

Week 4 Assignment

February 1, 2021

```
[23]: # Upload OSM Data to OSM
import osmnx as ox

# Manipulating Evidence
import pandas as pd

# Spatial details to manipulate and simulate
import geopandas as gpd

#basemaps
import contextily as ctx

# Plotting for
import matplotlib.pyplot as plt

# location
address = 'Dallas, TX'

# Tags on what sort of details we are receiving
tags = {'building':True}

# Width from the point of place in meters
dist = 2000
```

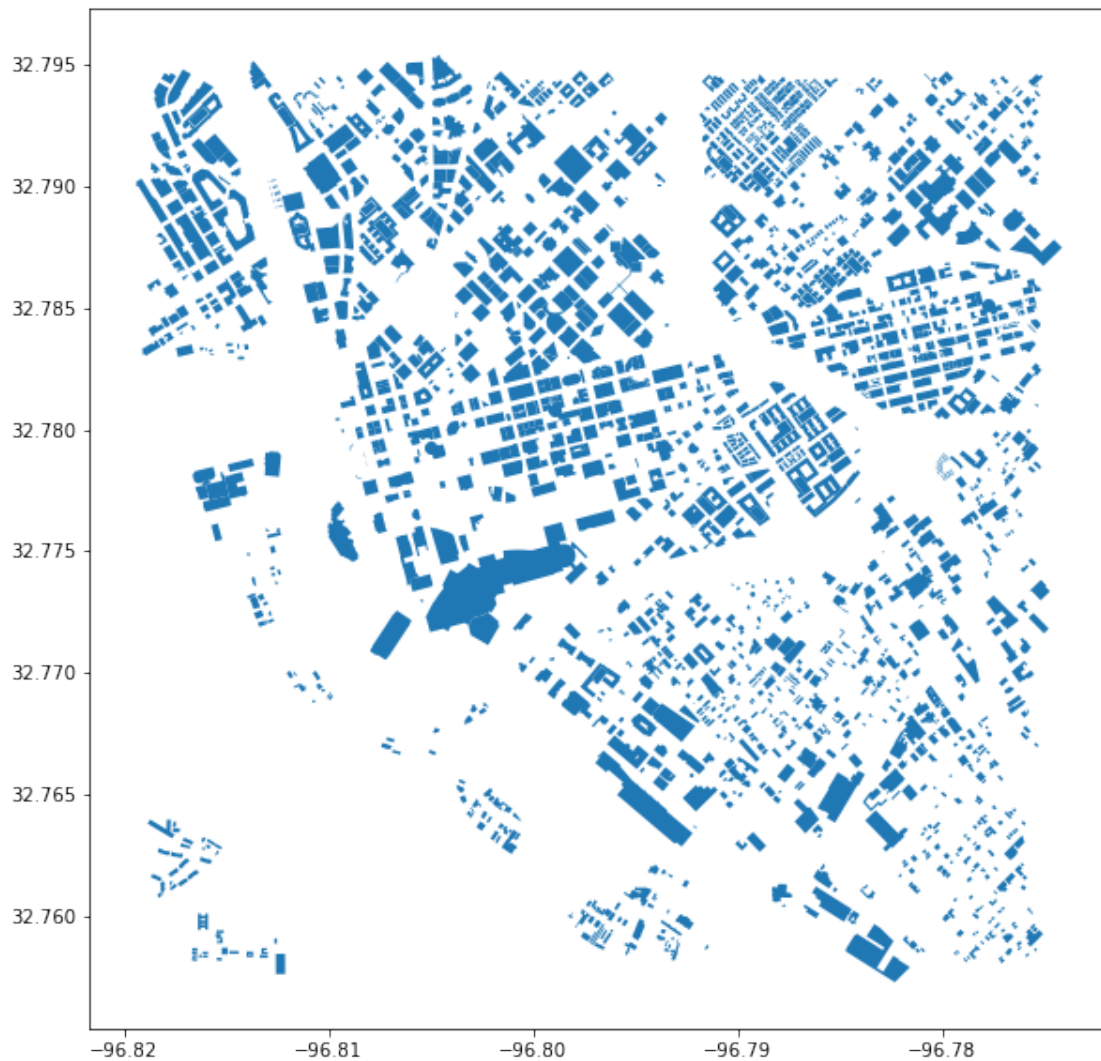
```
[24]: %%time
# %%Time is a mystical order to see how long this cell needs to function.

# obtain the data from OSM
osm = ox.geometries_from_address(address,
                                tags=tags,
                                dist=dist)
```

CPU times: user 666 ms, sys: 23.4 ms, total: 690 ms
Wall time: 10 s

```
[25]: # fast plot
osm.plot(figsize=(10,10))
```

[25]: <matplotlib.axes._subplots.AxesSubplot at 0x7f2b10cfe760>



```
[26]: # How many columns and rows?
osm.shape
# How many columns and rows?
osm.shape
#10 rows
osm.sample(10)
```

```
[26]:
```

	unique_id	osmid	element_type	addr:city	addr:housenumber	\
1819	way/619201201	619201201	way	NaN	2000	
288	way/361441664	361441664	way	NaN	NaN	
1330	way/465306641	465306641	way	NaN	1612	
1909	way/628116110	628116110	way	NaN	NaN	

212	way/221299574	221299574	way	NaN	NaN
1003	way/465293700	465293700	way	NaN	904
548	way/463988444	463988444	way	NaN	624
1290	way/465306601	465306601	way	NaN	2830
1574	way/471928301	471928301	way	NaN	314
1063	way/465306339	465306339	way	NaN	NaN

	addr:postcode	addr:state		addr:street	amenity	\
1819	NaN	NaN		McKinney Avenue	NaN	
288	NaN	NaN			NaN	NaN
1330	NaN	NaN	Martin Luther King Jr Boulevard		NaN	NaN
1909	NaN	NaN			NaN	NaN
212	NaN	NaN			NaN	NaN
1003	NaN	NaN		North Ewing Avenue	NaN	
548	NaN	NaN	North Good Latimer Expressway		NaN	NaN
1290	NaN	NaN		South Ervay Street	NaN	
1574	NaN	NaN	South Riverfront Boulevard		NaN	NaN
1063	NaN	NaN			NaN	NaN

	building	...	level	landuse	residential	brewery	microbrewery	ways	\
1819	commercial	...	NaN	NaN	NaN	NaN	NaN	NaN	
288	commercial	...	NaN	NaN	NaN	NaN	NaN	NaN	
1330	retail	...	NaN	NaN	NaN	NaN	NaN	NaN	
1909	detached	...	NaN	NaN	NaN	NaN	NaN	NaN	
212	commercial	...	NaN	NaN	NaN	NaN	NaN	NaN	
1003	commercial	...	NaN	NaN	NaN	NaN	NaN	NaN	
548	commercial	...	NaN	NaN	NaN	NaN	NaN	NaN	
1290	yes	...	NaN	NaN	NaN	NaN	NaN	NaN	
1574	commercial	...	NaN	NaN	NaN	NaN	NaN	NaN	
1063	yes	...	NaN	NaN	NaN	NaN	NaN	NaN	

	building_1	type	designation	apartment
1819	NaN	NaN	NaN	NaN
288	NaN	NaN	NaN	NaN
1330	NaN	NaN	NaN	NaN
1909	NaN	NaN	NaN	NaN
212	NaN	NaN	NaN	NaN
1003	NaN	NaN	NaN	NaN
548	NaN	NaN	NaN	NaN
1290	NaN	NaN	NaN	NaN
1574	NaN	NaN	NaN	NaN
1063	NaN	NaN	NaN	NaN

[10 rows x 108 columns]

[27]: `osm.info()`

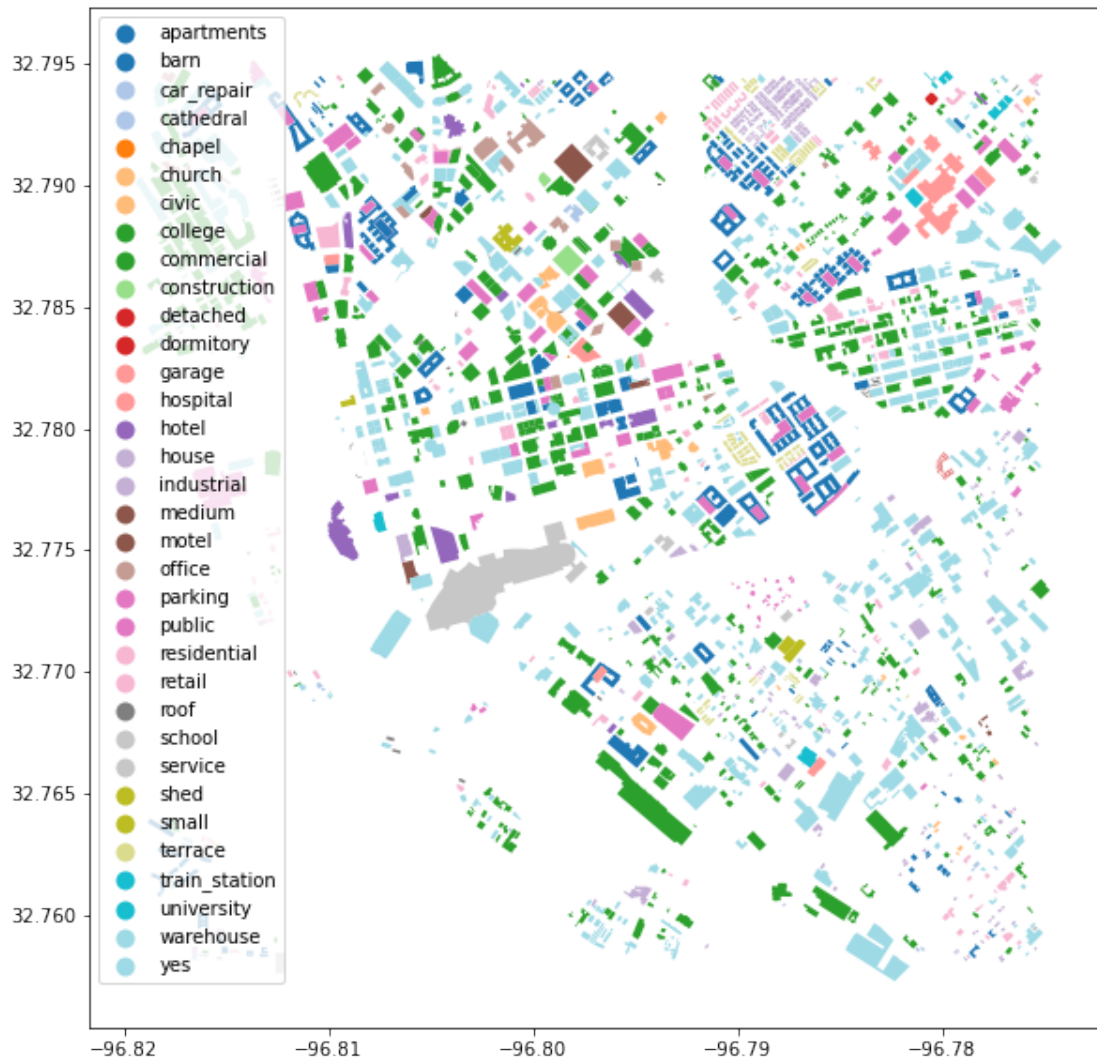
```
<class 'geopandas.geodataframe.GeoDataFrame'>  
Int64Index: 2100 entries, 0 to 2099  
Columns: 108 entries, unique_id to apartment  
dtypes: geometry(1), int64(1), object(106)  
memory usage: 1.7+ MB
```

```
[28]: osm['height'].describe()
```

```
[28]: count          70  
      unique        53  
      top         15.24  
      freq          4  
      Name: height, dtype: object
```

```
[29]: osm.plot(figsize=(10,10),  
              column='building',  
              cmap='tab20',  
              legend=True)
```

```
[29]: <matplotlib.axes._subplots.AxesSubplot at 0x7f2b14fe1220>
```



```
[30]: # Summarize data by year and use reset index to place the results in a dataframe
osm_date = osm['start_date'].value_counts().reset_index()
osm_date
# name columns again
osm_date = osm_date.rename(columns={'index': 'year', 'start_date': 'count'})
osm_date
# ordered by year
osm_date = osm_date.sort_values(by='year')
# plot
osm_date.plot(figsize=(10,5), x='year', title='Count of buildings and when they
↳ were built in ' + address)
```

```
[30]: <matplotlib.axes._subplots.AxesSubplot at 0x7f2b15184f40>
```

