P3. OpenStreetMap Project -- Data Wrangling with MongoDB

Map Area

Brooklyn, New York, United States

- Boundary of Kings County: https://www.openstreetmap.org/relation/369518#map=11/40.6444/-73.9449
- Metro extracts: https://mapzen.com/data/metro-extracts/metro/brooklyn_new-york/
- A smaller sample (sample.osm) of the map (brooklyn.osm) was generated.

Exploring and Auditing the Sample DataSet

1. Explore the tags in this file. Different tags and counts:

```
{'member': 307,
 'nd': 69534,
 'node':49706,
 'osm': 1,
 'relation': 35,
 'tag': 56355,
 'way': 9810}
```

2. Explore the 'k' value of the all the tags in 'way' element and count each of them.

    a. Some interesting tags with value- 'k' I would like to explore further:

    ```
    {'addr:postcode': 6604, 'addr:street': 6609}
    {'FIXME': 1, ...}
    ```

    b. I would like to explore tags with 'k' start with "tiger" as well.

3. Auditing street names and zip code.

    a. Surprisingly, almost all of the street names are clean and well formatted. There is only one over-abbreviated street name: { 'St.' ==> '9th St. '}

    b. The format of the zip code is consistent in this dataset. It is checked by the function "is_valid_zip" (valid zip code is consisted of 5 digits, eg. '11204', or 9 digits, eg. '11204-3965'). No invalid zip code is found in this sample dataset. However, a lot of them are not located in Brooklyn. Some are in Jersey City,

Lower Manhattan and Queens. According to NYC zip code, all the zip codes in Brooklyn should start with "112**".

4. Auditing tags in 'way' element which are pulled from Tiger GPS data.

    a. The 'tiger:name_type' values are all abbreviation. Some of them include multiple values, eg 'Ave; St; Ave'.

```
'tiger:name_type': {'Ave',
                    'Ave:Pky',
                    'Ave; St; Ave',
                    'Ave;St;Ave',
                    'Blvd',....},
```

    b. A lot of the 'tiger:zip_left' equal to 'tiger:zip_right', and 'tiger:zip_left_1' equals to 'tiger:zip_left'. It looks redundant to me. For example:

```
<tag k="name" v="Harrison Street"/>
<tag k="highway" v="residential"/>
<tag k="tiger:cfcc" v="A41"/>
<tag k="tiger:county" v="New York, NY"/>
<tag k="tiger:zip_left" v="10013"/>
<tag k="tiger:name_base" v="Harrison"/>
<tag k="tiger:name_type" v="St"/>
<tag k="tiger:zip_right" v="10013"/>
```

    Another example:

```
<tag k="name" v="112th Street"/>
<tag k="oneway" v="yes"/>
<tag k="highway" v="residential"/>
<tag k="tiger:cfcc" v="A41"/>
<tag k="tiger:county" v="Queens, NY"/>
<tag k="tiger:reviewed" v="no"/>
<tag k="tiger:zip_left" v="11419"/>
<tag k="tiger:name_base" v="112th"/>
<tag k="tiger:name_type" v="St"/>
<tag k="tiger:zip_right" v="11419"/>
<tag k="tiger:zip_left_1" v="11419"/>
<tag k="tiger:zip_right_1" v="11419"/>
```

    c. Inconsistent format of 'Tiger: reviewed'. According to the documentation [TIGER fixup] (http://wiki.openstreetmap.org/wiki/Talk:TIGER_fixup), If a tiger tag has been reviewed, the "tiger:reviewed" is supposed to be removed. So the only value for this tag should be "no". However, the actually values which have been printed out include: 'yes', 'no; yes; no', 'yes; no' and '; no; no'. It is quite confusing.

```
'tiger:reviewed': {'no', 'no; yes; no', 'yes; no', 'yes', '; no; no'}
```

Problems Encountered in this Sample

1. Over-abbreviated street names and ‘tiger: name_type’. They were all updated when converting from XML into JSON format for importing into MongoDB. PYTHON code

2. Wrong regions (regions that do not belong to Brooklyn) are revealed by the zip code. The correct zip code range for Brooklyn is between 11201 - 11256. It can be further explored and updated in MongoDB. I didn't notice any wrong format of zip code in the sample dataset, when processing the actual dataset, there is a function called "update_postcode" in the [process_data.py]. If the zip code is not in valid format, it will be marked as "Need_to_be_fixed".

3. Inconsistent format of DATA pulled from tiger GPS.

Data Overview

This section contains basic statistics about the dataset and the MongoDB queries used to gather them. Code: importing data into MongoDB

File sizes

```
- brooklyn.osm ......... 666.2 MB
- brooklyn.osm.json .... 725.4 MB
```

1. Number of documents

```
> db.brooklyn.find().count()
> 2975785
```

2. Number of nodes and ways

```
> db.brooklyn.find({"type":"node"}).count()
> 2485112
> db.brooklyn.find({"type":"way"}).count()
> 490509
```

3. Top 10 contributing users

```
> db.brooklyn.aggregate([{"$group":{"_id": "$created.user", "count":{"$sum":1}}},
{"$sort": {"count":-1}}, {"$limit":10}])
```

The top contributor: “Rub21_nycbuildings” contributed to this map 5 times more than the second contributor.

4. Number of addresses with non-Brooklyn zip code.

"_id" : "Rub21_nycbuildings"	"count" : 1740323
"_id" : "ingalls_nycbuildings",	"count" : 373554
"_id" : "ediyes_nycbuildings",	"count" : 189694
"_id" : "celosia_nycbuildings",	"count" : 117361
"_id" : "ingalls",	"count" : 105358
"_id" : "lxbarth_nycbuildings",	"count" : 79851
"_id" : "aaron_nycbuildings",	"count" : 42023
"_id" : "ewedistrict_nycbuildings",	"count" : 35002
"_id" : "smlevine",	"count" : 25054
"_id" : "robgeb",	"count" : 23676

```
>db.brooklyn.find({"address.postcode":{"$gte":"11201", "$lte":
"11256"}}).count()
> 290566
> db.brooklyn.find({"address.postcode":{"$exists": 1}}).count()
> 387592
> db.brooklyn.find({"address.postcode":{"$gte":"11201", "$lte": "11256"}},
{"tiger.zip_left":{"$gte":"11201", "$lte":
"11256"}},{"tiger.zip_right":{"$gte":"11201", "$lte": "11256"}}).count()
> 290566
```

So the total number of address (including those pulled from tiger GPS) that have valid Brooklyn zip code is 290566.

5. Update the zip code (delete all the documents that have non-brooklyn zip code)

```
> db.brooklyn.remove({"address.postcode":{"$gt":"11256"}})
> WriteResult({ "nRemoved" : 80613 })
> db.brooklyn.remove({"address.postcode":{"$lt":"11201"}})
> WriteResult({ "nRemoved" : 16413 })
```

6. Explore how many of the tiger tag still need to be reviewed after updating zip code.

```
> db.brooklyn.find({"tiger.reviewed":{"$exists": 1}}).count()
> 10106
> db.brooklyn.find({"tiger.reviewed":"yes"}).count()
> 251
```

If a tiger data has been reviewed, the review tag is supposed to be deleted. There are 10106 tags that still have review value, but 251 of them have review value “yes”. So the total tiger tags that still need to be reviewed are 9855.

7. Explore the top three amenities in Brooklyn.

```
>db.brooklyn.aggregate([{"$match":{'amenity':{"$exists":1}}},{"$group":{'_id':
'$amenity',"count":{"$sum":1}}},{"$sort":{"count":-1}},{"$limit":3}])
> { "_id" : "bicycle_parking", "count" : 2818 }
> { "_id" : "place_of_worship", "count" : 834 }
> { "_id" : "restaurant", "count" : 814 }

> db.brooklyn.aggregate([{"$match":{"amenity":{"$exists":1},
"amenity":"place_of_worship"}}, {"$group":{"_id":"$religion",
"count":{"$sum":1}}}, {"$sort":{"count":-1}}, {"$limit":1}])
> { "_id" : "christian", "count" : 655 }

> db.brooklyn.aggregate([{"$match":{"amenity":{"$exists":1},
"amenity":"restaurant"}}, {"$group":{"_id":"$cuisine", "count":{"$sum":1}}},
{"$sort":{"count": -1}}, {"$limit":3}])
> { "_id" : null, "count" : 307 }
> { "_id" : "pizza", "count" : 47 }
> { "_id" : "italian", "count" : 43 }
```

Conclusion: First, the large number of bicycle parking stands out. It is due to the bike share system (CityRacks) lunched by Mayor Bloomberg and Department of Transportation Commissioner Janette Sadik-Khan. Second, the dominant religion in Brooklyn is unsurprising: Christian. Finally, the documentation of restaurants in this dataset needs to be improved. Various of cuisines are not well categorized yet.

- Here are some examples of uncategorized restaurants:

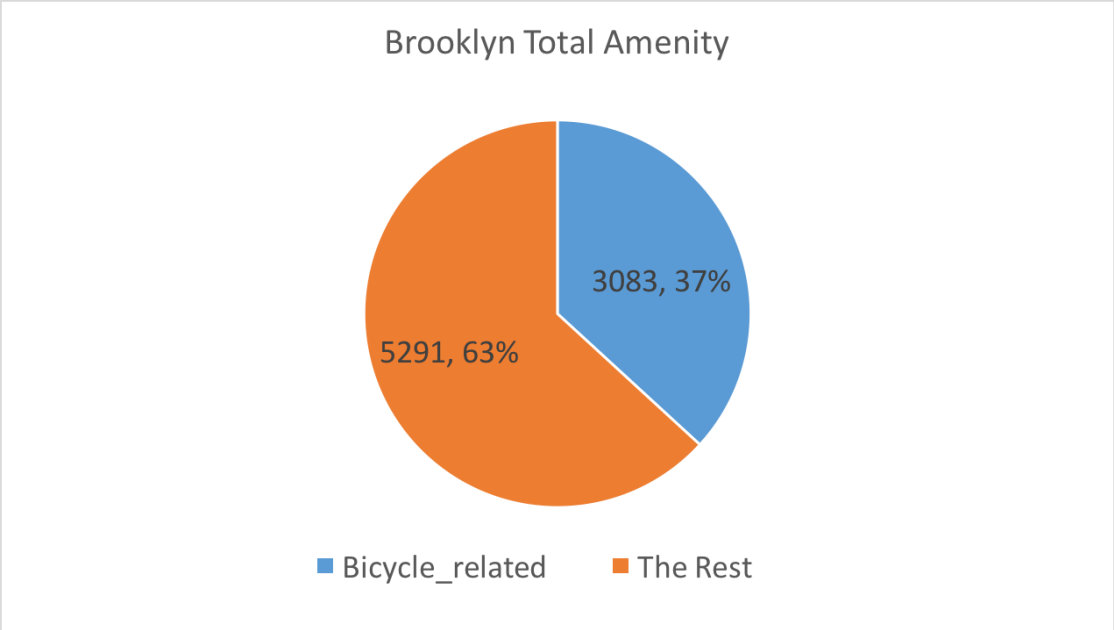
```
<node changeset="6496165" id="1013624842" lat="40.716904" lon="-73.999527"
timestamp="2010-11-30T17:39:27Z" uid="29040" user="chrismcnally" version="1">
<tag k="name" v="China Village Restaurant"/>
<tag k="amenity" v="restaurant"/>
<tag k="addr:street" v="Baxter Street"/>
<tag k="addr:housenumber" v="94"/>

<node changeset="11446248" id="1153146341" lat="40.7029557" lon="-74.0132642"
timestamp="2012-04-29T03:27:09Z" uid="290680" user="wheelmap_visitor"
version="4">
<tag k="name" v="Au Bon Pain"/>
<tag k="amenity" v="restaurant"/>
<tag k="wheelchair" v="limited"/>
```

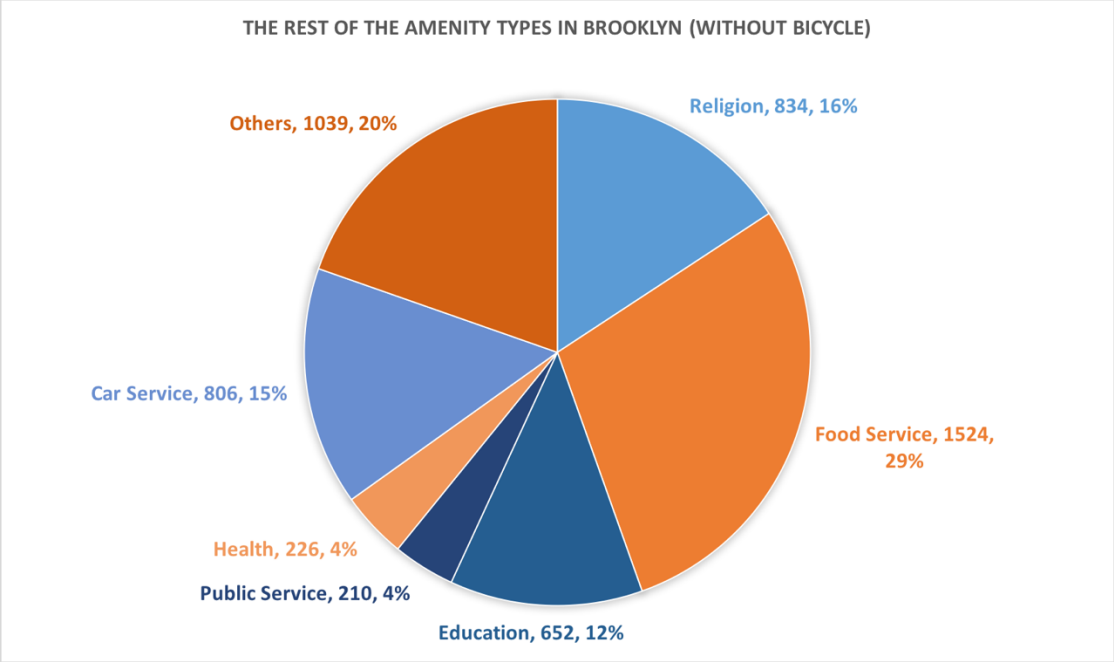
8. Visualization of the distribution of all amenity in Brooklyn.

```
> db.brooklyn.find({'amenity':{"$exists":1}}).count()
> 8374
>
db.brooklyn.aggregate([{"$match":{'amenity':{$exists:1}}},{"$group":{'_id':'$a
menity',"count":{"$sum":1}}},{"$sort":{"count":-1}},{"$limit":30}])
```

Pie Chart of the bike related amenity vs. all the rest amenity in Brooklyn:



Pie Chart of all the other amenity in Brooklyn except bike related:



Amenity_Type	Count_Type	Amenity_Name	Count#
Bicycle	3083	bicycle_parking	2818
		bicycle_rental	265
Religion	834	place of worship	834

Food Service	1524	restaurant	814
		café	287
		bar	179
		fast_food	177
		pub	67
Education	652	school	556
		library	59
		university	23
		college	14
Public Service	210	Fire_station	133
		police	30
		postoffice	47
Health	226	hospital	112
		pharmacy	100
		doctors	14
Car Service	806	parking	722
		fuel	69
		car_sharing	15
Others	1039	Others	1039
Total	8374	Total	8374

Those results indicate that Brooklyn is populated and good living place. It has great amount of food services around and plenty of community churches. However, compared with the living condition, the education part is a little weak. Maybe the government should consider more public libraries or schools in Brooklyn. However, more data about populations age, education level and economic situations is need to draw any conclusion on this. What's more, the health system seems pretty weak in Brooklyn. Data of family doctors and small clinics might be missing from this dataset. I believe more data about health related amenity need to be updated. However, it also suggested the health system need to be improved largely in Brooklyn.

Other Thoughts

1. First, the quality of OpenStreetMap data for Brooklyn is pretty good. Very little error in the format of addresses and zip code. However, all the data pulled from tiger GPS need to be fixed. It has some redundant fields, for example, some of the location has zip_left_1, zip_left_2, zip_right_1, zip_right_2 and more, but all those zip codes are exactly the same. I think it is reasonable to reduce and combine them to one if they all have the same value.

- Proposed improvement: simplify the "postcode" in tiger tag. If the zip_left, zip_left_1 and zip_left_2 (if it exists) are all the same value, they can be combined into one field - zip_left.

If they are different, they should be combined into one with all the unique value. The same procedure could be performed on zip_right too; Then if the value of zip_left equals to zip_right, combine them into one - zip_code.

- Benefits: It would save space to store those data by reducing those redundant fields. Also, it seems easier for people to review those tiger data in the future.

- Anticipated problems: some people do love the zip_left and zip_right function of tiger GPS, because sometimes the zip code can be different on the same street.

2. Second, as more people moving to Brooklyn recent years, the variety and number of restaurants increased enormously. Thus it definitely needs to be updated to provide more information for the economics of Brooklyn.

3. Finally, comparison of the development of biking system in Brooklyn with the traffic situation and community health conditions can be a very interesting project to dive in. It might help us gain valuable insight into the impact of biking system on local economy.

- Proposed improvement: make a app for biking in NYC or brooklyn. This app would include all the bicycle_parking spot which will help the bikers to find parking easily. As suggested by the reviewer, it would awesome to include all the biking lane on this app. Another important feature would be including the traffic situation in real time (I guess those information could be pulled out from google map and merge into the biking app). Basically it gives people options, if the traffic is really bad on one day, biking might be a better option. Another idea is to input the start location and destination, input the bike type and the biker's body type, the app could calculate how much calories are burned by the time used for the biking trip.

- Benefits: it would be an GPS + parking + fitness app for bikers. All in one.

- Anticipated problems: google map has biking routes already, and a lot of other fitness app has biking calories calculator. The major problem is to merge all those data into one app, and also add more features like bicycle_parking.