Wrangle OpenStreetMap Data

January 1, 2020

1 OpenStreetMap Data Case Study

1.1 Map Area

1.1.1 Marion County, FL

- https://www.openstreetmap.org/relation/1210723

This map is the of my home county.I've been away from here for most of the last decade so it is interseting to see what I can find out from the data.

[1]:
```
import sqlite3
conn = sqlite3.connect("dW_database.db")
cursor = conn.cursor()
```

[2]:
```
cursor.execute("SELECT COUNT(DISTINCT(e.uid)) FROM (SELECT uid FROM nodes UNION␣
↪ALL SELECT uid FROM ways) e;")

print cursor.fetchall()
```

```
[(525,)]
```

1.2 Problems Encountered in the Map

- I had a minor issue when cleaning using St to Street. Had errors with Street becoming Streetreet. Fixed it by including a space after St.
- Looking though the date I noticed that the postal codes could be cleaned up a bit. Some are listed twice with some values having the 4 digit extension. Also if I were to look deeper I would look at the few postal codes with only one result.
- For the cities values Ocala is listed twice. Also some of the cities are from outside of Marion County with other not even names just digits.

1.3 List of Postal Codes

[3]:
```
cursor.execute("""SELECT tags.value, COUNT(*) as count
FROM (SELECT * FROM nodes_tags
    UNION ALL
    SELECT * FROM ways_tags) tags
WHERE tags.key='postcode'
GROUP BY tags.value
ORDER BY count DESC;""")

print cursor.fetchall()
```

```
[(u'32702', 28), (u'34471', 18), (u'34474', 15), (u'32696', 14), (u'32667', 13),
(u'34472', 10), (u'34470', 9), (u'34420', 9), (u'34482-1479', 6), (u'34432', 5),
(u'34491', 4), (u'34482', 4), (u'34480', 4), (u'34475', 4), (u'32664', 4),
(u'34488', 3), (u'34476', 3), (u'32668', 3), (u'34481', 2), (u'34479', 2),
(u'32696-6683', 2), (u'32617', 2), (u'32159', 2), (u'34488-1719', 1), (u'34473',
1), (u'34470-5001', 1), (u'34434', 1), (u'34433', 1), (u'34431', 1), (u'34330',
1), (u'33480', 1), (u'32784', 1), (u'32667-7435', 1), (u'32640-8903', 1),
(u'32640-7862', 1), (u'32164', 1), (u'32134', 1), (u'32113', 1)]
```

1.4 List of Cities

[4]:
```
cursor.execute("""SELECT tags.value, COUNT(*) as count
FROM (SELECT * FROM nodes_tags UNION ALL
    SELECT * FROM ways_tags) tags
WHERE tags.key LIKE '%city'
GROUP BY tags.value
ORDER BY count DESC;""")

print cursor.fetchall()
```

```
[(u'Ocala', 79), (u'Altoona', 28), (u'Williston', 16), (u'Micanopy', 14),
(u'Dunnellon', 10), (u'Belleview', 9), (u'McIntosh', 5), (u'Summerfield', 4),
(u'Silver Springs', 4), (u'Morriston', 3), (u'Citra', 3), (u'8', 3), (u'The
Villages', 2), (u'Hawthorne', 2), (u'Anthony', 2), (u'ocala', 1), (u'Umatilla',
1), (u'Lady Lake', 1), (u'Fort McCoy', 1), (u'Citrus Springs', 1), (u'5', 1),
(u'49', 1), (u'3', 1), (u'21', 1), (u'20', 1), (u'2', 1), (u'1345', 1), (u'10',
1)]
```

1.5 Data Overview

1.5.1 File Sizes

- marionCounty.xml : 110mb
- audity.py : 2.17kb
- data.py: 7.36kb
- dW_database.db : 65mb
- nodes.csv : 41.3mb
- nodes_tags.csv : 1.17mb
- parsing.py : 1.31kb
- ways.csv : 2.48mb
- ways_nodes.csv : 13.2
- ways_tags.csv : 8.52

Here we are loading the database

1.6 Top 10 contributing users

---

[5]:
```
cursor.execute("""SELECT e.user, COUNT(*) as num
FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways) e
GROUP BY e.user
ORDER BY num DESC
LIMIT 10;""")

print cursor.fetchall()
```

```
[(u'NE2', 128333), (u'LogicalViolinist', 57759), (u'grouper', 50330),
(u'valerietheblonde', 41749), (u'woodpeck_fixbot', 33189), (u'DenisCarriere',
17905), (u'Horza', 13794), (u'pete404', 8823), (u'maxolasersquad', 8784), (u
'bot-mode', 8706)]
```

Two intersting point I see from the list of the top 10 contributing users are the two bots, woodpeck_fixbot and bot-mode. woodpeck_fixbot was created by user Frederik Ramm and the bot is used to correct typos and remove null features. bot-mode is a bot made by Emacsen, one of the founding members of OpenStreeMap. From what I could gather about the bot their goal seems similar to what this project does. Correcting minor issues with the data.

- woodpeck_fixbot
- Frederik Ramm ____
- bot-mode
- Emacsen

1.7 The 10 Ten Types of Locations

```
[6]: cursor.execute("""SELECT e.value, COUNT(*) as num
FROM (SELECT value FROM nodes_tags WHERE key = "amenity") e
GROUP BY e.value
ORDER BY num DESC
LIMIT 10;""")

print cursor.fetchall()
```

```
[(u'place_of_worship', 523), (u'school', 82), (u'grave_yard', 69), (u'bench',
44), (u'fire_station', 37), (u'restaurant', 32), (u'fast_food', 23), (u'police',
22), (u'post_office', 21), (u'parking', 21)]
```

Not shocking place of worship is the highest type of location. Some data point I found intesting were bench and grave yard. It's easy to forget how many grave yards are in my county. I'm not suprised but it's erie seeing the number. Benches are high and I wonder if that is the total number. 44 seems a bit low and it's most likely just a number that a few users have put in so far.

1.8 Number of Nodes

```
[7]: cursor.execute("SELECT COUNT(*) FROM nodes")

print cursor.fetchall()
```

```
[(495993,)]
```

1.9 Number of Ways

```
[8]: cursor.execute("SELECT COUNT(*) FROM ways")

print cursor.fetchall()
```

```
[(41077,)]
```

1.10 Number of Places of Worship by Religion

```
[9]: cursor.execute("""SELECT tags.value, COUNT(*) as num
FROM (SELECT * FROM nodes_tags
    UNION ALL
    SELECT * FROM ways_tags) tags
WHERE tags.key='religion'
GROUP BY tags.value
ORDER BY num DESC;""")

print cursor.fetchall()
```

```
[(u'christian', 516), (u'jewish', 3), (u'unitarian_universalist', 1), (u'hindu',
1)]
```

Not surprising christian is the top religion. Though it is nice to see other religions that I did not know about in the area.

1.11 Number of Places by Type

```
[10]: cursor.execute("""SELECT tags.value, COUNT(*) as num
FROM (SELECT * FROM nodes_tags
    UNION ALL
    SELECT * FROM ways_tags) tags
WHERE tags.key='place'
GROUP BY tags.value
ORDER BY num DESC;""")

print cursor.fetchall()
```

```
[(u'hamlet', 113), (u'island', 25), (u'neighbourhood', 10), (u'islet', 9),
(u'locality', 7), (u'village', 5), (u'city', 2), (u'town', 1), (u'county', 1)]
```

As I would expect there is only 1 county. The number is hamlet is large but there are a large number some rural and remote communies with in county. Below is a link that describes what each class of place means.

- Key:place