Project Report

Problems encountered in my map

Three main problems were encountered in the Gothenburg OSM data:

1. Incorrect city names
2. Inconsistent postcode
3. Lengthy house numbers

Incorrect city names and out of boundary cities.

The first issue encountered was regarding misspellings and inconsistencies in city names. Further some cities in the map did not belong to Gothenburg municipality.

- 436 58 => Hovås
    - Postcode found among city names. This was changed to the city name refers to.
- Våstra Frölunda => Västra frölunda
    - â is nåt a Swedish letter, further frölunda should only contain lowercase letters.
- Gothenburg => Göteborg
    - Gothenburg vas changed to Swedish spelling.

Inconsistent and lengthy postcodes

The first problem encountered was that the postcode of areas in gothenburg was not labeled correctly. A standard postcode in Swedes consists of 5 digits separated by a space. Some of the faulty postcodes however looked like the following:

- 417631 => 417 63
    - Searching for 417631 postcode in the database shows the location of the postcode
        - db.map.find({“address.postcode”:”417631”})
    - The result indicates that the 1 in the end of the sequence has been added for no understandable reason.
- 12 => 412 74

    - Using same technique db.map.find({“address.postcode”:”12”}) shows this postcode belong the following sequence 412 74
- Hovås => 436 50
    - This postcode was also searched for and adjusted to following postcode
- SE-42671 => 426 71
    - This lengthy postcode was adjusted to only contain integers.

Overall: A space was added on the 3:d index of the postcode to eliminate inconsistencies and according to rules seen in the following link.

Lengthy or inconsistent house numbers

The third problem encountered was in the addr:housenumber and this was mostly related to inconsistencies in the how the numbers were presented.
For example, Some house numbers contain ‘;’ such as 6;8, 41; 41B; 43, and others contain ‘-’ such as ‘20-18’ and ‘26-38’. Further according to this link, house numbers commonly has uppercase letters, so to increase consistency all letters were capitalised.

Overview of the data

I provide a statistical overview of the Gothenburg Sweden dataset.

#All in terminal

Cd into file using the terminal

```
cd /Users/merongoitom/Desktop/Nanodegree/DataAnalysisProjects/
P3WrangleOpenStreetMapData/Github_Wrangle_OpenStreetMap_Data
```

- **Check size of the collections**
    - ls -lh data/gothenburg_sweden.osm
        - 305M Feb 18 13:55 data/gothenburg_sweden.osm
    - ls -lh data/gothenburg_sweden.osm.json
        - 442M Feb 29 10:27 data/gothenburg_sweden.osm.json
- **Start up mongoDB**
    - mongod --dbpath ~/data/db
- **Import JSON data**
    - mongoimport --db osm --collection osmb_807 --type json --file /Users/merongoitom/Desktop/Nanodegree/DataAnalysisProjects/P3WrangleOpenStreetMapData/Github_Wrangle_OpenStreetMap_Data/Data/gothenburg_sweden.osm.json
- **Start mongo**
    - mongo
- **Use map as command**
    - use map

- **Check size of the collections**
  - `db.map.dataSize()`
    - 789411292
- **Number of documents**
  - `db.map.find().count()`
    - 3374962
- **Number of nodes**
  - `db.map.find({'type' : 'node'}).count()`
    - 2995516
- **Number of ways**
  - `db.map.find({'type' : 'way'}).count()`
    - 379409
- **Number of unique users**
  - `db.map.distinct("created.user").length`
    - 823
- **Top 5 contributor by number of contributions**

  -
    ```
    db.map.aggregate([{ "$group" : { "_id" : "$created.user",
                         "count" : { "$sum" : 1 } } },
                  { "$sort" : { "count" : -1 } },
                  { "$limit" : 5 } ] )
    ```

    -
      ```
      { "_id" : "HenrikW", "count" : 1032526 }
      { "_id" : "johnrobot", "count" : 389700 }
      { "_id" : "tothod", "count" : 350240 }
      { "_id" : "archie", "count" : 193716 }
      { "_id" : "Fringillus", "count" : 139292 }
      ```
- **What percent of unique contributors made only a single contribution**

  -
    ```
    db.map.aggregate( [ { "$group" : { "_id" : "$created.user",
                         "count" : { "$sum" : 1 } } },
                  { "$group" : { "_id" : "$count",
                         "num_users" : { "$sum" : 1 } } },
                  { "$sort" : { "_id" : 1 } },
                  { "$limit" : 1 } ] )
    ```
    - `{ "_id" : 2, "num_users" : 136 }`
    - percentage 136/823 => 16.5%
- **Top city mentions in this map**

  -
    ```
    db.map.aggregate([{ "$match" : { "address.city" : { "$exists" : 1 } } },
                         { "$group" : { "_id" : "$address.city",
                                       "count" : { "$sum" : 1 } } },
                         { "$sort" : { "count" : -1 } } ] )
    ```

    -
      ```
      { "_id" : "Göteborg", "count" : 9471 }
      { "_id" : "Hisings Backa", "count" : 4952 }
      { "_id" : "Västra frölunda", "count" : 1909 }
      ```
- **Top 3 sport mentions in this map**

  -
    ```
    db.map.aggregate([{ "$match" : { "sport" : { "$exists" : 1 } } },
                         { "$group" : { "_id" : "$sport",
                                       "count" : { "$sum" : 1 } } },
                         { "$sort" : { "count" : -1 } },
    ```

    { "$limit" : 3 }])
    - { "_id" : "soccer", "count" : 342 }
      { "_id" : "tennis", "count" : 109 }
      { "_id" : "swimming", "count" : 31 }

- **Top amenities**
  - db.map.aggregate([{ "$match" : { "amenity" : { "$exists" : 1 } } },
    { "$group" : { "_id" : "$amenity",
    "count" : { "$sum" : 1 } } },
    { "$sort" : { "count" : -1} },
    { "$limit" : 10 }])
    - { "_id" : "parking", "count" : 8000 }
      { "_id" : "restaurant", "count" : 1266 }
      { "_id" : "school", "count" : 1056 }
      { "_id" : "bench", "count" : 740 }
      { "_id" : "kindergarten", "count" : 702 }
      { "_id" : "post_box", "count" : 656 }
      { "_id" : "recycling", "count" : 632 }
      { "_id" : "cafe", "count" : 580 }
      { "_id" : "fast_food", "count" : 544 }
      { "_id" : "waste_basket", "count" : 394 }

- **Top cuisines**
  - db.map.aggregate([{ "$match" : { "amenity" : { "$exists" : 1 }, "amenity":"cafe"} },
    { "$group" : { "_id" : "$cuisine",
    "count" : { "$sum" : 1 } } },
    { "$sort" : { "count" : -1} },
    { "$limit" : 10 }])
    - { "_id" : null, "count" : 514 }
      { "_id" : "pizza", "count" : 124 }
      { "_id" : "regional", "count" : 76 }
      { "_id" : "sushi", "count" : 74 }
      { "_id" : "thai", "count" : 72 }
      { "_id" : "indian", "count" : 60 }
      { "_id" : "italian", "count" : 58 }
      { "_id" : "chinese", "count" : 44 }
      { "_id" : "asian", "count" : 30 }
      { "_id" : "burger", "count" : 18 }

Other ideas about the datasets

1. **Pareto Distribution:**

   An interesting idea was to test for the Pareto Distribution whether 80% of all contributions were made by 20% of contributors. The result showed that approximately 81% of all contributions were made by 1,8% (15 contributors) of all contributors/users as seen in the results below:

How many users contributed to 80% of contributions

- **Number of contributions**
  - db.map.aggregate([{ "$group" : { "_id" : "created.user", "count" : { "$sum" : 1 } } }])
    - { "_id" : "created.user", "count" : 3374962 }
- **80% of all contributions**
    - 3374962*0.8 => 2699970
- **Top 15 contributor**
  - db.map.aggregate([{ "$group" : { "_id" : "$created.user", "count" : { "$sum" : 1 } } }, { "$sort" : { "count" : -1 } }, { "$limit" : 15 }])

    -
      ```
      { "_id" : "HenrikW", "count" : 1032526 }
      { "_id" : "johnrobot", "count" : 389700 }
      { "_id" : "tothod", "count" : 350240 }
      { "_id" : "archie", "count" : 193716 }
      { "_id" : "Fringillus", "count" : 139292 }
      { "_id" : "Niklas Gustavsson", "count" : 127446 }
      { "_id" : "tomasy", "count" : 98234 }
      { "_id" : "uebk", "count" : 82848 }
      { "_id" : "Micket", "count" : 63024 }
      { "_id" : "magol", "count" : 59226 }
      { "_id" : "kentp", "count" : 58888 }
      { "_id" : "fatal", "count" : 52612 }
      { "_id" : "elk finder", "count" : 49526 }
      { "_id" : "Ojan", "count" : 35616 }
      { "_id" : "bengibollen", "count" : 33826 }
      ```
    - The top 15 contributors (1,8%) stands for more than 80% of all contributions
    - 2732894 Sum contributions by top 15 contributors

2. **Check Zip codes:**
   Here is a document with all the ZIP codes in the municipality of Gothenburg, these could be iterated through to look for any ZIP code which falls outside Gothenburg municipality, and can be eliminated