Page de garde

Sommaire

Contexte

1 – Jeux de données

Dans le cadre de notre projet sur les accidents routiers en France et leur gravité, nous avions accès à deux sources de données principales :

- https://www.data.gouv.fr

- https://www.kaggle.com

1.1 Jeux de données issus de Data.gouv.fr

Sur la plateforme Data.gouv.fr, nous disposions des données couvrant la période de 2005 à 2022, réparties en plusieurs fichiers :

- Usager\_annee.csv

- Véhicules\_annee.csv

- Lieux\_annee.csv

- Caractéristiques\_annee.csv

- Véhicule\_immatricules\_baac\_annee.csv

- Des fichiers descriptifs en format PDF, variables selon les années

- Des bases de données déjà agrégées

a. Récupération des fichiers

Pour extraire ces fichiers, nous avons utilisé l'outil \*\*Octoparse\*\* afin d'obtenir un fichier CSV contenant l'ensemble des liens vers les différents fichiers de données.

Ensuite, à l'aide de \*\*Python\*\*, nous avons développé une fonction spécifique permettant de télécharger et d'extraire automatiquement toutes les données disponibles sur la plateforme (voir Annexe 1 pour le code).

b. Analyse des données

Nous avons utilisé un modèle de template pour examiner en détail les colonnes de chaque fichier, en fournissant des informations telles que :

- Le nom de la colonne

- La description de la colonne

- Le type de données

- Le taux de valeurs manquantes (NaN)

- La distribution des valeurs

- Des observations et remarques pertinentes

Cette analyse a révélé que certaines colonnes ont évolué au fil du temps, n'apparaissant qu'à partir de certaines années, ou cessant d'être collectées à partir d'une certaine date.

- Fichier Caractéristiques: Ce fichier décrit les circonstances générales de chaque accident. Il contient 17 colonnes, dont deux sont des doublons (\*Accident\_ID\* et \*Num\_ACC\*). Nous avons constaté un grand nombre de valeurs manquantes dans les données géographiques (coordonnées GPS, adresses).

- Fichier Lieux : Ce fichier décrit le lieu principal de l'accident, y compris lorsque l'accident se produit à une intersection. Il contient 19 colonnes. De nombreuses valeurs sont manquantes, notamment pour des variables liées à la route (ex. indices de route comme 2bis, 3ter), ou des détails comme la distance par rapport à une borne, la largeur du terre-plein central, et la vitesse maximale autorisée.

- Fichier Véhicules : Ce fichier décrit les véhicules impliqués dans l'accident. Il contient 11 colonnes. De nombreuses valeurs sont manquantes, en particulier pour les variables liées au nombre d'occupants, à l'identifiant du véhicule et au type de moteur.

- Fichier Usager : Ce fichier décrit les personnes impliquées dans l'accident. Il contient 17 colonnes. Les valeurs manquantes concernent principalement les variables relatives à la sécurité (celles-ci ayant évolué dans le temps), l'identifiant du véhicule et celui de l'usager.

Nous avons décidé de ne pas inclure, pour le moment, le fichier Véhicule\_immatriculé\_baac dans notre analyse.

1.2 Jeux de données issus de Kaggle

Le jeu de données fourni par Kaggle est déjà agrégé et contient 55 colonnes. Toutefois, il ne couvre que la période jusqu'à 2016.

Détail fichier de Margot

1.3 Choix de la source de données et de la variable cible

Après cette première phase d'analyse, nous avons déterminé que la gravité de l'accident serait notre variable cible pour l'étude.

Après une discussion collective, nous avons fait le choix de conserver les données issues de Data.gouv.fr pour deux raisons principales :

1. Maîtrise des données : En sélectionnant cette source, nous avons un contrôle total sur la construction du jeu de données, incluant le choix des années à analyser et des jointures à réaliser.

2. Actualité des données : Les données sur Data.gouv.fr sont plus récentes. Nous estimons que l'évolution des infrastructures routières, les progrès en matière de sécurité des véhicules, ainsi que les mesures mises en place par l'État en faveur de la sécurité routière, sont des éléments importants à prendre en compte dans l'analyse de la gravité des accidents.

2 – Analyse du jeu de données

Une fois la source choisie, nous avons analyser avec Python plus en détails les composantes que nous avions à notre disposition en vue de créer un DF pour la data visualisation.

L'objectif de l'analyse a été de traiter et d'examiner des fichiers CSV provenant de différentes années et de divers types. Voici les principales étapes et résultats de l'analyse :

1. Contexte et Objectif

L'analyse portait sur plusieurs fichiers CSV situés dans un dossier spécifique, couvrant les années de 2005 à 2022. Les fichiers appartenaient à différents types, notamment des caractéristiques, des lieux, des usagers, des véhicules, et des immatriculations de véhicules.

2. Méthodologie

Les fichiers ont été traités en essayant plusieurs séparateurs (virgule, point-virgule, tabulation) pour assurer une lecture correcte des données. Pour chaque fichier, un DataFrame a été créé et les statistiques descriptives ont été calculées pour chaque colonne. Ces statistiques incluent le type de données, le taux de valeurs manquantes, le mode, les valeurs minimales et maximales, la moyenne, la médiane, et l'écart-type.

3. Résultats

- Résultats d'Analyse : Un résumé détaillé a été produit pour chaque fichier, incluant le nombre de colonnes et de lignes, les noms des colonnes, et les informations sur le succès ou l'échec de la lecture du fichier.

- Statistiques des Colonnes: Des statistiques détaillées ont été extraites pour chaque colonne des fichiers analysés, fournissant des informations clés sur la distribution des données.

4. Exportation des Résultats

Les résultats d'analyse et les statistiques des colonnes ont été préparés pour une éventuelle exportation en fichier Excel, facilitant la présentation et l'interprétation des données.

En conclusion, cette analyse a permis de compiler des informations précieuses sur la structure et la qualité des données contenues dans les fichiers CSV, offrant une base solide pour des analyses ultérieures.

3 - Concaténation des fichiers par types et fusion d’un DF

Chronologie :

Fréquence de point

Partage des tâches du mode projet

24-06 : Attribution du projet et présentation de Christophe

01-07 : Cadrage initial et découverte de l’équipe

11-07 : Mise en place du drive, analyse des deux sources de données par binôme.

17-07 : Mise en commun exploration des données

18-07 : Choix de la source, mise en place d’un github, data viz et partage du travail

23-07 : Point sur avancement, sélection des variables, méthode de fusion du DF

01-08 : Point sur data viz et choix des visuels à présenter

06-08 : Mise au propre du github, et sélection finale des présentation

08-08 : Point avec Christophe, rajout de visuel type carte, début du préprocessing, amélioration de visuel existant

14-08: A compléter je n’étais pas là

09-09 : Point sur le préprocessing, début du rapport

13-09 : Point avec Christophe pour préprocessing et rapport

Annexes 1 :

Fonction pour récupérer les fichiers Data.gouv.fr

import os

import csv

import requests

# Chemin vers le fichier CSV contenant les titres et les liens

csv\_file = 'liens.csv'

# Dossier où les fichiers seront enregistrés

output\_folder = 'data/'

# Vérifier si le dossier output\_folder existe, sinon le créer

os.makedirs(output\_folder, exist\_ok=True)

# Fonction pour télécharger un fichier depuis une URL et l'enregistrer localement

def download\_file(url, filename):

try:

# Effectuer la requête HTTP pour récupérer le contenu du fichier

response = requests.get(url)

if response.status\_code == 200:

# Écrire le contenu du fichier téléchargé dans un fichier local

with open(filename, 'wb') as f:

f.write(response.content)

print(f"Le fichier '{filename}' a été téléchargé avec succès.")

else:

print(f"Échec du téléchargement du fichier '{filename}'. Statut HTTP {response.status\_code}")

except Exception as e:

print(f"Erreur lors du téléchargement du fichier '{filename}': {str(e)}")

# Lecture du fichier CSV et téléchargement des fichiers

with open(csv\_file, 'r', encoding='utf-8') as file:

csv\_reader = csv.DictReader(file)

for row in csv\_reader:

titre = row['Titre']

lien = row['frbtn\_lien']

# Construire le chemin complet où enregistrer le fichier

file\_path = os.path.join(output\_folder, titre)

# Télécharger le fichier depuis l'URL spécifiée

download\_file(lien, file\_path)