*' and '*df.latitude*' coordinates as scatter points on the '*ruh_m*' map image. Note that it is important to set up the X-axis and Y-axis as per the bounding box '*BBox*'

```
fig, ax = plt.subplots(figsize = (8,7))

ax.scatter(df.longitude, df.latitude, zorder=1, alpha= 0.2, c='b',
s=10)

ax.set_title('Plotting Spatial Data on Riyadh Map')
ax.set_xlim(BBox[0],BBox[1])
ax.set_ylim(BBox[2],BBox[3])

ax.imshow(ruh_m, zorder=0, extent = BBox, aspect= 'equal')
```



Data Science Spatial Analysis AI Visualization Python