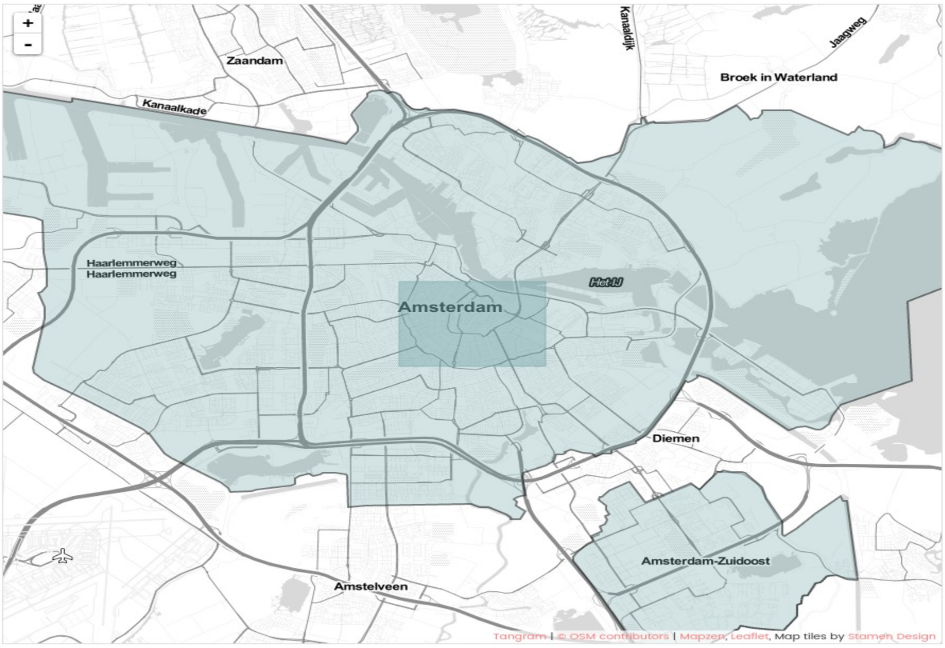
Yasser Alnakhli

Background:

- OpenStreetMap of Amsterdam City, The Netherlands, has been used.

- The data has been downloaded as XML format from (https://mapzen.com). Thu, Jun 15, 2017.



- Only the old center of the city has been selected. It is the most visited area of Amsterdam." It is known for its traditional architecture, canals, shopping, and many coffeeshops. Dam Square is considered its ultimate centre, but just as interesting are the areas around Nieuwmarkt and Spui. The Red Light District is also a part of the Old Centre."
For more information please visit: (http://wikitravel.org/en/Amsterdam/Old_Centre).

- I spent some time between 2012 and 2013 in Amsterdam. I liked the culture there. There are very big differences between Arabic culture and the Dutch...I liked the Old Center the most in Amsterdam. It is like an openair big museum. Many things to see and discover. That is why the Old center was my decision in this project.

- The data wrangling process of Amsterdam XML OSM has been done by discovering auditing and cleaning the dataset before further analysis using MongoDB.

Problems Faced:

- Becouse of my computer's hardware and the bad Internet I have, I decided to use just little above the suggested file size (>50MB). I had to use another Internet and computer to download the proper xml file size. It took some time.

- Choosing just a small part of the city did not reflect the overall finding of the analysis. So my current work is just for the old center of Amsterdam and might not be the same if I choose a larger area.
- The Decision to use MySQL or MongoDB also took some time, I spent some time to look at both as I did not have much knowledge about both.
- The OSM XML file of Amsterdam is quite nice and clean comparing to the other cities in the world. I saw Chicago's data at Udacity videos and I checked the files of Riyadh, KSA where I am from, and they are little bit messy. Although Dutch Language is similar to English, but mostly English been used in XML files OSM Amsterdam. This is actually was challenge to decide what do I need to clean in the file and I came up with another idea :)
- One of the huge cultural differences I faced when I was in Amsterdam, the homosexuality. Amsterdam is a homosexual friendly and there are many bars or other stuff which I want to locate
- Almost in all the world, a coffeeshop is a shop to have a nice coffee. Sometimes it been called Cafe. However, in Amsterdam, a coffeeshop is a drug shop or a shop to legally smoke some cannabis. This misunderstanding will be dressed in future steps.

Data Wrangling Process:

The data wrangling process has 3 main steps:

- Investigating the dataset:

1. **- With Python:**

The file size = 61 MB *Obtained by import os*

The sample file size = 1.3 MB. It has been used for testing. Obtained by *creatSample.py*

counts of primary tags: {'node': 218600, 'nd': 228655, 'bounds': 1, 'member': 7678, 'tag': 515651, 'osm': 1, 'way': 24823, 'relation': 654}) Obtained by *count_all.py*

counts of secondary tags: 771 Obtained by *count_secondary_tag.py*

There are 38 gay friendly place at the old center of Amsterdam. Obtained by *countGay.py*

And from *FindGay.py* I got the following output. The full result is at the jupyter notebook.

Some of the gay friendly places:

```
IHLIA Homodok
lat: 52.3762463, lon: 4.9081983
```

De Barderij
lat: 52.3758329, lon: 4.9004544
Mankind
lat: 52.3607475, lon: 4.888122
Café Rouge
lat: 52.3669068, lon: 4.8961508
Pink Point
lat: 52.3741824, lon: 4.8844544

2. - With MongoDB

```
('The number of data', 2341746)
('The number of ways', 238796)
('The number of nodes', 2102932)
('The number of bicycle parkings', 1006)
('The number of tourism attractions', 597)
```

```
The top 5 cuisine:
[{u'_id': u'italian', u'count': 549},
 {u'_id': u'burger', u'count': 296},
 {u'_id': u'thai', u'count': 229},
 {u'_id': u'regional', u'count': 207},
 {u'_id': u'indian', u'count': 202}]
```

```
top 5 common amrity:
[{u'_id': u'restaurant', u'count': 5144},
 {u'_id': u'cafe', u'count': 2368},
 {u'_id': u'pub', u'count': 1916},
 {u'_id': u'fast_food', u'count': 1759},
 {u'_id': u'bench', u'count': 1026}]
```

```
The top 10 common shops:
[{u'_id': u'clothes', u'count': 2527},
 {u'_id': u'gift', u'count': 1074},
 {u'_id': u'shoes', u'count': 630},
 {u'_id': u'jewelry', u'count': 458},
 {u'_id': u'supermarket', u'count': 431},
 {u'_id': u'convenience', u'count': 373},
 {u'_id': u'bakery', u'count': 324},
 {u'_id': u'books', u'count': 306},
 {u'_id': u'hairdresser', u'count': 279},
 {u'_id': u'alcohol', u'count': 265}]
```

```
The Universities in Amsterdam:
[{u'_id': u'Oudemanhuispoort (UvA)', u'count': 40},
 {u'_id': u'Amstelcampus', u'count': 9},
 {u'_id': u'UvA Binnengasthuisterrein', u'count': 9},
 {u'_id': u'UvA Bungehuis', u'count': 9},
 {u'_id': u'UvA PC Hoofthuis', u'count': 9},
 {u'_id': u'Service & Informatiecentrum UvA', u'count': 9},
 {u'_id': u'UvA BG5', u'count': 9},
 {u'_id': u'P', u'count': 9},
 {u'_id': u'Universiteit van Amsterdam', u'count': 9},
 {u'_id': u'Aula UvA', u'count': 9},
 {u'_id': u'Ruimtelijke Wetenschappen (FMG)', u'count': 9},
 {u'_id': u'UvA', u'count': 9},
 {u'_id': u'UvA Faculteit der Maatschappij-en Gedragswetenschappen',
  u'count': 9},
 {u'_id': u'Spui25', u'count': 9},
 {u'_id': u'UvA Faculteit der Economische Wetenschappen en Econometrie',
  u'count': 9},
 {u'_id': u'Kohnstammhuis', u'count': 9},
 {u'_id': u'Theo Thijssenhuis', u'count': 9},
 {u'_id': u'Roeterseiland (UvA)', u'count': 9}]
```

- Cleaning and Auditing:

The "Coffeeshops" in Amsterdam it might be confusing. Dutch call a Coffeeshop for drug shop where you can legally smoke weed, cannabis or other drugs. This information has been audited to be more clearer for tourists. The program Audit_coffeeshop.py has been used for this step.

Another auditing which has been done on the data is changing the soft drugs in both shops and cuisines.

What else can be done to the dataset?

- Amsterdam OSM dataset is very nice and tidy. Many things can be studied and analysis.
- Amsterdam is a bicycle friendly and it is the most popular city with bicycles in the world. It will be more interested if I checked the ways which contains bicycles paths.
- It is also nice to check the number of places which contains wheelchair access and which dont. The Amsterdam city council or Geementa as the Dutch call it, can use such information to improve the quality of life for those who need such service.
- Bars and pubs can be also checked and find what is the most popular.
- Amsterdam is also has many museums. I noticed that osm contain many of them. That might be a good choice to check.
- Amsterdam's old center has a lot of canals. Canals has been build very nicely. The data should contain at least the name of that canal or as the Dutch call Gracht. Gracht is Amsterdam like streets where Amsterdammers and tourists can transporting and moving around the city for site seeing and having fund.
- Also Amsterdam is also famous of open air free markets. I did not have a look deeply into the current dataset. but it would be great if they are included and then studied and analyses.
- The [mapzen](https://mapzen.com) where I downloaded the dataset, contains separated geojson (Datasets grouped into individual layers by OpenStreetMap tags (IMPOSM)) and there is an importer called [Imposm](https://imposm.org/docs/imposm/latest/) which can be used. We can insert the files directly to MongoDB nd do the analysis.
- I downloaded all those files and you can have a lot into it in a separated file with the submission.
- If someone interested, there are other ways to manipulate the OSM data I will mention just few of them here:

- [Overpass API](http://wiki.openstreetmap.org/wiki/Overpass_API)
- [Node-Mongosm](https://github.com/sammerry/node-mongosm)
- [overpy 0.4](https://pypi.python.org/pypi/overpy/)

and many others.

(I will add more ideas as suggested from the 1st reviewer.)

- OSM data can be be fixed or improved to be interactive with the users. A mobile application can be a solution for this. Via this application, the user can change directly to the file and add new information, description or rating a particular place. This way will improve and enhance the dataset very fast.
- However, one main disadvantage of implementing the above idea is the quality of the added information. Also how reliable is the added information and how useful it is.
- Another idea is link it with google maps. However, what I dont like about google map is all owned by google and that might affect OSM as open source and then it can be used in advertisements.
- We can link OSM somehow to wikipedia XML for more information and descriptions. MongoDB is useful in this concept.
- The social media plays a significant role in all Internet usage. Linking OSM to th social media specially to twitter might help in different aspects [example1](https://www.theatlantic.com/technology/archive/2013/11/how-online-mapmakers-are-helping-the-red-cross-save-lives-in-the-philippines/281366/) and [example2](https://www.newscientist.com/article/dn24565-social-media-helps-aid-efforts-after-typhoon-haiyan#.U-QaA2MmUro).
- The above are very nice examples in using OSM. However, we should encourage the users to do such great work by allocating prices and nice competitions for the best practice.
- The current dataset contains many secondary tags. Those can be summarized and grouped. However, we might loose some details which might be important for some users to answer a particular question.

Conclusion:

This kind of data format has alot of advantages. It is clear from the analysis above that many valuable information can be obtained and extracted. The cleaning process was also informative and easy to get a particular kind of information. Moreover, the auditing is another process to handel such information and change many information in easy way. Finally, there are many other ideas can be done in this king of dataset and further analysis needed to cover them and to adress some issues to improve the quality of the work.

References:

- https://docs.python.org/2/library/xml.etree.elementtree.html
- Udacity Data Wrangling with MongoDB - Exerciseshttp://fch808.github.io/Data%20Wrangling%20with%20MongoDB%20-%20Exercises.html
- http://napitupulu-jon.appspot.com/posts/wrangling-openstreetmap.html