Projet NoSQL : Best place to live in New York city

Février 2018

AUTEUR : ABOUBACAR ALKA Mahamadou Salissou
Elève Ingénieur en dernière année de formation - FIlière SID - ENSAI Rennes

Vue d'ensemble

Dans ce projet, nous essayons de mettre en oeuvre les compétences acquises dans le cadre du module de NoSQL, dans un contexte “quasi-réel”, qui va de la recherche des données brutes jusqu’à la réponse à la question posée par le projet.

Objectifs

1. Télécharger des jeux de données depuis le portail open data de la ville de New York
2. Traiter et insérer ces données dans une base de données NoSQL
3. Requêter cette base selon quelques critères afin de déterminer l’endroit idéale pour vivre à New York

I. Choix de base NoSQL & Installation préalables de modules nécessaires

1. Base de données utilisée

Pour la réalisation de ce projet, j'ai choisi MongoDB comme base de données NoSQL.

- La première raison de ce choix est la nature des données. En effet, le portail Open Data de New York est une compilation de données provenant de divers services publics de la ville, qui n'ont pas forcément le même schéma de stockage, et présentant souvent un grand nombre de données manquantes. Or MongoDB en tant que base de données "document" offre un schéma flexible nous permettant de ne stocker que les données disponibles sans s'imposer de schéma a priori.
- La deuxième raison est liée à la popularité de cette base de données. En effet, d'après le site https://db-engines.com/en/ranking , MongoDB est la première base NoSQL la plus populaire. Il m'est apparu judicieux de chercher à monter en compétence concernant cette base à l'aube de mon entrée dans le monde professionnel.
- La dernière raison est relative à la parfaite intégration qu'offre MongoDB avec Python qui est le langage de programmation que je maitrise le plus.

2. Installations de modules

Pour tester ce projet, j'ai utilisé une VM Ubuntu 17.10 avec 4G de RAM, et j'ai utilisé python pour la partie programmation. Dans cette section je liste les commandes utilisées (dans l'ordre) après l'installation de la VM vierge, afin de disposer de packages nécessaires à l'exécution du script du projet. Les commandes sont en couleur bleu pour faciliter la lecture:

- Téléchargement et Installation d'anaconda qui contient aussi la dernière version de python:
  - téléchargement : wget https://repo.continuum.io/archive/Anaconda3-5.0.1-Linux-x86_64.sh
  - installation depuis le terminal : bash ./Anaconda3-5.0.1-Linux-x86_64.sh
  - rendre anaconda accessible depuis le terminal : ma VM s'appelle "alka", donc changer ce nom en fonction de la VM : export PATH=/home/alka/anaconda3/bin:$PATH
- Installer les modules pythons nécessaires:
  - pour quelques petits calculs : conda install -c anaconda pandas
  - API pour utiliser mongodb sous python : conda install -c anaconda pymongo
  - API pour interagir avec le portail Open Data de New York : pip install sodapy
- Installation de mongodb version 3.6.2 comme expliquée sur le site officiel:
  - sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 2930ADAE8CAF5059EE73BB4B58712A2291FA4AD5
  - echo "deb [arch=amd64,arm64] https://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/3.6 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.6.list
  - sudo apt-get update
  - sudo apt-get install -y mongodb-org

Les installations étant faites, on passe au téléchargement des données. Pour cela il faut démarrer MongoDB sur la machine depuis le terminal ubuntu : sudo service mongod start

II. Téléchargement, traitement et requêtage des données

1. Téléchargement des données : fichier Script_load_data.py

Pour télécharger les données nous utilisons l'API sodapy du portail Open Data de New York depuis python qui nous permet de télécharger n'importe quel jeu de données, avec la possibilité d'effectuer des filtres, tris, etc...

Une fois les données téléchargées, on crée une BDD mongoDB et on insère ces données. Pour télécharger les données et les insérer les données dans la base de données il suffit de lancer le fichier Script_load_data.py depuis le terminal ubuntu avec la commande python3 Script_load_data.py. Si le lecteur est intéressé par plus de détails, il pourra consulter le script python Script_load_data.py qui est bien commenté pour chaque bloc d'instructions.

A l'issue de l'exécution de ce code nous aurons dans notre MongoDB une BDD appelée OpendataNY comportant 4 collections :

- Requests : ce sont les plaintes non urgentes déposées par les New Yorkais au niveau du service 311 (plaintes pour bruit, rats/souris, ...)
- policeComplaints : plaintes enregistrées par la police New Yorkaise (meurtres, viols, vols, drogues, ...)
- WaterQuality : plaintes déposées par les New Yorkais, relatives à la qualité de l'eau (particules dans l'eau, couleur anormale, odeur anormale, ...)
- Housing : la base de données des logements avec leurs adresses dans la ville de New York

Remarque : Dans ma VM, MongoDB écoute par défaut sur le port 27017. Si tel n'est pas le cas, alors modifier la ligne 44 du script Script_load_data.py pour mettre le port adéquat :

- clientMongoDB = MongoClient('mongodb://localhost:numéro_du_port/')

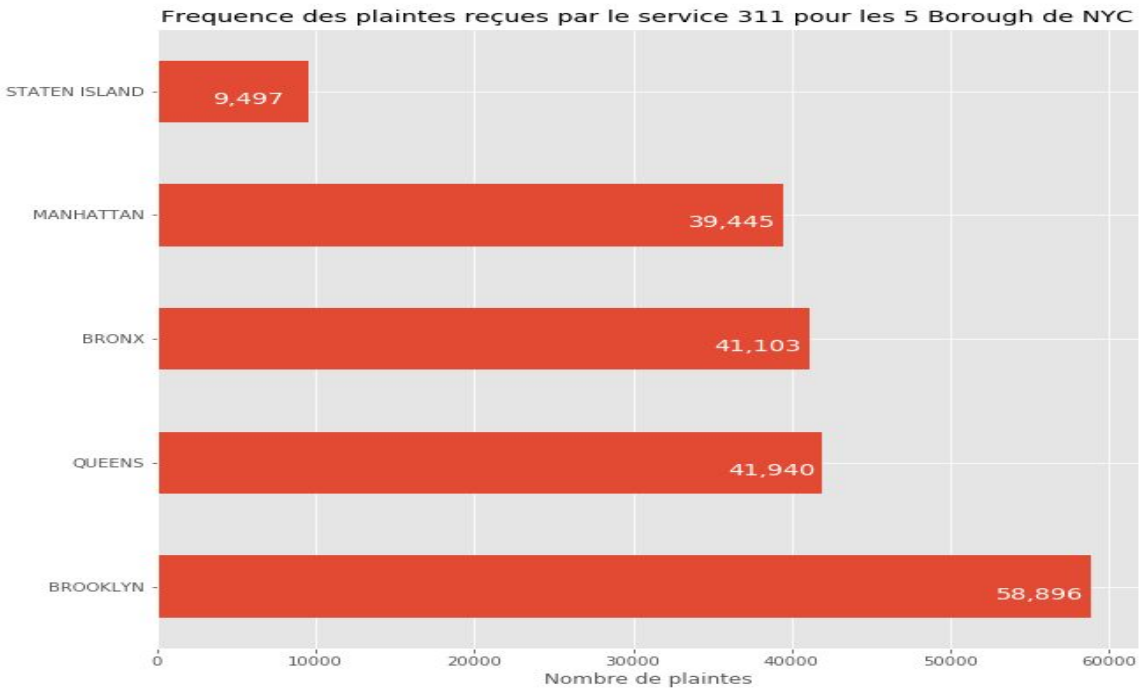
2. Requêtage des données : fichier Queries_bestPlaces.py

Remarque : pour obtenir les résultats de cette partie "analyse" du projet, il suffit juste d'exécuter le script Queries_bestPlaces.py depuis un terminal Ubuntu avec la commande python3 Queries_bestPlaces.py. Cette commande affichera dans le terminal quelques commentaires des résultats obtenus, et elle produira également 4 graphiques (présentés ci-dessous) qui peuvent être récupérés dans le répertoire où se trouve le script.

Une fois les données téléchargées, nous passons au requêtage de ces dernières afin de répondre à la problématique du projet : "**le meilleur endroit pour vivre à New York"**. Précisons que les critères utilisés pour déterminer le meilleur endroit sont purement subjectifs, quoique réalistes.

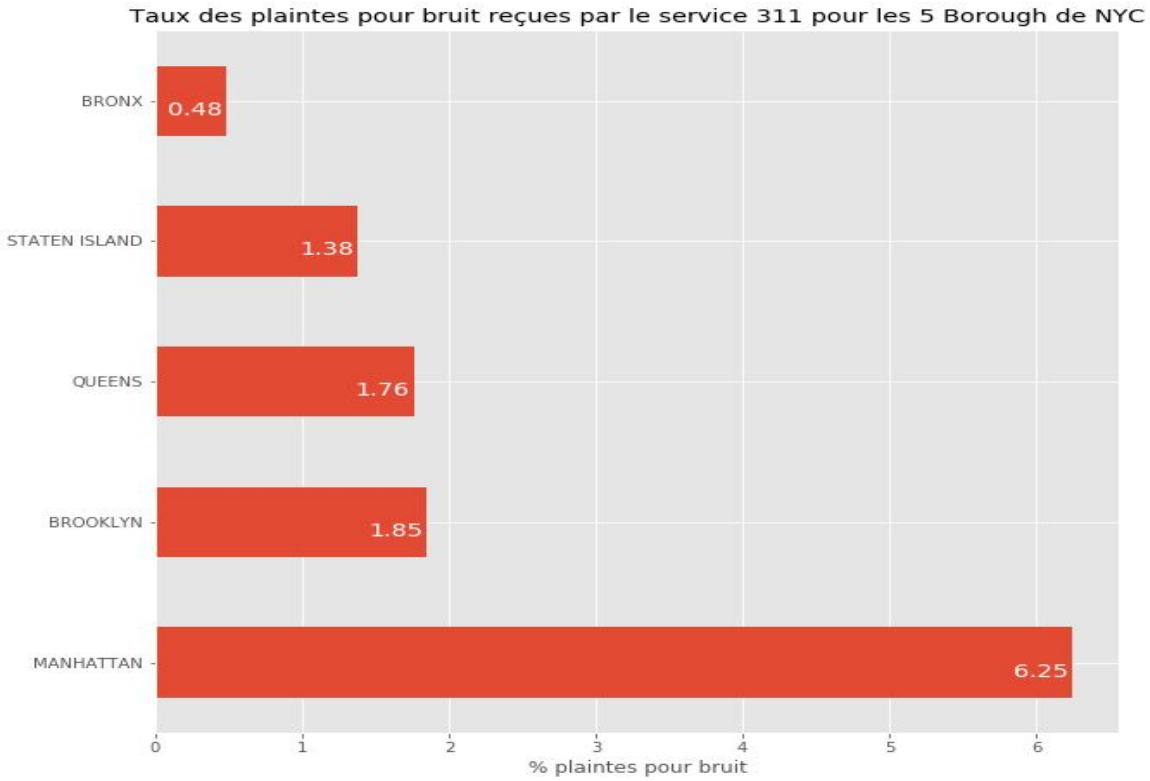
New York est constituée de 5 entités administratives appelées **Borough** : **MANHATTAN, BRONX, BROOKLYN, QUEENS,** et **STATEN ISLAND**. La démarche adoptée ici est hiérarchique : tout d'abord déterminer le "Borough idéal", puis la "résidence idéale" au sein de ce Borough. Ci-après la démarche:

- Déterminer la répartition des plaintes reçues par le service 311 par Borough



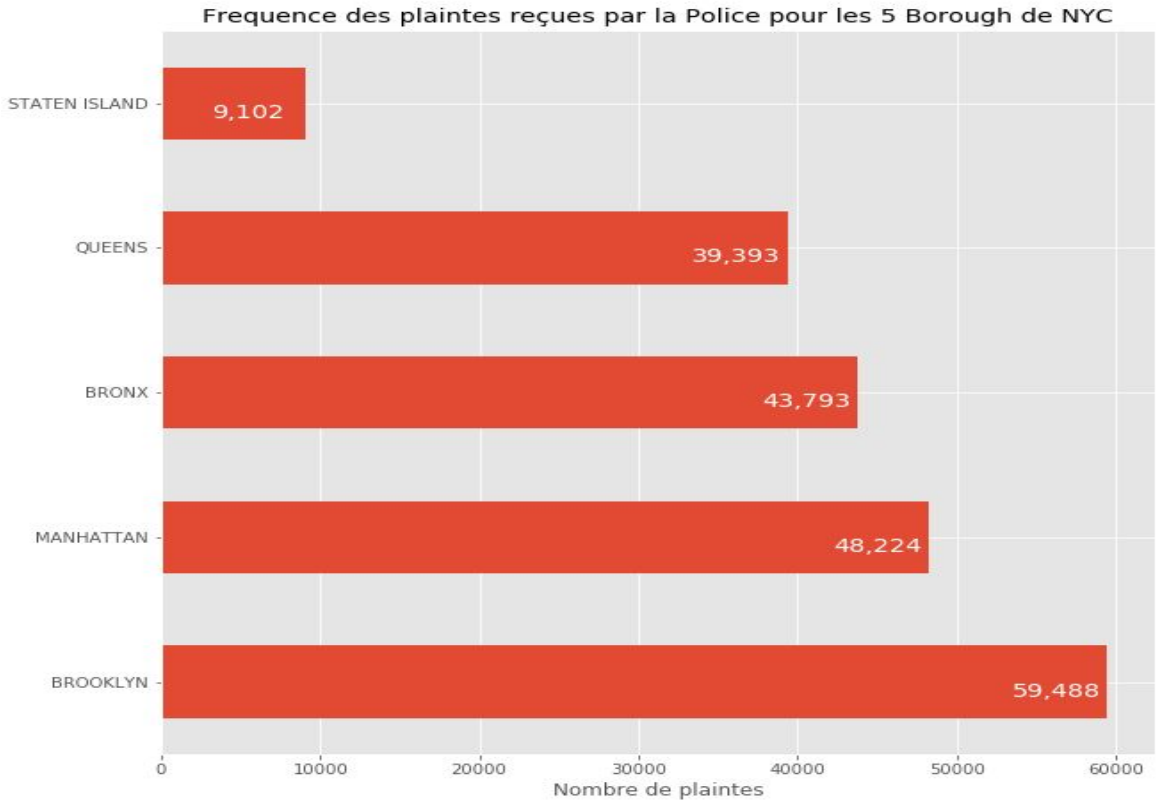
On remarque que le Borough de **STATEN ISLAND** (9 497) est celui qui a enregistré de le moins de plaintes déposées au service 311, contrairement à BROOKLYN qui en a enregistré le plus (58 896).

- Voir le taux de plaintes pour bruits enregistrées par le service 311 pour chaque Borough. Pour cela on utilise les expressions régulières pour rechercher les plaintes dont la description contient l'un des termes : **noise, noisy, nois**



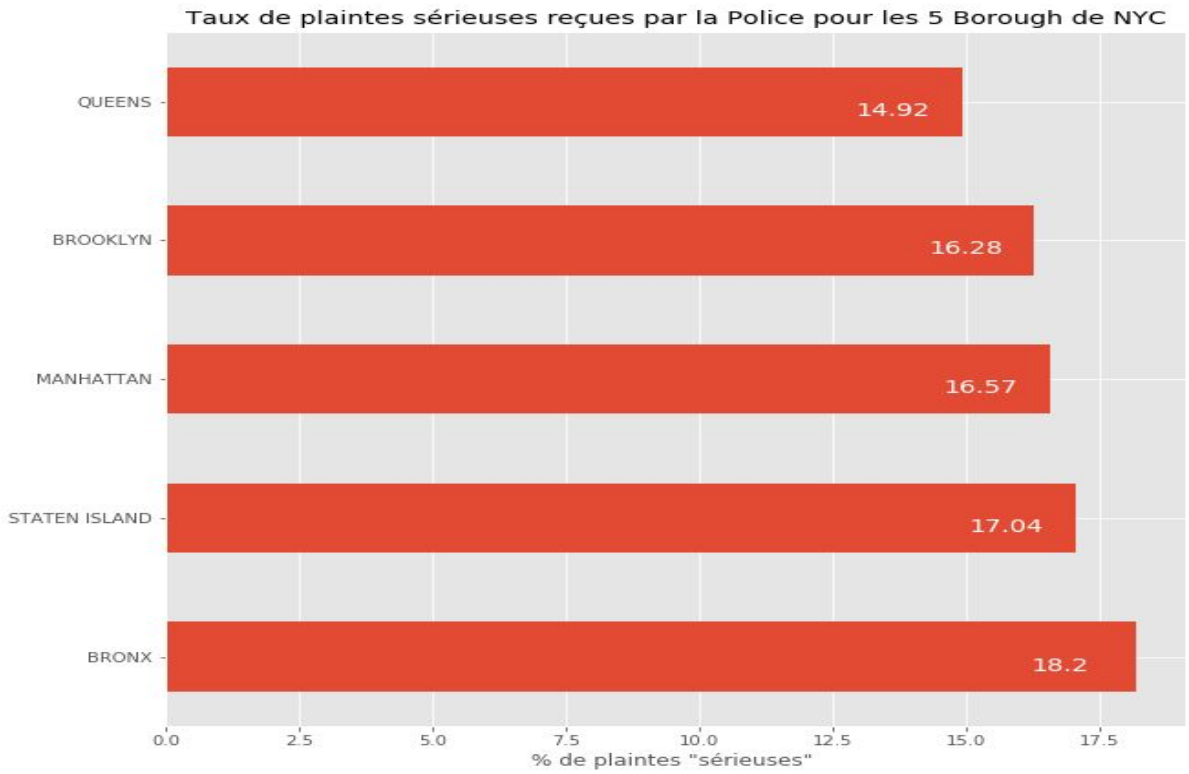
Là par contre c'est le BRONX qui sort gagnant avec le plus faible taux de plaintes pour bruit (0.48 %).

- Par la suite nous nous intéressons aux plaintes déposées auprès de la Police et nous regardons la répartition de ces dernières entre les 5 Borough:



Comme pour les plaintes du service 311, ici aussi ce sont les Borough de STATEN ISLAND et BROOKLYN qui ont enregistré respectivement le moins et le plus de plaintes déposées à la Police. Ceci fait penser à un effet taille : les Borough avec plus sont plus enclins à enregistrer le plus de plaintes.

- Pour décider finalement quel Borough choisir, nous allons regarder celui ayant enregistré le plus faible taux de plaintes pour “**crimes graves**” parmi celles enregistrées par la Police. Pour ce faire nous utilisons les expressions régulières pour rechercher les plaintes contenant les termes : "crim", "drug", "sex", "alcoh", "murder", "kidnap"



Ici c'est **QUEENS** qui sort gagnant. Ce Borough occupe des "positions plus ou moins médianes" dans les précédents classements. Donc l sera finalement retenu pour la recherche de la résidence idéale.

- Une fois le Borough déterminé, nous procédons à la recherche de la résidence. Pour cela nous utilisons les coordonnées géographiques (longitude, latitude) de chaque résidence située dans le QUEENS pour déterminer le nombre de plaintes déposées pour mauvaise qualité de l'eau dans un rayon de 5 kilomètres.

Les résidences idéales où je compte déposer mon dossier de candidature seront les 5 résidences avec le moins de plaintes dans un rayon de 5 kilomètres.

	_id	borough	coord	lot	nombre_plaintes_eau	parcel_address	postcode
0	5a7eeeb94b63c0132a263a26	QUEENS	[-73.864681, 40.569046]	227	46.0	149-25 ROCKAWAY BEACH BOULEVARD	11694
1	5a7eeeb94b63c0132a2635a4	QUEENS	[-73.728814, 40.657913]	15	64.0	258-18 FRANCIS LEWIS BOULEVARD	11422
2	5a7eeeb94b63c0132a2635af	QUEENS	[-73.740018, 40.649029]	90	65.0	HOOK CREEK BOUL	11422
3	5a7eeeb94b63c0132a2635cf	QUEENS	[-73.740018, 40.649029]	6	65.0	HOOK CREEK BOUL	11422
4	5a7eeeb94b63c0132a2635a9	QUEENS	[-73.736395, 40.653006]	1	68.0	253-50 149 AVENUE	11422