université
PARIS-SACLAY

Rapport

Analyse et Visualisation des Données de Transport à New York

Tarek ATBI

25 Novembre 2024

Université Paris Saclay

Introduction

Ce projet consiste en l'analyse des données de trajets réalisés par des taxis jaunes, taxis verts et véhicules de transport avec chauffeur (VTC) à New York. L'objectif est de transformer des données brutes en une application interactive permettant de visualiser les trajets et d'extraire des statistiques utiles pour des prises de décision basées sur les données.

Objectifs

- Préparer et nettoyer les données pour harmoniser les formats entre les datasets.
- Intégrer les coordonnées géographiques (si besoin) pour chaque trajet afin de permettre la visualisation sur carte.
- Développer une interface utilisateur interactive pour :
    - Filtrer les données par type de véhicule, date et localisation (pick-up/drop-off).
    - Afficher des cartes interactives et des heatmaps des trajets.
    - Générer des statistiques détaillées, comme le nombre de trajets, la distance totale et les données sur les passagers.
- Faciliter l'exploration des données à l'aide de kepler.gl et/ou Streamlit.

Architecture

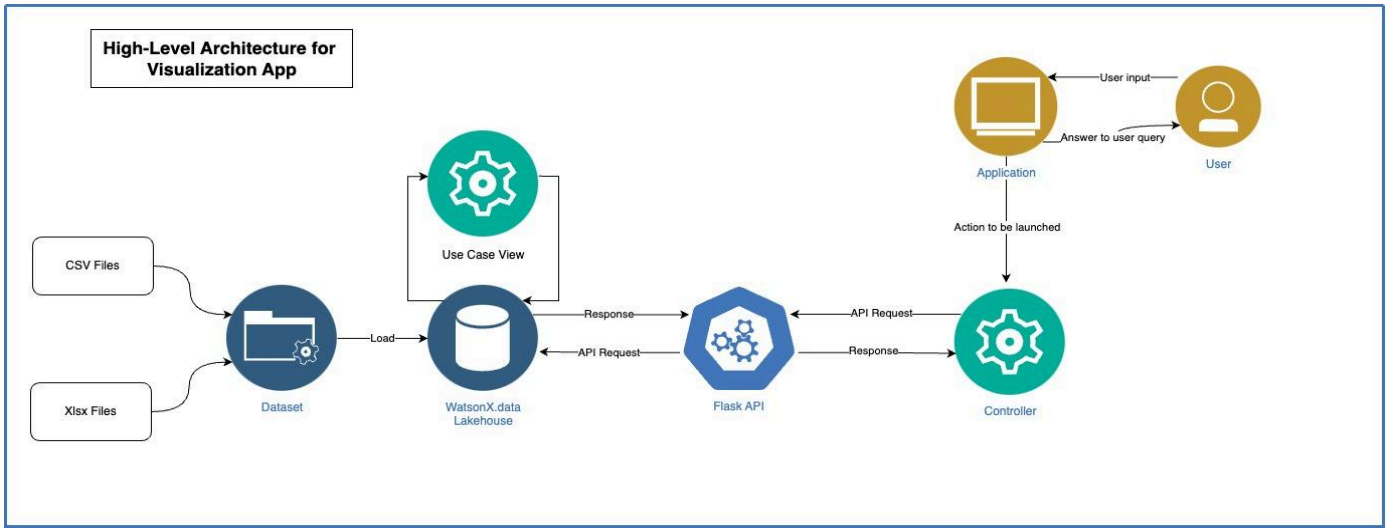


Figure -1- : Architecture High level de l'Application

Description des Données

Pour ce projet, trois datasets principaux ont été utilisés : yellow taxis, green taxis, et VTC. Chaque dataset contient des informations détaillées sur les trajets réalisés à New York.

1. **Datasets initiaux**
    a. Yellow Taxis
        i. Données : Coordonnées des points de départ et d'arrivée, temps, nombre de passagers, distance, montants des courses, et type de paiement.
        ii. Particularité : Représente les taxis traditionnels opérant principalement dans **Manhattan**.
    b. Green Taxis
        i. Données : Similaires aux yellow taxis, avec des services concentrés sur les zones périphériques et **hors Manhattan**.
    c. VTC (Véhicules de Transport avec Chauffeur)
        i. Données : Coordonnées des pick-ups manquantes (seul un identifiant LocationID était disponible), temps, et identifiants des bases de dispatch.
        ii. Particularité : Absence des coordonnées des **drop-offs**.
        iii. Solution : Utilisation d'un **dataset externe** contenant les coordonnées géographiques (latitude, longitude) associées aux **LocationID**.
2. **Structure des données nettoyées**
    a. Colonnes communes
        i. *pickup_time, pickup_lat, pickup_lon*
        ii. *dropoff_time, dropoff_lat, dropoff_lon* (sauf pour VTC, limité aux pick-ups)
        iii. *type :* Indique le type de véhicule (yellow, green, vtc).
3. **Format et stockage**
    a. Les données nettoyées sont stockées dans des fichiers **Parquet** sur un COS pour une lecture rapide et efficace.
    b. Résumé des données générées en fichiers **CSV** pour les statistiques globales.
4. **Portée temporelle**
    a. Les données couvrent la période du **1er au 18 janvier 2015**, offrant une vue détaillée d'une période fixe pour simplifier l'analyse.
5. **Limites et contraintes**
    a. Absence des coordonnées des drop-offs dans les données VTC.
    b. Nécessité d'intégrer un dataset complémentaire pour convertir les `LocationID` en coordonnées géographiques dans les données VTC.
    c. Uniformisation des colonnes et des formats temporels pour une exploitation harmonisée.

Étapes du Projet

4.1 Préparation des Données

1. **Chargement des datasets**
   - Importation des fichiers sources pour les taxis jaunes, verts, et les VTC.
2. **Nettoyage et harmonisation**
   - Suppression des colonnes non pertinentes pour réduire la complexité.
   - Conversion des timestamps (`pickup_time`, `dropoff_time`) en formats lisibles.
   - Ajout des coordonnées géographiques pour les VTC en mappant les `LocationID` à un dataset externe.

```python
import pandas as pd
from shapely import wkt

zones['coordinates'] = zones['the_geom'].apply(wkt.loads)  # Convertir en objets shapely
zones['pickup_longitude'] = zones['coordinates'].apply(lambda x: x.centroid.x)
zones['pickup_latitude'] = zones['coordinates'].apply(lambda x: x.centroid.y)

print(zones[['LocationID', 'pickup_longitude', 'pickup_latitude']].head())

# Joindre les données des coordonnées aux données VTC en utilisant 'pickup_location_id'
vtc_data_with_coords = filtered_vtc_data.merge(zones[['LocationID', 'pickup_longitude', 'pickup_latitude']],
                                               how='left',
                                               left_on='pickup_location_id',
                                               right_on='LocationID')

# Sauvegarder le dataframe enrichi (optionnel)
vtc_data_with_coords.to_parquet("vtc_data_with_coordinates.parquet")
```

Figure -2- : Code Python pour rajouter les coordonnées géographique au dataset VTC

3. **Concaténation des datasets**
   - Uniformisation des colonnes pour fusionner les trois datasets en un seul.
   - Ajout d'une colonne `type` pour différencier les trajets yellow, green, et VTC.

4.2 Développement de l'Application

1. **Choix de la plateforme**
   - Utilisation de **Streamlit** pour une interface web interactive et simple.
   - Intégration de **Kepler.gl** pour une visualisation cartographique avancée.

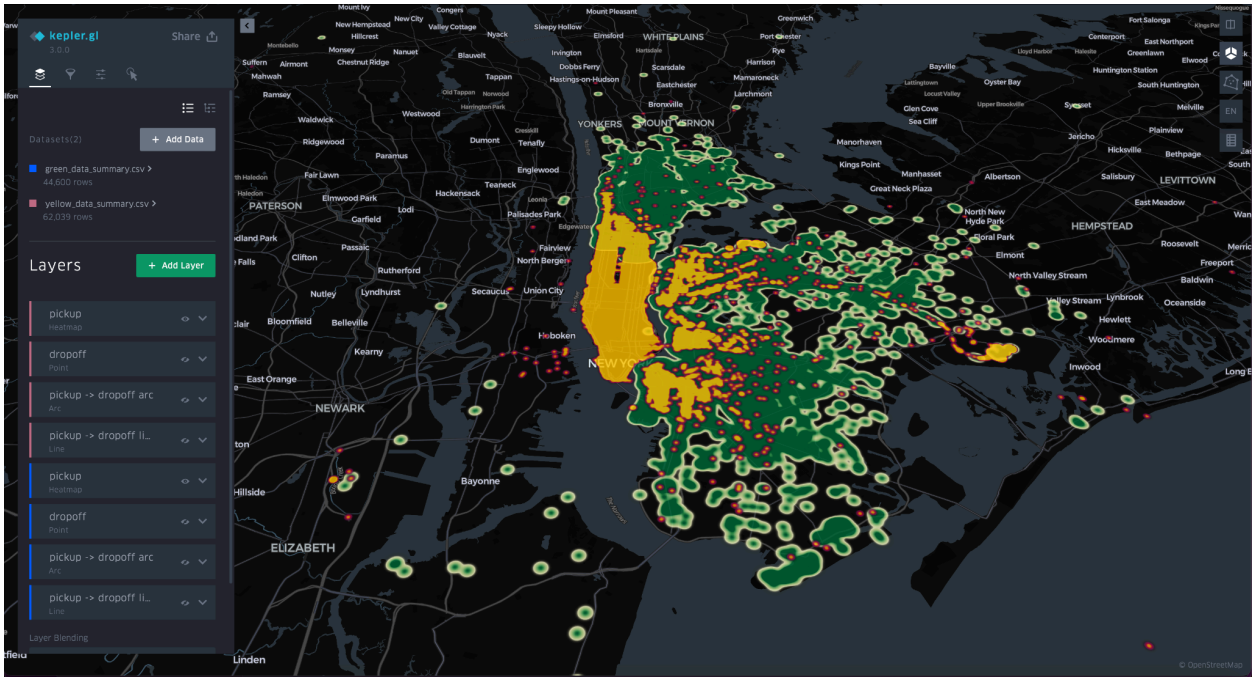


Figure -3- : Visualisation des Pickup Yellow vs Green

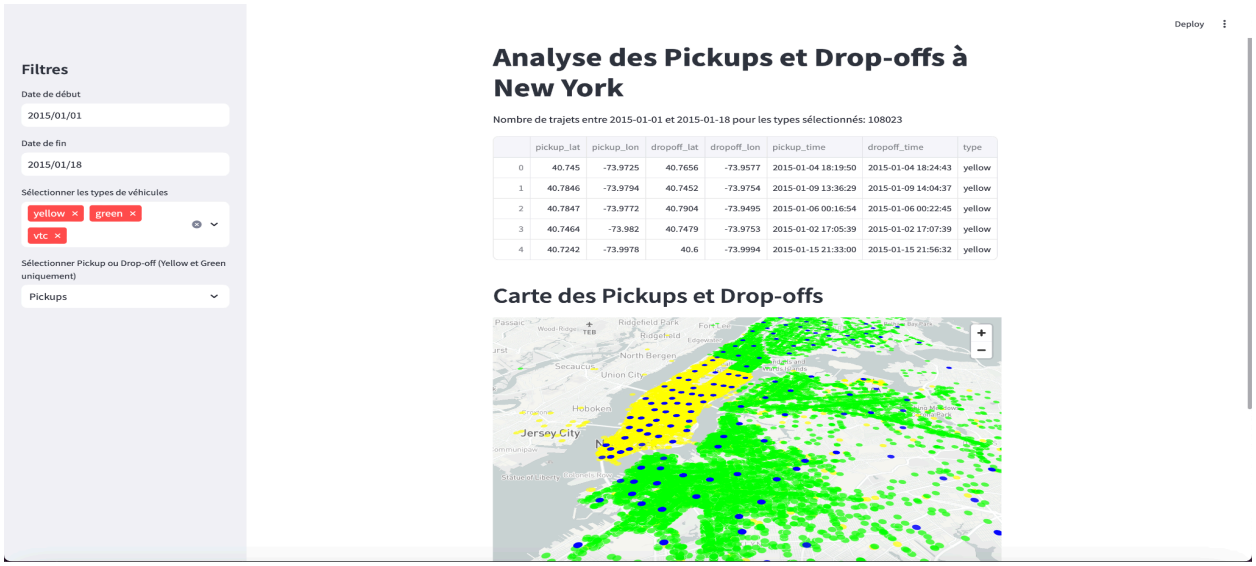


Figure -4- : Interface Customisé pour le calcul des statistiques

2. **Fonctionnalités implémentées**
    - **Cartes interactives** : Visualisation des points de départ et d'arrivée.
    - **Filtres dynamiques** : Sélection par plage de dates, type de véhicule, et type de localisation (pick-up ou drop-off).

- **Statistiques** : Calculs agrégés sur les distances, le nombre de passagers, et les revenus totaux.

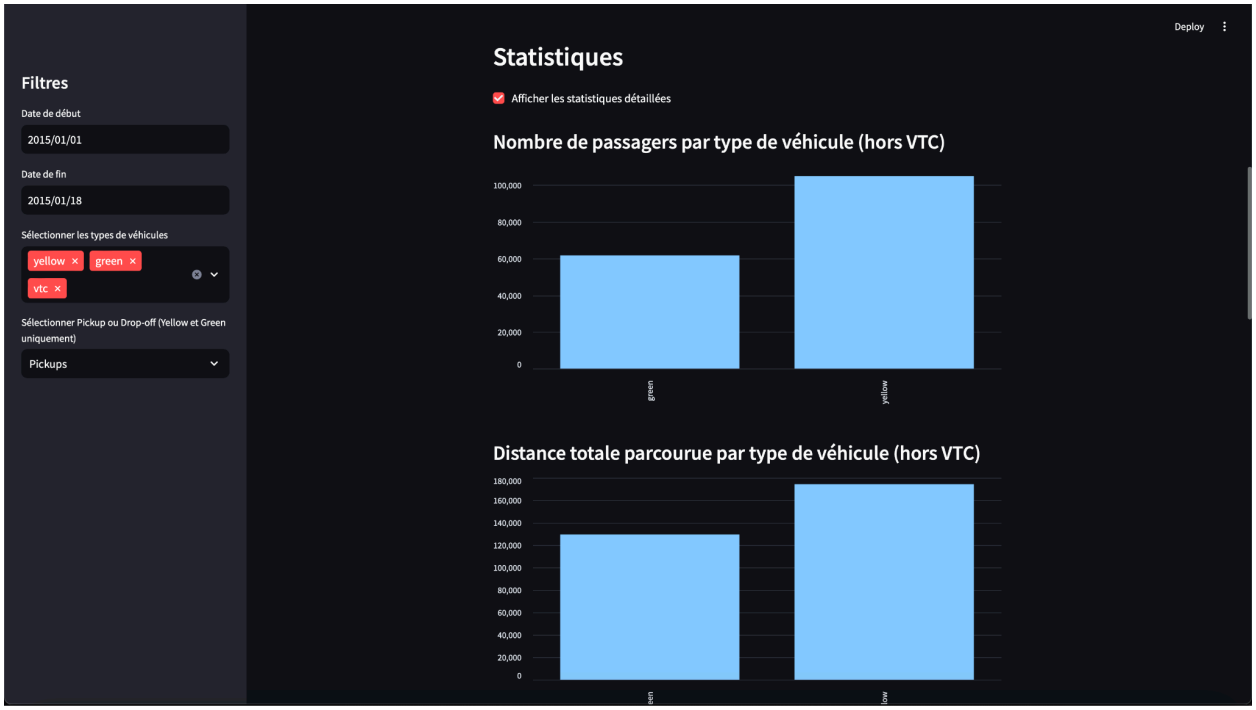


Figure -5- : Statistiques sur le dataset

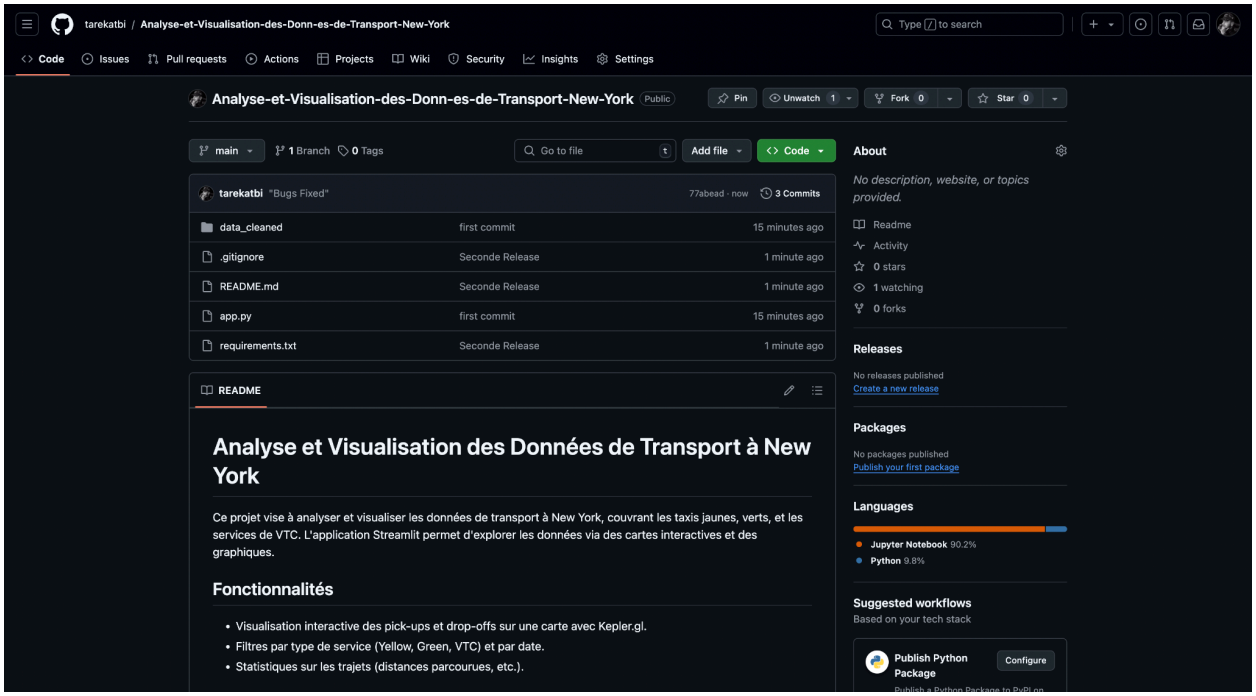
4.3 Analyse et Visualisation

1. **Exploration des données**
   - Création de heatmaps pour visualiser les zones les plus fréquentées (pick-ups et drop-offs).
   - Mise en évidence des différences d'usage entre taxis jaunes, verts, et VTC.
2. **Statistiques descriptives**
   - Agrégation par type de véhicule pour analyser les comportements de trajet et les tendances.
3. **Rapport visuel**
   - Génération de graphiques dynamiques et de cartes interactives pour accompagner les résultats analytiques.

4.4 Livrables

1. Fichiers nettoyés en **Parquet** et résumés en **CSV**.

2. Application **Streamlit/Kepler.gl** fonctionnelle pour la visualisation et l'exploration des données.
3. Rapport final détaillant les méthodologies, résultats et insights.



Conclusion

Ce projet a permis de centraliser et de visualiser les données relatives aux trajets effectués par les taxis jaunes, les taxis verts, et les VTC à New York. En intégrant des outils tels que Streamlit et Kepler.gl, nous avons développé une application interactive qui facilite l'analyse des comportements de déplacement, Les principales réalisations incluent :

- La préparation et la normalisation des données issues de différentes sources.
- La création de cartes interactives et de filtres dynamiques pour une exploration approfondie des trajets.
- L'identification des zones les plus fréquentées et des tendances spécifiques aux différents types de transport.

Cette approche offre une base solide pour des analyses futures et des prises de décisions stratégiques dans le domaine du transport urbain. L'application peut également être étendue pour inclure des prédictions ou des recommandations basées sur des données historiques.