[PROJET FIL ROUGE]

ACCIDENTS ROUTIERS EN FRANCE ENTRE 2005 à 2022

Équipe projet :

|  |  |
| --- | --- |
| Noms |  |
| Sophie Doublier | Gabriel Del Vecchio |
| Marcel Njapo | Stéphane Martinez |
|  |  |
| Mentor Projet | Elliot |

Rédaction du document :

|  |  |
| --- | --- |
|  | Nom |
| Rédacteurs | Équipe projet |
| Vérificateur |  |
| Approbateur |  |

Versioning du document :

|  |  |  |
| --- | --- | --- |
|  | Désignation | Date |
| V.1.0 | Draft du document | Mai 2024 |
|  |  |  |
|  |  |  |

SOMMAIRE

[1. Contexte et périmètre du projet 4](#_Toc168384122)

[1.1. Contexte d’insertion du projet dans notre métier 4](#_Toc168384123)

[1.2. Les membres du groupe 4](#_Toc168384124)

[1.3. Objectifs du projet 5](#_Toc168384125)

[2. PARTIE 1 – Consolidation des données 6](#_Toc168384126)

[2.1. Méthodologie 6](#_Toc168384127)

[2.2. Analyse des sources de données 7](#_Toc168384128)

[2.2.1. Structure des données disponibles 7](#_Toc168384129)

[2.2.2. Les données issues de Kaggle 7](#_Toc168384130)

[2.2.3. Les données issues du gouvernement 7](#_Toc168384131)

[2.2.4. Macro-Analyse des données de chacune des 2 sources 7](#_Toc168384132)

[2.3. Analyse détaillée et Consolidation des données 8](#_Toc168384133)

[2.3.1. Étapes de consolidation des données 8](#_Toc168384134)

[2.3.2. Règles d’enrichissement appliquées 9](#_Toc168384135)

[2.3.3. Consolidation fichiers « Caractéristiques » 9](#_Toc168384136)

[2.3.4. Consolidation fichiers « Lieux » 12](#_Toc168384137)

[2.3.5. Consolidation fichiers « Vehicules » 14](#_Toc168384138)

[2.3.6. Consolidation fichiers « usagers » 18](#_Toc168384139)

[2.4. Merge global des données 23](#_Toc168384140)

[2.4.1. Contrôles préliminaires des données 23](#_Toc168384141)

[2.4.2. Merge du fichier caractéristiques avec le fichier lieux 24](#_Toc168384142)

[2.4.3. Merge avec fichier véhicules 24](#_Toc168384143)

[2.4.4. Merge avec fichier usagers 24](#_Toc168384144)

[2.4.5. Agrégation des données 25](#_Toc168384145)

[2.4.6. Renommage de chacune des variables 27](#_Toc168384146)

[2.5. Synthèse Consolidation des données 27](#_Toc168384147)

[3. PARTIE 2 – Analyse exploratoire des données 29](#_Toc168384148)

[3.1. Méthodologie 29](#_Toc168384149)

[3.2. Analyse préliminaire 30](#_Toc168384150)

[3.2.1. Périmètre de données retenu 30](#_Toc168384151)

[3.2.2. Contrôles préalables des données (visualisation globale) 31](#_Toc168384152)

[3.2.3. Séparation de la base et méthode d’évaluation du modèle 35](#_Toc168384153)

[3.2.4. Regroupement des variables par typologie 36](#_Toc168384154)

[3.3. Analyse de la variable cible ‘user\_gravite’ 36](#_Toc168384155)

[3.4. Analyse des variables explicatives 37](#_Toc168384156)

[3.4.1. Les variables d’index 37](#_Toc168384157)

[3.4.2. Les variables date/time 37](#_Toc168384158)

[3.4.3. Analyse détaillée des variables catégorielles binaires 39](#_Toc168384159)

[3.4.4. Analyse détaillée des variables catégorielles non binaire 40](#_Toc168384160)

[3.4.5. Analyse détaillée des variables quantitatives 43](#_Toc168384161)

[3.4.6. Matrice de corrélation 44](#_Toc168384162)

[3.5. Analyse des résultats statistiques 45](#_Toc168384163)

[4. PARTIE 3 - Automatisation des opérations de préprocessing 49](#_Toc168384164)

[4.1. Déclaration des fonctions de transformation 49](#_Toc168384165)

[4.2. Déclaration des pipelines de préprocessing 50](#_Toc168384166)

[4.3. Déclaration du pipeline global 52](#_Toc168384167)

[4.4. Exécution de notre pipeline via un GridSearchCV 54](#_Toc168384168)

[4.5. Fin étape automatisation du préprocessing 55](#_Toc168384169)

# Contexte et périmètre du projet

## Contexte d’insertion du projet dans notre métier

***Point de vue technique***

Ce projet nous permet de mettre en pratique bon nombre de notions et concepts théoriques abordés au fil de notre cursus de formation.

***Point de vue économique***

Par l’acquisition de nouvelles compétences pratiques en analyse de données et en Machine Learning durant l’élaboration de ce projet, notre métier actuel est voué à progresser et à être davantage valorisé.

***Point de vue scientifique***

Ce projet nous oblige à acquérir ou/et perfectionner une certaine rigueur scientifique et de revoir des notions essentielles en mathématiques et statistiques.

## Les membres du groupe

L’équipe est constituée de membres d’horizon différents ayant une expertise autour des métiers de la data, de l’informatique et des méthodes de travail qui leur sont propres.

***Gabriel Del Vecchio :***

Commercial dans le domaine de la data, je travaille avec des outils analytics au quotidien. Bien que la problématique des accidents routiers et la méthodologie d’un projet de Machine Learning soient nouvelles pour moi, je suis enthousiaste à l'idée de relever ce défi. Mon expérience avec l'analyse des données me donne une bonne base pour aborder ce projet de Data Science, en particulier sur un jeu de données de grande dimension. Je suis convaincu que cette opportunité me permettra d'acquérir de nouvelles compétences précieuses et d'apporter une valeur ajoutée à notre entreprise.

***Sophie Doublier :***

En tant que chercheuse en biologie, la problématique des accidents routiers ne m’est pas familière. Bien qu’habituée à analyser des données, j’apprends pratiquement depuis zéro à mener à bien un projet de Data Science et, plus particulièrement, la méthodologie d’un projet de Machine Learning concernant un jeu de données de grande dimension. Mon objectif est de monter en compétence par l’acquisition de nouvelles méthodes de travail et ainsi d’adapter mon bagage professionnel au monde du travail d’aujourd’hui et de demain. Je n’ai pas connaissance de projet similaire au sein de mon entreprise.

***Stéphane Martinez :***

Chef de projet en informatique, mon expérience professionnelle et mon appétence pour les sujets techniques devraient être un plus pour ce projet. Il n’en demeure pas moins que d’une part, les spécificités autour de la data Science restent nouvelles pour moi et d’autre part, je suis habitué dans mon domaine à répondre à des problématiques clairement définies, ce qui n’est pas le cas ici.

***Marcel Njapo :***

Consultant technique en système d'information plus particulièrement dans le développement d'applications web métier dans le secteur de la finance, Je suis amené à manipuler des chiffres et vérifier la cohérence de ceux-ci. La problématique de la data science sur les accidents de la route bien que nouvelle pour moi, je pense pouvoir apporter mes compétences en développement d’application tout en mettant en pratique les bonnes méthodes de data science que nous voyions actuellement dans notre formation.

## Objectifs du projet

L’objectif de ce projet est d’essayer de prédire la gravité des accidents routiers en France. Les prédictions seront basées sur les données historiques.

L’approche pourra être multiple :

1. Restitution visuelle de certains indicateurs statistiques (évolution des accidents dans le temps, répartition géographique de ces accidents, etc.)
2. Permettre de mesurer l’incidence des différentes caractéristiques sur ces indicateurs afin de pouvoir étudier certaines typologies d’accidents (impliquant telle(s) typologie(s) de véhicule(s), telle(s) condition(s) météorologique(s), telle(s) caractéristique(s) de l’infrastructure, etc.)
3. Prédire la gravité des accidents
4. Identifier un « scoring des zones de danger » : ce genre d’analyse pourrait permettre de mieux cibler :
   1. Les campagnes de prévention routières
   2. Les messages de sécurité routière affichés en temps réel sur nos routes

Nous garderons à l’esprit que l’objectif premier de ce projet est de mettre en pratique le plus de techniques abordées au fil de notre cursus de formation. Pour la suite, nous avons conscience que certaines décisions que nous pourrions prendre ne seraient pas optimales dans un contexte professionnel.

# PARTIE 1 – Consolidation des données

## Méthodologie

Nous avons deux sources de données à notre disposition :

* Site gouvernemental : à l’origine de la collecte et de la mise à disposition des données en lien avec les accidents routiers en France
* Kaggle : elles-mêmes issues des données gouvernementales mais qui ont cet avantage d’être déjà (partiellement) consolidées

En l’état, ces données sont difficilement exploitables. Nous avons dû réaliser différentes opérations afin d’obtenir une donnée mise en forme pour faciliter son analyse et son exploitation pour la suite du projet. Ces opérations ont été réalisées en trois étapes distinctes :

1. Analyser les données à notre disposition pour chacune des sources :
   1. Comprendre leur contenu, identifier un périmètre, commencer à identifier des variations (de structure, de contenu, de règles de gestion)
   2. Identifier les différences entre chacune des sources afin de privilégier celle qui nous permettrait de répondre au mieux à nos objectifs

* Cette étape doit nous permettre de définir la(es) source(s) à privilégier pour la suite.

1. Analyser plus finement chacune des variables afin :
   1. D’en assurer une cohérence afin que chaque variable puisse être exploitée sans risquer d’entraîner de biais.
   2. Dans le cas où cette cohérence ne puisse être assurée, identifier les limites d’utilisation de chacune de ces variables (limitation à prendre en compte pour la suite)

* Cette étape doit nous permettre de nous approprier les données de manière à pouvoir les comprendre et les exploiter au mieux

1. Consolider ces données afin de pouvoir les exploiter au mieux dans des outils d’analyse (exploration de données, réalisation de tests statistiques), de visualisation de données et de traitement par le biais de modèles de prédiction

* Cette étape doit nous permettre de générer un fichier consolidé à la maille souhaitée contenant l’ensemble des informations à notre disposition.

A ce stade, les seules opérations de transformation de données n’auront eu pour but que d’assurer une cohérence globale. Les opérations de préprocessing et/ou de Features engeneering seront réalisées ultérieurement (après cette phase de consolidation des données).

## Analyse des sources de données

### Structure des données disponibles

Les données (via Kaggle ou gouv.fr) sont disponibles sous forme de différents fichiers csv selon 4 typologies d’informations :

* Des données regroupant les caractéristiques liées à l’accident
* Des données en lien avec le lieu de l’accident
* Des données en lien avec les véhicules impliqués
* Des données en lien avec les usagers impliqués

A ces 4 typologies de données, nous pouvons rajouter un fichier récapitulant les périodes de vacances, information qui pourrait être intéressante d’associer à notre étude.

### Les données issues de Kaggle

Les données Kaggle semblent déjà consolidées et proposées sous la forme de 5 fichiers distincts : caracteristics.csv, places.csv, vehicles.csv, users.csv et holidays.csv

Les données sont disponibles de 2005 à 2016.

### Les données issues du gouvernement

Les données du gouvernement sont proposées elles aussi par typologie d’information mais sont consolidées par années. Nous avons donc autant de fichiers que de n-uplets typologie/année d’historique.

Nous devrions avoir un travail de consolidation en apparence plus exigeant sur ces données.

Par contre, l’historique de données du gouvernement va de 2005 à 2022 soit 6 ans de plus que les données Kaggle.

### Macro-Analyse des données de chacune des 2 sources

Nous avons réalisé une visualisation croisée des données afin d’identifier des différences qu’elles soient structurelles, et/ou en termes de contenu (nombre d’enregistrements, valeurs des variables catégorielles, etc.)

Très rapidement, il est apparu quelques écarts entre les deux sources de données que nous ne pouvions pas forcément expliquer :

* Différence de type sur de nombreuses variables
* Quelques différences de contenus (consolidation de données Kaggle)

A l’inverse, les données du gouvernement ont cet avantage d’être :

* Les données originelles : aucun intervenant n’a pu altérer ces informations
* Très bien documentées : un fichier explique l’ensemble des informations (nature de chaque variables, explication des valeurs catégorielles, etc.) ainsi que les évolutions de règles au fil du temps.
* De nouvelles données devraient être publiées pour les années postérieures à 2022.

Nous prenons donc la décision de n’utiliser pour la suite que les données officielles communiquées par le gouvernement.

## Analyse détaillée et Consolidation des données

Conscient que la profondeur d’historique de données à notre disposition est très importante. Nous prenons le parti de toutes les conserver (à cette étape du projet) et ainsi les consolider ensemble de manière à pouvoir ultérieurement décider de se concentrer sur tout ou partie de ces données suite :

* Un écran d’analyse graphique d’indicateurs statistiques sera plus intéressant avec une profondeur d’historique importante
* La modélisation d’une solution ne devait pas nécessiter une profondeur aussi importante
* L’utilisation d’outils de la librairie Scikit-learn nous contraindra à un volume largement plus faible

### Étapes de consolidation des données

Cette consolidation s’est réalisée en deux temps :

* Chaque typologie de données (caractéristiques, lieux, véhicules et usagers) a été consolidée individuellement afin de n’avoir qu’un fichier par typologie contenant l’exhaustivité des données (de 2005 à 2022) :
* Une fois les 4 fichiers consolidés, nous les avons liés entre eux afin de n’avoir plus qu’un fichier contenant l’ensemble des données utiles pour la suite.

Lors de ces étapes de consolidation, nous n’avons pas pris la peine d’estimer l’importance de chacune des informations : nous avons pris le parti de tout conserver tout en respectant une cohérence de chacune des données afin qu’elles puissent être exploitables sans entraîner de biais par la suite.

Nous avons dû prendre des choix (que nous voulions le moins structurant possible pour la suite) :

* Transformation de certaines données.

Ex. certaines données de classification possèdent une valeur « non renseigné » : nous avons dans ce cas pris le parti de remplacer les NaN par cette valeur.

* Formatage de données.

Ex. La variable année de l’accident passe de 2 caractères à 4 à partir de 2017 : décision d’homogénéiser cette information sur 4 caractères.

* Certaines règles de saisie des données ont pu évoluer ce qui pourrait entraîner un biais lors de leur analyse.

Ex. Les règles de la classification de la gravité de l’accident (information d’autant plus importante qu’elle sera utilisée comme variable cible) évolue 2 fois au cours de la période d’historique :

* + En 2018 : évolution du process de saisie d’identification des hospitalisés
  + En 2019 : L’indicateur « Blessé hospitalisé » n’est plus labellisé

Nous verrons par la suite, pour chaque variable concernée, les choix que nous avons décidé de prendre ainsi que les raisons qui les ont motivées.

### Règles d’enrichissement appliquées

***Les identifiants***

Les identifiants (d’accident, de véhicule, d’usager) seront préfixés de id\_xxxxx afin de simplifier leur utilisation future.

***Les variables catégorielles***

Toutes les données catégorielles sont codées en numérique sous la forme :

1 : caractéristique 1

2 : caractéristique 2

…

La plupart ont même une valeur -1=non renseigné

De part ce constat, décision est prise de :

* Passer les valeurs Nan à -1 (Non renseigné) : la gestion des missing value qui sera réalisée ultérieurement devra intégrer cette règle.
* Forcer le type de ces variables en ‘int’.

Ces deux ajustements ont été retenu :

* Afin de pousser l’analyse de la qualité de chaque variable et ainsi de détecter des erreurs de saisie (valeurs non numériques)
* D’avoir une valeur -1 pour les missing values qui pourra assez simplement être visualisée, si on le souhaite, lors de la phase de data visualisation ce qui serait plus délicat en les laissant à NaN.

### Consolidation fichiers « Caractéristiques »

Contient les caractéristiques liées à l’accident

***Structure du fichier brut***

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Nat. | Désignation | Évolutions |
| Num\_acc | Num. | Identifiant unique | Renommage en « id\_accident » |
| An | Num. | Année de l’accident | * Passe sur 4 caractères à partir de 2017 |
| Mois | Num. | Mois de l’accident (1-12) |  |
| Jour | Num. | Jour de l’accident (1-31) |  |
| hrmn | Num. | Heure minute de l’accident | * Retrait du séparateur ‘ :’ qui apparaît à partir de 2019 |
| Lum | Cat. | Luminosité |  |
| Agg | Cat. | Agglo O/N |  |
| Int | Cat. | Type d’intersection | * NaN  -1 =Non renseigné |
| Atm | Cat. | Conditions atmosphériques | * NaN  -1 =Non renseigné |
| Col | Cat. | Type de collision | * NaN  -1 =Non renseigné |
| Com | Cat. | Code INSEE Commune | * Évolution du format à partir de 2019 |
| Adr | Txt | Adresse |  |
| Gps |  | Donnée catégorielle permettant de définir emplacement géographique | * Manquante de 2019 à 2022 |
| Lat | Num. | Latitude | * Manque l’année 2009 complète * Beaucoup de NaN jusqu’en 2018 * Diviser par 100000 jusqu’en 2018 * Utiliser en l’état à partir de 2019 * Renommage en gps\_lat et gps\_long |
| Long | Num. | Longitude |
| Dep | Cat. | Code département | * Évolution du format à partir de 2019 |
|  |  |  |  |

***Évolutions apportées aux données***

***Variable « Num\_acc » :***

Correspond à un identifiant unique associé à chaque accident. Il est présent dans chaque typologie de données et sera renommé en : id\_accident

***Variable « an » :***

L’information évolue de 2 digits jusqu’en 2018 à 4 digits à partir de 2019 : décision de les passer sur 4 digits (arbitrairement) plutôt que sur 2.

***Variable « hrmn » :***

Timestamp de l’accident.

Apparition d’un séparateur ‘:’ entre les heures et les minutes à partir de 2019 : décision de privilégier un format numérique plus pratique à utiliser dans un modèle de ML.

***Variable « Gps » :***

Donnée catégorielle permettant de définir l’emplacement de l’accident (Métropole, Guyane, etc.).

Cette information n’est plus renseignée à partir de 2019, date à partir de laquelle les coordonnées GPS sont correctement renseignées.

Décision de maintenir la colonne. Sa valeur sera forcée à NaN pour les accidents de 2019 et après. Si nous décidons par la suite de conserver cette information, nous pourrons essayer de l’enrichir via des API de gestion de la géolocalisation.

A priori, cette information ne sera pas véritablement exploitée par la suite (à confirmer).

***Variables « Lat » et « Long » :***

Les coordonnées GPS sont manquantes en 2009, jusqu’à 2018 inclus elles sont multipliées par 100000 et utilisable en l’état à partir de 2019.

Décision de diviser les coordonnées antérieures à 2019 pour avoir des coordonnées cohérentes sur toute la période d’historique.

Les coordonnées de 2009 restent à NaN : à voir l’impact ultérieurement si la variable est utilisée.

Le nom de ces deux variables seront préfixés par « gps\_ » afin de simplifier leur compréhension pour la suite.

***Variable « Com » :***

Code INSEE de la Commune : Le format évolue à partir de 2019, décision de ne garder que les 3 derniers digits du code de manière à retomber sur la même nomenclature qu’avant 2019.

Un « code commune » est mal renseigné sur l’année 2022 : les coordonnées GPS étant renseignées, recherche du bon code commune via un site de cartographie et correction de l’anomalie.

Décision de forcer le type en ‘int’ pour faciliter son utilisation via un modèle de ML.

***Variable « Dep » :***

Code INSEE du Département. La règle de codification évolue à partir de 2019 où les départements de Corse passent de 201, 202 à 2A et 2B.

Décision de maintenir la règle antérieure à 2019 en remplaçant les A et B par ’01’ et ‘02’.

Avant 2019, le code département est multiplié par 10 et les 2 départements Corse sont codés en 201 et 202.

A partir de 2019, les codes département repassent sur 2 caractères : les départements Corse passent au format ‘2A’ et ‘2B’.

Les départements d’outre-mer peuvent avoir un code département sur 3 digits.

Décision de privilégier la règle postérieure à 2019 tout en décidant de transcoder les départements Corse en 201 et 202 au lieu de ‘2A’ et ‘2B’ et ce afin de privilégier un format ‘int’ à la variable.

Afin de faciliter la compréhension du sens de cette variable, nous la renommons ‘dept’.

***Merge des fichiers en un seul***

L’ensemble des fichiers bruts issus du site gouv.fr ont été modifiés fichier par fichier en tenant compte des constatations / décisions présentées ci-avant.

Ces transformations ont été réalisées à partir d’un script dédié qu’il serait assez aisé de relancer afin d’apporter des modifications à la structure du fichier consolidé.

*In fine*, l’ensemble des fichiers est mergé de manière à sauvegarder un fichier consolidé contenant l’exhaustivité de l’historique 2005/2022.

Le notebook

«../scripts/0–dataConsolidation/0–Concatenation\_Caracteristiques.ipynb » regroupe l’ensemble de ces règles et permet de générer le fichier « ../dataConcat.gouv/caracteristiques.csv » qui pourra être utilisé à l’étape de merge.

### Consolidation fichiers « Lieux »

Nous allons faire la même analyse sur l’ensemble des fichiers lieux qui contient des informations complémentaires en lien avec le lieu de chaque accident.

La jointure avec les données caractéristiques s’effectuera via l’id\_accident (Num\_acc).

***Structure des fichiers brutes***

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Nat. | Désignation | Évolutions |
| Num\_acc | num | Identifiant unique | Renommage en « id\_accident » |
| catr | Cat. | Catégorie de la route | * Un seul enregistrement en NaN |
| circ | Cat. | Régime de circulation | * Les NaN  -1=Non renseigné |
| env1 | Cat. | Proximité d’une école O/N |  |
| Infra | Cat. | Aménagement & infrastructure |  |
| lartpc | Num. | Largeur du terre-plein | * A typer en float |
| larrout | Num. | Largeur de la chaussée | * A typer en float |
| nbv | Num. | Nombre de voie |  |
| plan | Cat. | Tracé en plan |  |
| pr | Num. | Numéro de point de rattachement (borne amont) | * Contient des données alpha |
| pr1 | Num. | Distance en mètre par rapport à la borne amont | * Contient des données alpha |
| prof | Cat. | Profil déclinité chaussée |  |
| situ | Cat. | Situation de l’accident |  |
| surf | Cat. | État de la surface |  |
| v1 | txt | Indice numérique du numéro de route |  |
| v2 | Txt | Indice alphanumérique de la route |  |
| vma | Num. | Vitesse maximum autorisée |  |
| voie | String | Numéro de voie | * Contient des données alpha |
| vosp | Cat. | Voie réservée |  |
|  |  |  |  |

***Évolutions apportées aux données***

***Variable « catr » :***

Donnée catégorielle définissant la catégorie de la route (autoroute, nationale, départementale).

Un seul enregistrement sans valeur de renseignée : cette valeur a été corrigée via un outil de cartographie et avec l’adresse + coordonnées GPS. La précision des données nous permet d’identifier que la valeur devrait être soit égale à 3=Route départementale, soit 4=Voie communale.

Dans la mesure où il ne s’agit que d’un seul enregistrement, décision de privilégier la valeur 4 qui est la valeur la plus représentée entre les deux (4 -> 571873 occurrences contre 385809 pour la valeur 3).

La variable est forcée en ‘int’.

***Variable « circ » :***

Donnée catégorielle définissant le régime de circulation (sous terrain, Pont, voie-ferrée, aucun, etc.).

Les valeurs NaN sont forcées en -1=non renseigné. Et le type est forcé en ‘int’.

***Variable « vosp » :***

Donnée catégorielle définissant si la voie de l’accident est une voie réservée (sans objet, piste cyclable, etc.).

Les valeurs NaN sont forcées en -1=non renseigné. Et le type est forcé en ‘int’.

***Variable « prof » :***

Donnée catégorielle définissant le profil en log. Elle décrit la déclivité de la route à l'endroit de l'accident (plat, pente, etc.).

Les valeurs NaN sont forcées en -1=non renseigné. Et le type est forcé en ‘int’.

***Variable « plan » :***

Donnée catégorielle définissant le tracé en plan à l'endroit de l'accident (partie rectiligne, en courbe à gauche, à droite, etc.).

Les valeurs NaN sont forcées en -1=non renseigné. Et le type est forcé en ‘int’.

***Variable « surf » :***

Donnée catégorielle définissant l’état de la surface de la voie à l'endroit de l'accident (Normale, mouillée, Flaques, etc.).

Les valeurs NaN sont forcées en -1=non renseigné. Et le type est forcé en ‘int’.

***Variable « infra » :***

Donnée catégorielle définissant le type d’infrastructure à l'endroit de l'accident (aucun, sous terrain, pont, etc.).

Les valeurs NaN sont forcées en -1=non renseigné. Et le type est forcé en ‘int’.

***Variable « situ » :***

Donnée catégorielle définissant la situation de l’accident (sur la chaussée, sur bande d’arrêt d’urgence, sur trottoir, sur piste cyclable).

Les valeurs NaN sont forcées en -1=non renseigné. Et le type est forcé en ‘int’.

***Variable « vma » :***

Donnée quantitative identifiant la vitesse maximale autorisée à l'endroit de l'accident.

Cette donnée apparait qu’à partir de 2019. La valeur est forcée à NaN pour les années antérieures.

***Variable « env1 » :***

Donnée catégorielle binaire permettant de définir la présence d’une école à l'endroit de l'accident.

Cette donnée disparait à partir de 2019. La valeur est forcée à NaN pour les années postérieures à 2018.

***Variable « nbv » :***

Donnée quantitative permettant de définir le nombre de voie à l’endroit de l’accident.

Les valeurs NaN sont forcées en -1=non renseigné. Et le type est forcé en ‘int’.

***Variable « lartpc » :***

Donnée quantitative permettant de définir la largeur du terre-plein central (en mètre).

Nous détectons un changement de règle de saisie sur cette variable qui est renseignée à un taux de plus de 96% jusqu’en 2017 puis chute à 23% en 2018 pour passer en dessous des 1% à partir de 2019.

Si ce changement de règle n’est pas documenté, il semblerait que la principale différence entre avant 2018 et après ne concerne que les valeurs à 0. Les données semblent dire qu’en cours d’année 2018, l’absence de terre-plein n’est plus saisie avec la valeur 0 comme c’était le cas précédemment.

La modalité étant largement la plus utilisée reste la valeur 0=pas de terre-plein ce qui semble logique.

Décision de corriger cette différence en forçant à 0 toutes les valeurs manquantes. La variable sera forcée

La variable est forcée en ‘float’ afin de privilégier son utilisation possible dans un modèle de ML.

***Variable « larrout » :***

Donnée quantitative permettant de définir la largeur de la chaussée (en mètre).

La variable est forcée en ‘float’ afin de privilégier son utilisation possible dans un modèle de ML.

***Variables « voie », « pr » et « pr1 » :***

Données textuelles contenant respectivement le numéro de la voie, le point de rattachement (numéro de la borne amont) et la distance à cette même borne.

Les variables sont forcées en ‘str’ car elles semblent très hétérogènes en termes de contenu. Une analyse approfondie pourra être réalisée ultérieurement.

***Merge des fichiers en un seul***

Selon la même méthodologie que pour le fichier « caractéristiques », un fichier lieux est constitué à partir des constatations précédemment citées.

Le notebook

«../scripts/0–dataConsolidation/0–Concatenation\_Lieux.ipynb » regroupe l’ensemble de ces règles et permet de générer le fichier « ../dataConcat.gouv/lieux.csv » qui pourra être utilisé à l’étape de merge.

### Consolidation fichiers « Vehicules »

Même analyse sur l’ensemble des fichiers contenant le détail des véhicules.

La jointure avec les données caractéristiques et lieux s’effectuera également via l’id\_accident (Num\_acc).

***Structure des fichiers brutes***

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Nat. | Désignation | Évolutions |
| Num\_acc | num | Identifiant unique | Renommage en « id\_accident » |
| Id\_vehicule | num | Identifiant de chaque véhicule |  |
| catv | Cat. | Catégorie du véhicule |  |
| choc | Cat. | Point de choc initial |  |
| manv | Cat. | Manœuvre principale avant l’accident |  |
| motor | Cat. | Type de motorisation | * L’information apparait en 2019 |
| obs | Cat. | Obstacle fixe heurté | * Ajout en 2017 de -1=non renseigné mais surtout de la valeur 17=’buse tête d’aqueduc’ |
| obsm | Cat. | Obstacle mobile heurté | * Valeur NaN  -1 |
| occutc | Num. | Nombre d’occupants dans le transport en commun | * Chgt de règle de gestion à partir de 2019 : si cette variable ne concerne que les véhicules de trasnport, ce n’est qu’en 2019 que cette info n’est renseignée que pour cette catégorie. * Avant 2019, nous retrouvons des 0 pour tous les véhicules non catégorisés transport en commun |
| sensc | Cat. | Sens de circulation | * La catégorie -1 bien que représentée n’est pas documentée. Par contre, il existe une catégorie 0 =inconnu * Valeur NaN et -1  0 |
|  |  |  |  |

***Évolutions apportées aux données***

***Variables « id\_vehicule » et « num\_veh » :***

Cet identifiant « id\_vehicule » apparait en 2019 (dans véhicule et usager). Son apparition est due au changement de règle sur les piétons qui, à partir de 2019 se retrouvent associés à un véhicule.

Cet identifiant n’est pas suffisant pour assurer la jointure avec les usagers : il devra être utilisé avec l’information « num\_veh ».

Afin de permettre cette jointure pour l’ensemble du périmètre, l’information « id\_vehicule » sera initialisée à 0.

Le contrôle d’absence de doublon sur les véhicules devra se faire à la maille magasin car les « num\_vehi » ne sont pas uniques

Suppression du caractère '\xa0’ qui empêche le typage en ‘int’ de la variable.

***Variable « senc » :***

Donnée catégorielle permettant de définir le sens de circulation du véhicule. 3 valeurs possibles sur l’ensemble des années d’historique :

* 0 : inconnu
* 1 : PK ou PR ou numéro d’adresse postale croissant
* 2 : PK ou PR ou numéro d’adresse postale décroissant

Une majorité de 0 = inconnu jusqu’en 2015 inclus. A partir de 2016, nous pouvons constater que les règles de saisie ont évoluées (l’occurrence 0 qui était largement majoritaire jusqu’en 2015 devient minoritaire à partir de 2016).

A ces trois valeurs, sont ajoutées à partir de 2019 :

* 3 : Absence de repère
* -1 : non renseignée

Nous pouvons également constater quelques NaN entre 2016 et 2018.

Afin d’apporter une cohérence sur l’ensemble de l’historique des données, nous prenons la décision de forcer à 0=inconnu les valeurs -1 et NaN.

***Variable « catv » :***

Donnée catégorielle permettant de définir la catégorie du véhicule. Nous constatons une évolution des catégories au fil des années : des catégories disparaissent à partir de 2006 au profit d’autres valeurs.

Nous avons pu reconstituer cette table de correspondance :

|  |  |
| --- | --- |
| Codes inutilisés depuis 2006 | Codes privilégiés depuis 2006 |
| * 04= scooter immatriculé | * 32= Scooter > 50 cm3 et <= 125 cm3 * 34=Scooter > 125cm3 |
| * 05= motocyclette | * 31= motocyclette > 50cm3 et < 125cm3 * 33= motocyclette > 125cm3 |
| * 06= side-car | * 33= motocyclette > 125cm3 |
| * 08= VL + caravane | * 07= VL seul |
| * 09= VL + remorque | * 07= VL seul |
| * 11= VU(10) + caravane | * 10=VU seul 1,5T <= PTAC <= 3,5T avec ou sans remorque |
| * 12= VU(10) + remorque | * 10=VU seul 1,5T <= PTAC <= 3,5T avec ou sans remorque |
| * 18= transport en commun | * 37=Autobus * 38=Autocar |
| * 19= tramway | * 40 = Tramway |
|  |  |

Remarque : bien que ces codes aient été remplacés en 2006, nous en retrouvons jusqu’en 2008.

Décision de privilégier les nouveaux codes aux codes obsolètes. Si l’opération de substitution est simple pour les codes remplacés par un seul autre, la situation est différente pour ceux associés à 2 codes distincts. Nous avons pris le parti de réaliser cette correction de manière aléatoire tout en préservant la distribution initiale. Cette opération a été réalisée fichier par fichier donc par année afin de garder la distribution de chacune des années pour chacun des codes.

***Variable « obs » :***

Donnée catégorielle permettant de définir le type d’obstacle fixe heurté par le véhicule.

Nous constatons l’apparition de deux occurrences à partir de 2019 :

* -1=non renseigné
* 17=Buse, tête d’aqueduc

/ !\ L’absence du code ‘17’ avant l’année 2019 devra peut-être être pris en compte afin d’éviter d’ajouter un biais à l’analyse. Cela représente au total 414 enregistrements (véhicules) sur 2 009 395 véhicules au total.

Décision de basculer les NaN  -1 apparu en 2019 et de forcer le type en ‘int’.

***Variable « obsm » :***

Donnée catégorielle permettant de définir le type d’obstacle mobile heurté par le véhicule.

Décision de basculer les NaN  -1 apparu en 2019 et de forcer le type en ‘int’.

***Variable « choc » :***

Donnée catégorielle permettant de définir le point de choc initial.

Décision de basculer les NaN  -1 apparu en 2019 et de forcer le type en ‘int’.

***Variable « manv » :***

Donnée catégorielle permettant de définir la manœuvre principale au moment de l’impact.

Apparition de 3 catégories à partir de 2019 :

* -1 : non renseigné
* 25 : circulant sur trottoir (lié au fait qu’à partir de 2019, les trotinettes et roller sont considérés comme des véhicules)
* 26 = autre manœuvre

/!\ L’ajout de ces codes devra être pris en compte dès lors que l’historique des données utilisées remontent par la suite à une période antérieure à 2019. Si l’utilisation du code ‘25’ reste marginal (1040 au total), le code 26 est quant à lui plus représenté (12678 occurrences).

Décision de basculer les NaN  -1 et forcer le type en ‘int’.

***Variable « motor » :***

Donnée catégorielle permettant de définir le type de motorisation du véhicule.

Elle apparait qu’en 2019. Dans l’hypothèse d’une utilisation antérieure à 2019, cette variable devra être écartée.

Afin de permettre la fusion des données d’historique, la valeur NaN sera forcée pour cette variable pour les données antérieures à 2019.

***Variable « occutc » :***

Donnée quantitative correspondant au nombre de passagers dans les véhicules de transport en commun.

Jusqu’à 2018 inclus, tous les véhicules ont une information renseignée dans cette variable.

A partir de 2019, seuls les véhicules catégorisés « transport en commun (cat in [18, 19, 37, 38, 39, 40]) ont une valeur de renseignée (ce qui est cohérent par rapport à la documentation).

Nous constatons également quelques erreurs sur les fichiers de 2005 à 2008 avec respectivement (1, 58, 12 et 14) erreurs. Des véhicules non catégorisés véhicule de transport se retrouvent avec une valeur différente de 0. Un comptage du nombre d’usagers par véhicule a permis de confirmer l’erreur et de repositionner ces valeurs à 0.

Pour maintenir la cohérence des données, décision est prise de passer toutes les NaN  0 et la colonne est forcée en ‘int’.

***Merge des fichiers en un seul***

Selon la même méthodologie que pour les fichiers précédemment traités, un fichier vehi.csv est constitué à partir des constatations précédemment citées.

Le notebook

«../scripts/0–dataConsolidation/0–Concatenation\_Vehicules.ipynb » regroupe l’ensemble de ces règles et permet de générer le fichier « ../dataConcat.gouv/vehicules.csv » qui pourra être utilisé à l’étape de merge.

### Consolidation fichiers « usagers »

Même analyse sur l’ensemble des fichiers contenant le détail des usagers.

La jointure avec les données caractéristiques et lieux s’effectuera via « id\_accident » (Num\_acc).

La jointure avec les données « véhicules » s’effectuera à partir de « id\_accident », de « id\_vehicule » et du « Num\_vehi ».

***Structure des fichiers brutes***

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Nat. | Désignation | Évolutions |
| Num\_acc | num | Identifiant unique | Renommage en « id\_accident » |
| Id\_vehicule | num | Identifiant de chaque véhicule | Clé de rapprochement avec le véhicule |
| num\_veh | Txt | Référence alphanumérique | Clé de rapprochement avec le véhicule |
| Id\_usager | Num | Identifiant unique | Unicité assurée avec les autres id |
| An\_naiss | Num | Année de naissance |  |
| catu | Cat. | Catégorie d’usager | * Changement de règle des usagers de trottinette en 2018 |
| actp | Cat. | Action du piéton | * 2 nouvelles modalités ‘A’ et ‘B’ apparues en 2019 |
| etatp | Cat. | Piéton seul ou non |  |
| locp | Cat. | Localisation du piéton |  |
| place | Cat. | Emplacement dans le véhicule |  |
| secu1 | Cat. | Équipements de sécurité | * Évolution majeure de la saisie de ces informations à partir de 2019 |
| secu2 | Cat. |
| secu3 | Cat. |
| sexe | Cat. | Genre de l’usager |  |
| trajet | Cat. | Catégorie de trajet |  |
| grav | Cat. | Gravité de l’accident | * Évolution des règles de répartition en 2018 qui interdit une comparaison entre les données antérieures à 2018 et celles à partir de 2018. * La catégorisation n’est pas ordonnée |
|  |  |  |  |

***Évolutions apportées aux données***

***Variables « id\_vehicule » et « num\_veh » :***

Même constat que dans le fichier vehicule (cf.véhicule). Nous appliquons les mêmes actions.

***Variable « id\_usager » :***

Apparition en 2021 : son utilisation doit permettre de vérifier l’unicité des usagers. Dans notre analyse, cette information n’apportera rien.

La variable est maintenue dans la phase de consolidation mais sera très vraisemblablement supprimée ultérieurement.

La variable est initialisée à 0 dans les années antérieures à 2021.

***Variable « catu» :***

Donnée catégorielle permettant de définir l’usager (Conducteur, passager, Piéton, etc.).

Changement important en 2018 sur la comptabilisation des usagers en Roller et Trottinettes comptabilisé comme piéton jusqu’en 2018 (catégorie 4) alors que l’information se déplace dans la donnée véhicule (catv =99).

/ !\ Da part sa complexité, ce changement de règle n’est pas pris en compte à ce stade. Par la suite, si les données antérieures à 2018 sont utilisées, ce point devra être étudié.

***Variable « place» :***

Donnée catégorielle permettant de définir l’emplacement de la personne dans le véhicule de rattachement.

Décision de basculer les NaN  -1 apparu en 2019 et de forcer le type en ‘int’.

***Variables « secuX » :***

Ces trois variables catégorielles permettent de définir les équipements de sécurité utilisés au moment de l’accident.

Les deux variables secu2 et 3 sont venu compléter la variable secu1 en 2019. Le contenu de secu1 change à partir de cette date.

Nous avons pu reconstituer cette table de correspondance :

|  |  |
| --- | --- |
| Secu1 avant 2019 | Correspondance secu1, 2 et 3 à partir de 2019 |
| Secu1 composé de 2 digits car1 et car2  Car1 : 1=Ceinture, 2=Casque, 3=Dispositif enfant, 4=Équipement réfléchissant, 9=Autre  Car2 : défini si l’équipement était utilisé  1=oui, 2=non, 3=non déterminable | Secu1, 2 ou 3 références les équipements utilisés lors de l’accident :  -1– Non renseigné 0 – Aucun équipement 1 – Ceinture  2 – Casque 3 – Dispositif enfants 4 – Gilet réfléchissant 5 – Airbag (2RM/3RM) 6 – Gants (2RM/3RM) 7 – Gants + Airbag (2RM/3RM) 8 – Non déterminable 9 – Autre |

Nous avons décidé d’harmoniser ces informations en implémentant l’algorithme suivant :

Si car2 == 2 --> secu1=0

Si car2 == 3 --> secu1=8

Si car2 == 1 --> secu1=car1

Sinon secu1 --> 0

Cela permet d’avoir une information homogène pour secu1. Il ne sera pas possible de reconstituer secu2 et 3 (information manquante) :

/ !\ Il faut donc garder en tête que secu2 et 3 ne sont disponibles qu’à partir de 2019

***Variable « trajet» :***

Donnée catégorielle permettant de définir le type de trajet. Deux modalités sont associées à la même notion -1 et 0 = non défini

Décision de basculer les NaN ainsi que les valeurs 0  -1 (afin d’être cohérent avec les autre variables) et de forcer le type en ‘int’.

***Variable « locp» :***

Donnée catégorielle permettant de définir la localisation du piéton au moment de l’accident.

De nombreuse données en missing value concernent des usagers qui ne sont pas identifiés comme piéton.

Décision de corriger les NaN et les -1 des non piétons (catu<>3) 0=Sans Objet

Il existe une catégorie 9=inconnue qui pourrait faire double emploi avec -1

Décision de laisser en l’état, la décision de conserver ou non ces catégories sera à challenger ultérieurement.

***Variable « actp» :***

Donnée catégorielle permettant de définir l’action du piéton au moment de l’accident.

Quelques enregistrements à nettoyer (des espaces en trop dans certaines valeurs) et correction des NaN ou -1 associés à des usagers non piétons que l’on force à 0=non renseigné ou sans objet.

Nous avons également 2 modalités similaire que nous décidons de laisser en l’état pour l’instant : -1=non renseigné et 0=non renseigné ou sans objet.

***Variable « etatp» :***

Donnée catégorielle permettant de définir si le piéton accidenté était seul ou non.

De nombreuse données en missing value concernent des usagers qui ne sont pas identifiés comme piéton.

Décision de corriger les NaN et les -1 des non piétons (catu<>3) 0=Sans Objet

***Variable « grav» :***

Donnée catégorielle permettant de définir le niveau de gravité de l’accident. Cette variable est d’autant plus importante qu’elle servira fort probablement comme variable cible.

4 modalités disponibles pour cette variable :

* + - 1 = Indemne
    - 2 = Tué
    - 3 = Blessé hospitalisé
    - 4 = Blessé léger

Plusieurs évolutions sont à prendre en compte :

* Les données sur la qualification de « blessé hospitalisé » depuis 2018 ne peuvent être comparés aux années précédentes suite à des modifications de process de saisie des forces de l’ordre
* L’indicateur « Blessé hospitalisé » n’est plus labellisé par les autorités de la statistique depuis 2019 : il devient légitime de s’interroger sur la pertinence de conserver cette information (l’absence de labélisation est-elle due à une fiabilité insuffisante ?)
* A partir de 2021, les usagers en fuite ont été rajoutés, cela entraîne des manques d’informations sur ces derniers (valeurs manquantes possibles)

*Décisions :*

1. *Malgré des recherches réalisées quant à la fiabilité de l’information « blessé hospitalisé », nous n’avons pu obtenir plus d’information permettant de décider de conserver ou non cette catégorie.*

* *Décision est prise de la conserver : nous pourrons éventuellement réfléchir ultérieurement à regrouper « blessé léger » et « blessé hospitalisé » en une seule catégorie.*

1. Il sera impossible d’utiliser cette variable comme variable cible avant 2018.
2. Nous serons amenés par la suite à supprimer des informations en lien avec les usagers en missing value sur cette variable. En l’état, les enregistrements sont bien évidemment conservés.
3. Peut-être inutile mais nous décidons de transcoder les valeurs de manière à les ordonner par ordre de gravité :

* 0 continue de correspondre à indemne
* 1 correspondra aux blessés légers
* 2 correspondra aux blessés graves (hospitalisés)
* 3 correspondra aux tués

Afin d’éviter des erreurs d’interprétation, nous décidons de renommer la variable en « gravite ». La variable grav sera supprimée.

***Merge des fichiers en un seul***

Le notebook

«../scripts/0–dataConsolidation/0–Concatenation\_Usagers.ipynb » regroupe l’ensemble de ces règles et permet de générer le fichier « ../dataConcat.gouv/usagers.csv » qui pourra être utilisé à l’étape de merge.

## Merge global des données

Une fois l’analyse de chaque typologie de données (caractéristiques, lieux, véhicules et usagers) réalisée, nous nous retrouvons avec un fichier par type de données contenant l’exhaustivité de l’historique.

Maintenant, nous allons consolider ces 4 fichiers entre eux. Pour ce faire,

1. Nous allons réaliser quelques contrôles préliminaires (volumétrie des données, absence de doublons, complétude des données) dans chacun des fichiers

2. Merger les fichiers entre eux

3. Enrichir l’ensemble de données agrégées

4. Appliquer une règle de nommage afin de faciliter l’utilisation futur du fichier

Ces étapes seront facilement ré-exécutable dans la mesure où nous les avons regroupées dans un script unique : «../scripts/0–dataConsolidation/0–Merge\_final.ipynb » .

### Contrôles préliminaires des données

***Fichier « caractéristiques » :***

Nous partons du postulat que « id\_accident » doit être renseigné et unique. Nous considérerons deux caractéristiques partageant un même « id\_accident » comme étant un doublon (situation anormale). Nous pourrions pousser ce contrôle en recherchant des accidents ayant les mêmes caractéristiques mais avec des « id\_accident » différents afin de les considérer également comme des doublons.

La complexité d’une telle analyse, les questions qu’elles risquent d’engendrer quant au caractère normal ou non de chaque situation ramenée, d’une part, au volume des données (plus d’un million de caractéristiques sur la période) et d’autre part au niveau exceptionnel de qualité des données que nous avons pu constater jusque-là, nous avons jugé inutile de réaliser ce contrôle : seuls les enregistrements avec un même « id\_accident » seront considérés comme des doublons.

Les doublons identifiés devront être supprimés de la base avant l’opération de merge.

Sur cette base, nous confirmons que sur les 1.176.873 enregistrements (donc accidents), tous ont un « id\_accident » de renseigné et aucun n’est commun.

***Fichier « lieux » :***

Nous appliquons le même contrôle que pour « caractéristiques » sur le fichier lieux : seul la recherche de doublon de « id\_accident » sera considéré comme un accident ce qui d’ailleurs est logique car nous pourrions avoir des lieux très accidentogènes qui pourraient revenir plusieurs fois dans les données (associés à des « id\_accident » différents).

Nous devrions avoir autant d’enregistrement dans le fichier lieux que de caractéristiques car il existe un lien un pour un entre caractéristique et lieux.

Sur cette base nous confirmons que sur les 1.176.873 enregistrements attendus (autant que de caractéristiques), nous retrouvons exactement ce nombre de lieux, que chaque lieu possède un « id\_accident » et que ces « id\_accident » sont commun avec ceux du fichier « caractéristiques ».

***Fichier « véhicules » :***

Nous allons ici vérifier l’absence de doublons non plus uniquement sur « id\_accident » car il peut arriver que plusieurs véhicules soient impliqués dans un même accident mais sur le n-uplet « id\_accident », « id\_vehicule » et « num\_veh ».

La complétude des données devra également être vérifiée : nous devrions retrouver les mêmes « id\_accident » que ceux définis dans « caractéristiques » et « lieux ».

Là encore, la qualité du jeu de données est confirmée car aucun doublon ni perte d’information n’est identifiée. Ce sont 2.009.395 véhicules qui sont ainsi identifiés ce qui représente un ratio de 1,7 véhicules par accident.

***Fichier « usagers » :***

Pour clore ces contrôles, nous réalisons les mêmes sur le dernier fichier « usagers ». Les piétons étant associés au véhicule qui les a percutés, tous les usagers devraient être associés à un véhicule défini dans le fichier « véhicules ».

Les contrôles confirment une nouvelle fois la qualité des données. Ce sont 2.636.377 usagers qui sont ainsi référencés et tous associés à un véhicule (/accident) défini dans le fichier « véhicules ».

### Merge du fichier caractéristiques avec le fichier lieux

Le merge des deux fichiers est réalisé en « inner join » via la clé « id\_accident ». Nous confirmons la relation un/un entre les deux fichiers : le DataFrame résultant se retrouve composé du bon nombre d’enregistrements à savoir 1.176.873.

### Merge avec fichier véhicules

Le fichier précédemment obtenu est maintenant mergé avec le fichier « véhicules », toujours en « inner join » sur la clé « id\_accident » afin de confirmer la complétude des données contrôlée à l’étape précédente.

Le fichier résultant se retrouve, tel qu’attendu, composé de 2.009.395 enregistrements

### Merge avec fichier usagers

Le dernier merge en « inner join » mais sur les trois clés « id\_accident », « id\_vehicule » et « num\_veh » nous permet d’obtenir un fichier unique contenant l’exhaustivité des variables mises à disposition du gouvernement sur une période courant de janvier 2005 à décembre 2022.

Ce fichier contient au total 2.636.377 soit autant que d’usagers ce qui est conforme à l’attendu.

Remarque : Dans la mesure où la notion « id\_vehicule » n’est apparue qu’à partir de 2019, pour que le merge fonctionne, il est impératif de forcer la valeur de « id\_vehicule » à une constante non NaN à la fois dans le fichier « véhicules » mais aussi dans « usagers » ce que nous avions fait lors de l’étape de consolidation des données.

### Agrégation des données

A cette étape nous avons un fichier contenant le détail de chaque usager concerné par un accident routier survenu sur le territoire Français entre 2005 et 2022.

Chaque enregistrement (donc usager) est composé de ses caractéristiques en tant qu’usager (catégorie, emplacement dans le véhicule, gravité de l’accident en ce qui le concerne, etc. auxquelles on retrouve des informations associées à son véhicule de rattachement ainsi qu’à l’accident de rattachement.

Cela représente un nombre assez considérable d’information. C’est d’ailleurs ce qui semble caractériser ce projet à savoir la richesse des informations disponibles tant en profondeur (d’historique) qu’en nombre de variables explicatives.

Nous pourrions légitimement à ce stade du projet considérer avoir suffisamment (voir trop) de données à disposition.

Ceci étant, peut-être serait-il intéressant de faire ressortir de nouvelles données à partir de celles déjà disponible :

* Soit en jouant sur la granularité de l’information
* Et/ou en consolidant ces données avec des données agrégées

Deux approches complémentaires :

* En restant à la granularité « usagers » que nous avons déjà, peut-être qu’il serait intéressant d’enrichir chaque enregistrement d’information portée par d’autres enregistrements comme :
  + Le nombre de personnes dans le véhicule de l’usager
  + Le nombre de personnes concernées par l’accident
  + Le nombre de personnes par catégorie (piétons, conducteur, passager)
  + Le nombre de personnes de chaque catégorie de gravité
  + Le nombre de catégorie de véhicules concerné (nombre de vélos, de PL, etc.

Ceci est tout à fait possible par agrégation des données à la maille « id\_vehicule » puis à la maille « id\_accident ». Une restitution des données pourrait être faite :

* + Soit en restant sur une granularité usager
  + Soit en privilégiant une granularité accident
* En privilégiant une dimension nouvelle qui serait la maille « accident » plutôt que « usager ».

Décision est prise de réaliser ces agrégations et de sauvegarder *in fine* non pas un mais deux fichiers de données tous deux enrichis des données agrégées :

* Le premier à la maille usager : « ../dataConsolidees.gouv/agg\_usagers.csv »
* Le second à la maille accident : « ../dataConsolidees.gouv/agg\_accidents.csv »

Nous pourrons par la suite, privilégier l’un ou l’autre des fichiers à disposition.

Le regroupement de catégorisations des véhicules pourrait être intéressant dans la suite de l’analyse. Nous décidons de garder la variable.

Problème. En l’état, les données identifient une quarantaine de catégories différentes ce qui semble bien trop important pour l’usage que nous souhaitons en faire.

Décision de rajouter une variable « regroupement de catégories de véhicules » qui sera privilégiée pour les données que nous souhaitons agréger. Voici le regroupement retenu :

* 1 = Vélo
* 2 = Véhicule sans permis
* 3 = Motocyclette
* 4 = Véhicule léger
* 5 = Poids lourd
* 6= Autres

***Information d’agrégation calculées (agg\_catv\_perso)***

Particularité de ces données : afin de les identifier rapidement comme tel, chaque nom de données agrégées sera préfixé par agg\_

Nous avons ainsi rajouté aux données déjà disponibles :

* agg\_nb\_total\_vehicule
* agg\_nb\_total\_velo
* agg\_nb\_total\_vsp
* agg\_nb\_total\_moto
* agg\_nb\_total\_vl
* agg\_nb\_total\_pl
* agg\_nb\_total\_va
* agg\_nb\_total\_conducteur
* agg\_nb\_total\_conductrice
* agg\_nb\_total\_pieton
* agg\_nb\_total\_passager
* agg\_nb\_total\_indemne
* agg\_nb\_total\_blesse\_leger
* agg\_nb\_total\_blesse\_grave
* agg\_nb\_total\_tue
* agg\_nb\_pieton\_vehicule
* agg\_nb\_passager\_vehicule
* agg\_nb\_indemne\_vehicule
* agg\_nb\_blesse\_leger\_vehicule
* agg\_nb\_blesse\_grave\_vehicule
* agg\_nb\_tue\_vehicule

Par convention (simplification d’usage), les accents ainsi que la forme plurielle ne seront pas retenus.

Ces données agrégées étant le fruit d’un calcul sur des données déjà disponible, nous devrions identifier par la suite une corrélation plus ou moins importantes entre elles. Nous devons rester vigilant au risque d’engendrer des problèmes de multi colinéarité : nous aurons toujours la possibilité de privilégier l’une ou l’autre des données disponibles pour la suite du projet.

### Renommage de chacune des variables

Avant de sauvegarder chacun des deux fichiers et afin d’en simplifier leur utilisation future, nous décidons de préfixer chaque variable issue d’un des 4 typologies de données :

* carac\_ : pour les variables contenues dans caractéristiques
* lieu\_ : pour les variables issues dans lieux
* vehi\_ : pour les variables issues de véhicules
* user : pour les variables issues de users
* Les variables agrégées étant déjà préfixées de agg\_

## Synthèse Consolidation des données

***Voici les caractéristiques du fichier ainsi obtenu :***

RangeIndex: 2636355 entries, 0 to 2636354

Data columns (total 81 columns):

# Column Dtype

--- ------ -----

0 id\_accident int64

1 id\_vehicule int64

2 num\_veh object

3 id\_usager object

4 carac\_an int64

5 carac\_mois int64

6 carac\_jour int64

7 carac\_hrmn int64

8 carac\_adr object

9 carac\_agg int64

10 carac\_atm int64

11 carac\_col int64

12 carac\_com float64

13 carac\_dept int64

14 carac\_gps object

15 carac\_gps\_lat float64

16 carac\_gps\_long float64

17 carac\_int int64

18 carac\_lum int64

19 lieu\_catr int64

20 lieu\_circ int64

21 lieu\_env1 float64

22 lieu\_infra int64

23 lieu\_larrout float64

24 lieu\_lartpc float64

25 lieu\_nbv int64

26 lieu\_plan int64

27 lieu\_pr object

28 lieu\_pr1 object

29 lieu\_prof int64

30 lieu\_situ int64

31 lieu\_surf int64

32 lieu\_v1 float64

33 lieu\_v2 object

34 lieu\_vma float64

35 lieu\_voie object

36 lieu\_vosp int64

37 vehi\_catv int64

38 vehi\_choc int64

39 vehi\_manv int64

40 vehi\_motor float64

41 vehi\_obs int64

42 vehi\_obsm int64

43 vehi\_occutc int64

44 vehi\_senc float64

45 user\_an\_nais float64

46 user\_catu int64

47 user\_actp object

48 user\_etatp int64

49 user\_locp int64

50 user\_place int64

51 user\_secu1 int64

52 user\_secu2 int64

53 user\_secu3 int64

54 user\_sexe int64

55 user\_trajet int64

56 user\_gravite int64

57 agg\_catv\_perso int64

58 agg\_is\_conducteur\_vehicule int64

59 agg\_is\_conductrice\_vehicule int64

60 agg\_nb\_pieton\_vehicule int64

61 agg\_nb\_passager\_vehicule int64

62 agg\_nb\_indemne\_vehicule int64

63 agg\_nb\_blesse\_leger\_vehicule int64

64 agg\_nb\_blesse\_grave\_vehicule int64

65 agg\_nb\_tue\_vehicule int64

67 agg\_nb\_total\_velo int64

68 agg\_nb\_total\_vsp int64

69 agg\_nb\_total\_moto int64

70 agg\_nb\_total\_vl int64

71 agg\_nb\_total\_pl int64

72 agg\_nb\_total\_va int64

73 agg\_nb\_total\_conducteur int64

74 agg\_nb\_total\_conductrice int64

75 agg\_nb\_total\_pieton int64

76 agg\_nb\_total\_passager int64

77 agg\_nb\_total\_indemne int64

78 agg\_nb\_total\_blesse\_leger int64

79 agg\_nb\_total\_blesse\_grave int64

80 agg\_nb\_total\_tue int64

dtypes: float64(11), int64(61), object(9)

# PARTIE 2 – Analyse exploratoire des données

## Méthodologie

1. Sélection du périmètre de données éligible pour la suite
2. Contrôles préalables des données (visualisation globale de nos données)
3. Séparation de la base et technique d’évaluation du modèle
4. Regroupement des variables par types (quantitatives, qualitatives (non) binaires)
5. Analyse unitaire des variables catégorielles
6. Analyse unitaire des variables quantitatives
7. Matrice de corrélation

Cette étape doit nous permettre d’identifier les transformations nécessaires de nos données pour être exploitées dans un modèle de ML.

Nous baserons nos décisions sur des tests statistiques afin d’estimer le niveau d’adhérence de chaque variable avec notre variable cible.

* Un test de chi2 + Cramer pour les données catégorielles
* Un test Anova + coefficient de corrélation pour les données quantitatives

Ces tests seront exécutés :

* Sur les variables après remplacement des valeurs manquantes par la catégorie ‘-1’
* Sur les variables après remplacement des valeurs manquantes par la modalité la plus fréquente.

L’analyse de ces résultats statistiques nous permettra de décider pour chaque variable :

* De l’exclure du scope des features (p-value > 0.05 ou indice de Cramer/coef de corrélation trop faible)
* De définir la règle de remplacement des valeurs manquantes : sera privilégiée la méthode qui apporte une adhérence maximisée

Nous profiterons de cette analyse pour étudier plus en détail la nature de chaque variable afin de définir :

* La règle à privilégier pour le traitement des valeurs aberrantes
* La nécessité de prévoir un encodage de la variable (catégorielle non ordonnée)
* La nécessité de normaliser/standardiser la variable (quantitative)

L’objectif étant de définir le périmètre des transformations à prévoir pour obtenir des données compatibles avec un modèle de ML.

Pour atteindre cet objectif, nous allons devoir définir des seuils à partir desquels la décision sera prise de conserver ou au contraire d’exclure une variable du périmètre des Features. N’ayant pas d’expérience de ces sujets, notre choix sera motivé principalement sur du ressenti (décision au « doigt mouillé »).

Il faut donc considérer ces décisions de transformations comme étant des décisions « à priori » susceptibles d’évoluer en fonction des constatations et des contraintes auxquelles nous devrions nous heurter par la suite.

Afin d’en atténuer au maximum les effets, ces transformations seront réalisées dans des fonctions spécialisées et paramétrables, compatibles avec une exécution depuis un pipeline afin de limiter ses ajustements à de simples ajustements d’arguments passés à nos fonctions.

L’avantage d’une telle approche étant de nous permettre de profiter d’un GridSearch afin d’identifier les transformations les plus efficaces.

## Analyse préliminaire

### Périmètre de données retenu

A ce stade, nous allons pouvoir explorer les données à notre disposition. Cette phase sera d’autant plus importante qu’une des finalités reste l’implémentation d’un modèle de prédiction.

Si l’exhaustivité des données à notre disposition fait sens dans l’utilisation d’un outil de visualisation de différents indicateurs (partie plus analytique du projet), il ne semble pas opportun de garder ce volume notamment si l’on privilégie des outils de la librairie de scikit-learn.

Un tel volume pourrait être intéressant dans des solutions de deep learning et avec des outils tel que TensorFlow.

Nous décidons donc de challenger le périmètre retenu pour la partie exploration des données.

***Contraintes d’usage des données identifiées :***

* La gravité de l’accident à partir de 2018 ne peut être comparée avec les années antérieures.
* Certaines variables n’apparaissent qu’à partir de 2019 :
  + motor : type de motorisation du véhicule
* A contrario certaines variables disparaissent à partir de 2019 :
  + gps : double emploi avec les coordonnées gps
  + env1 : proximité d’une école
* Beaucoup de coordonnées gps manquantes en 2018, aucune à partir de 2019.
* De nouvelles modalités de catégorisation apparaissent en 2019 :
  + obj : Tête de buse, aqueduc
  + manv : Manœuvre avant accident
  + actp : action du piéton avant impact
* Le référencement des trottinettes et rollers évolue significativement à partir de 2019 : cette catégorie se retrouve comptabilisée comme véhicule et non plus comme catégorie d’usagers.

***Volumétrie (cumulée) des données disponibles :***

* A partir de 2018 : 624351
* A partir de 2019 : 494182
* A partir de 2020 : 361205
* A partir de 2021 : 255910
* Pour 2022 : 126662

A partir de 2019, nous avons une complétude des données qui est parfaite. Le volume reste élevé ce qui pourrait nous laisser privilégier la plage 2020 à 2022 voire même 2021 à 2022 mais 2019-2022 devrait également convenir.

Pour la suite, nous privilégions la période 2020-2022 soit 3 ans d’historique.

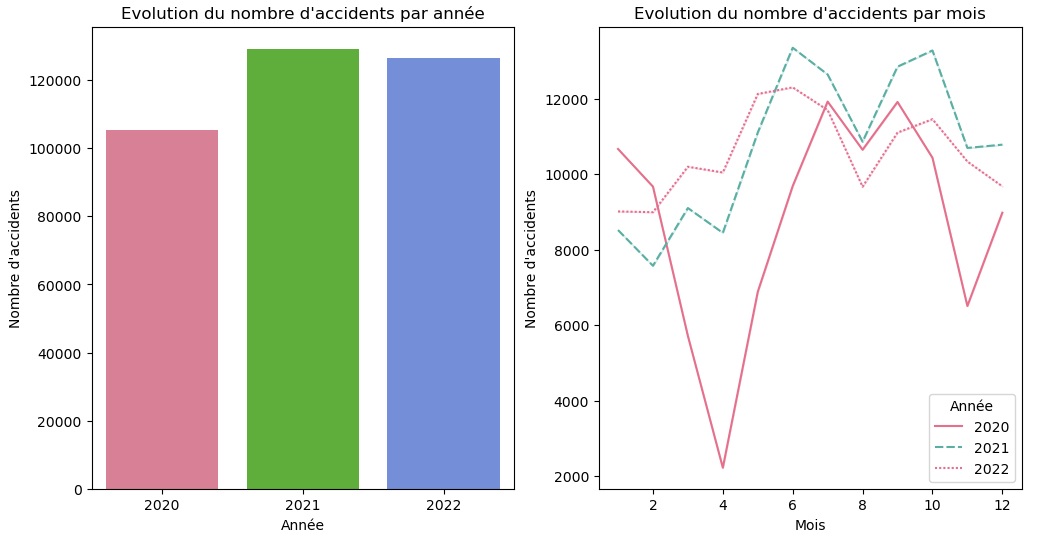
### Contrôles préalables des données (visualisation globale)

Les données seront chargées dans un DataFrame afin de faciliter leur analyse.

***Contrôle des dimensions de notre df***

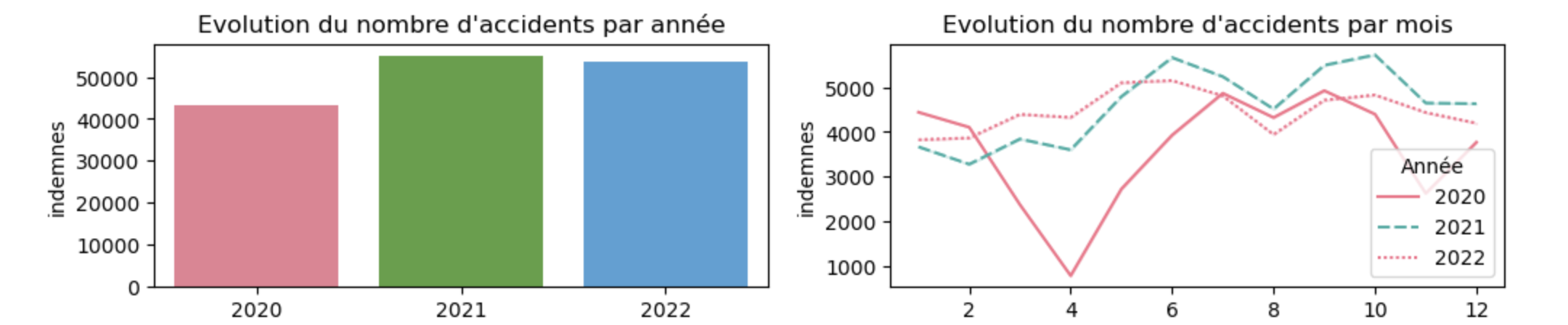
* 361.205 enregistrements pour 81 variables (dont 3 index)
* Pour un total de 159.564 accidents (répartis sur 3 années)

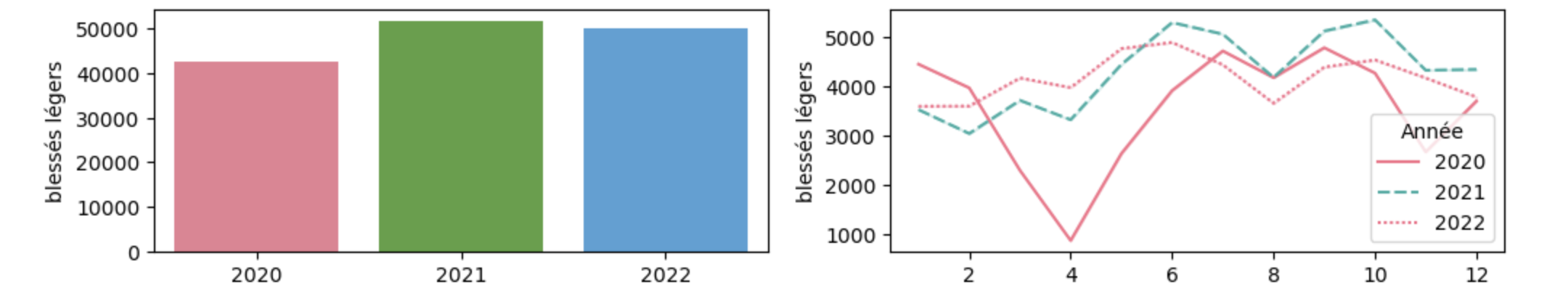
***Visualisation du nombre d’accidents***

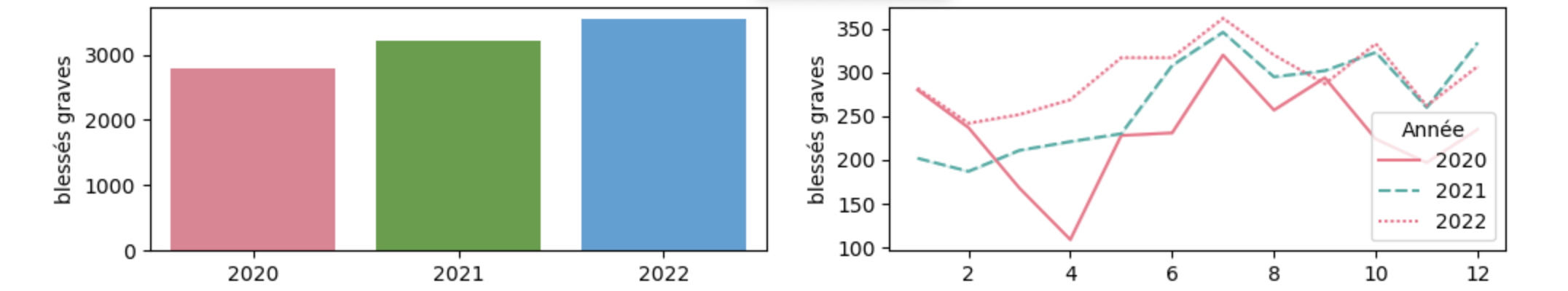


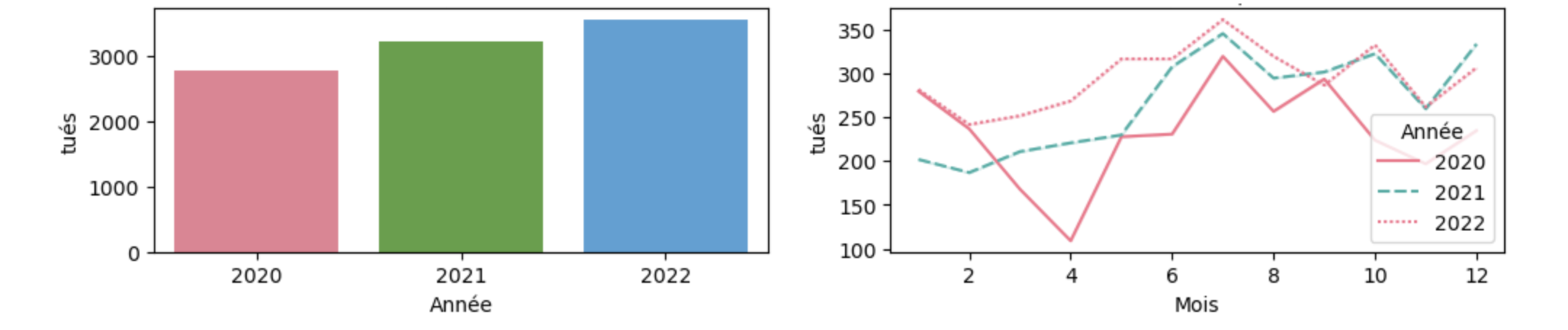
Le nombre d’accidents par année est relativement stable même si nous pouvons noter un décrochage sur 2020 et tout particulièrement en avril et en novembre qui s’expliquent par les confinements liés au Covid.

***Répartition par gravité et par année***



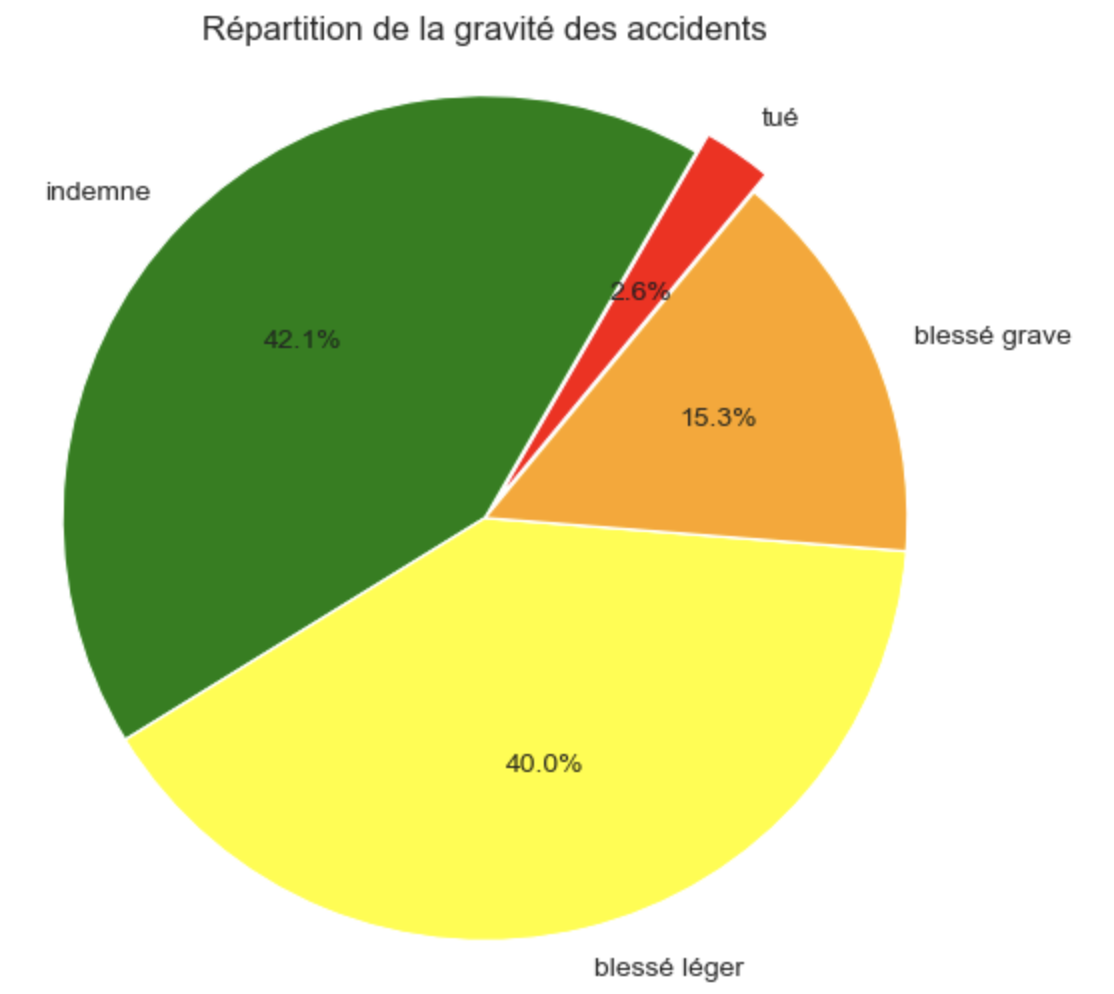
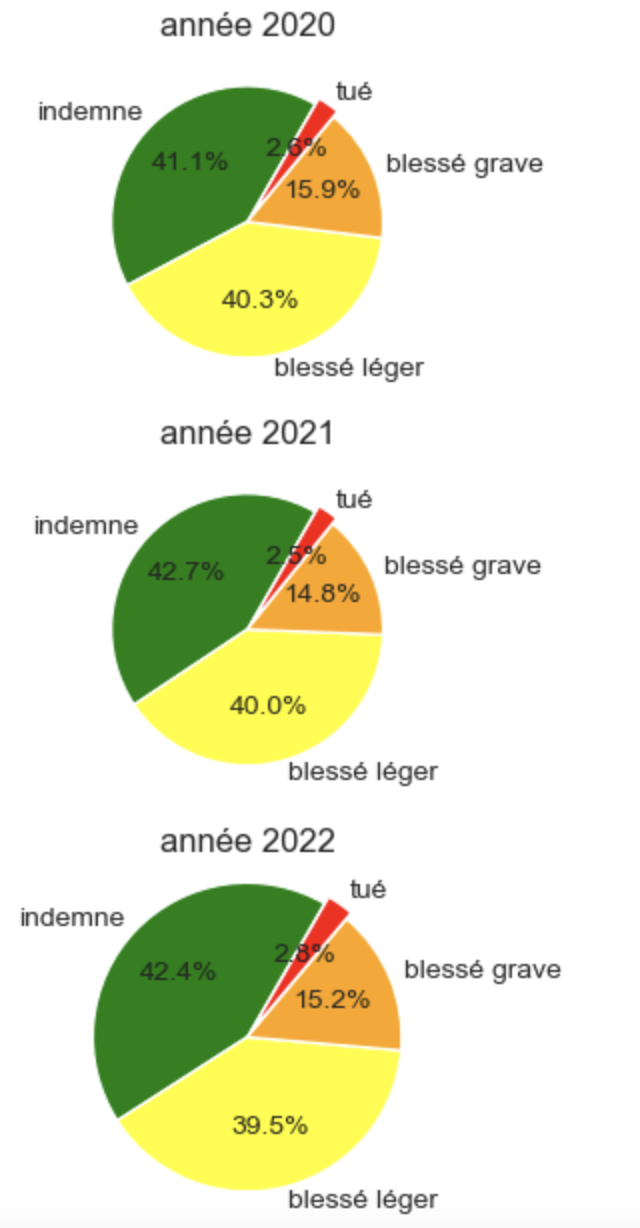






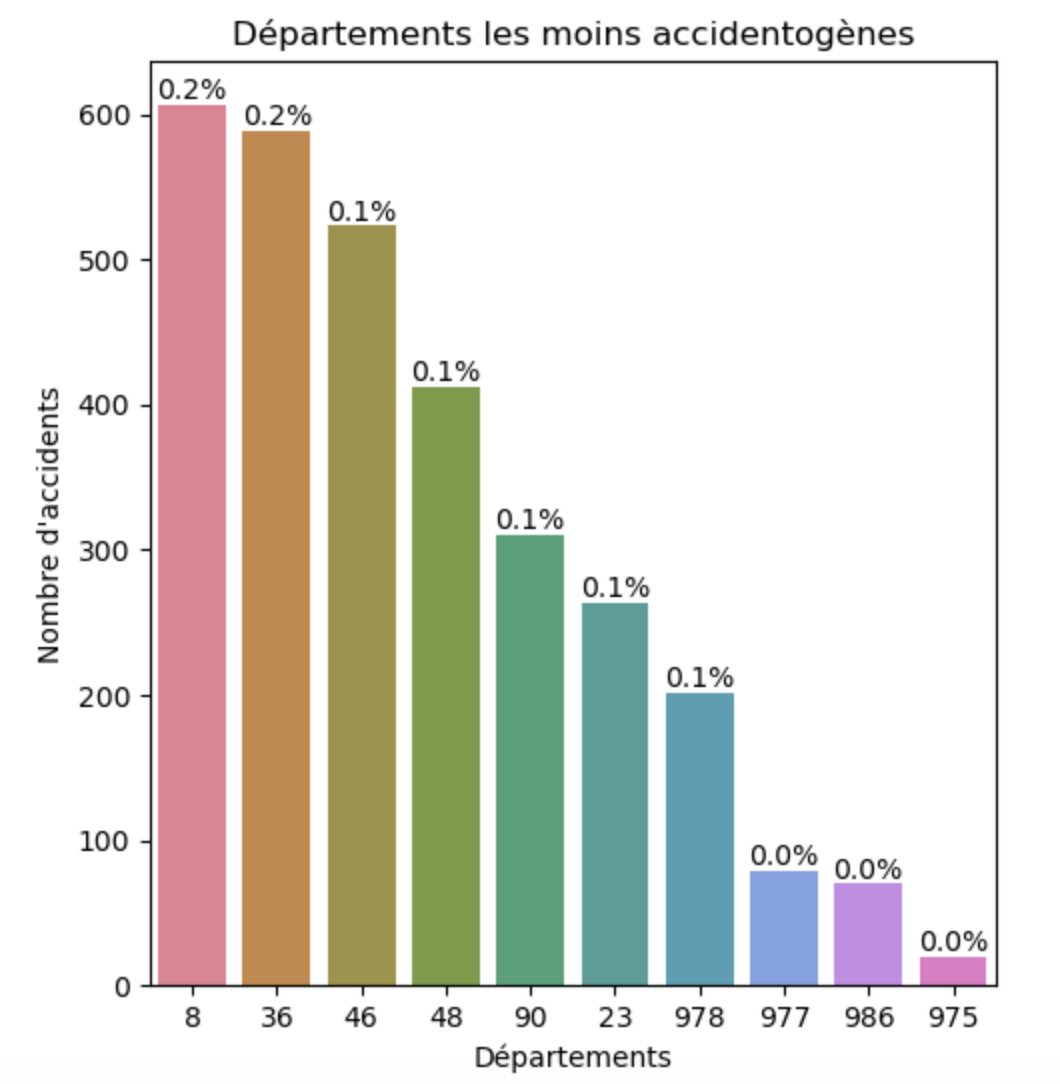
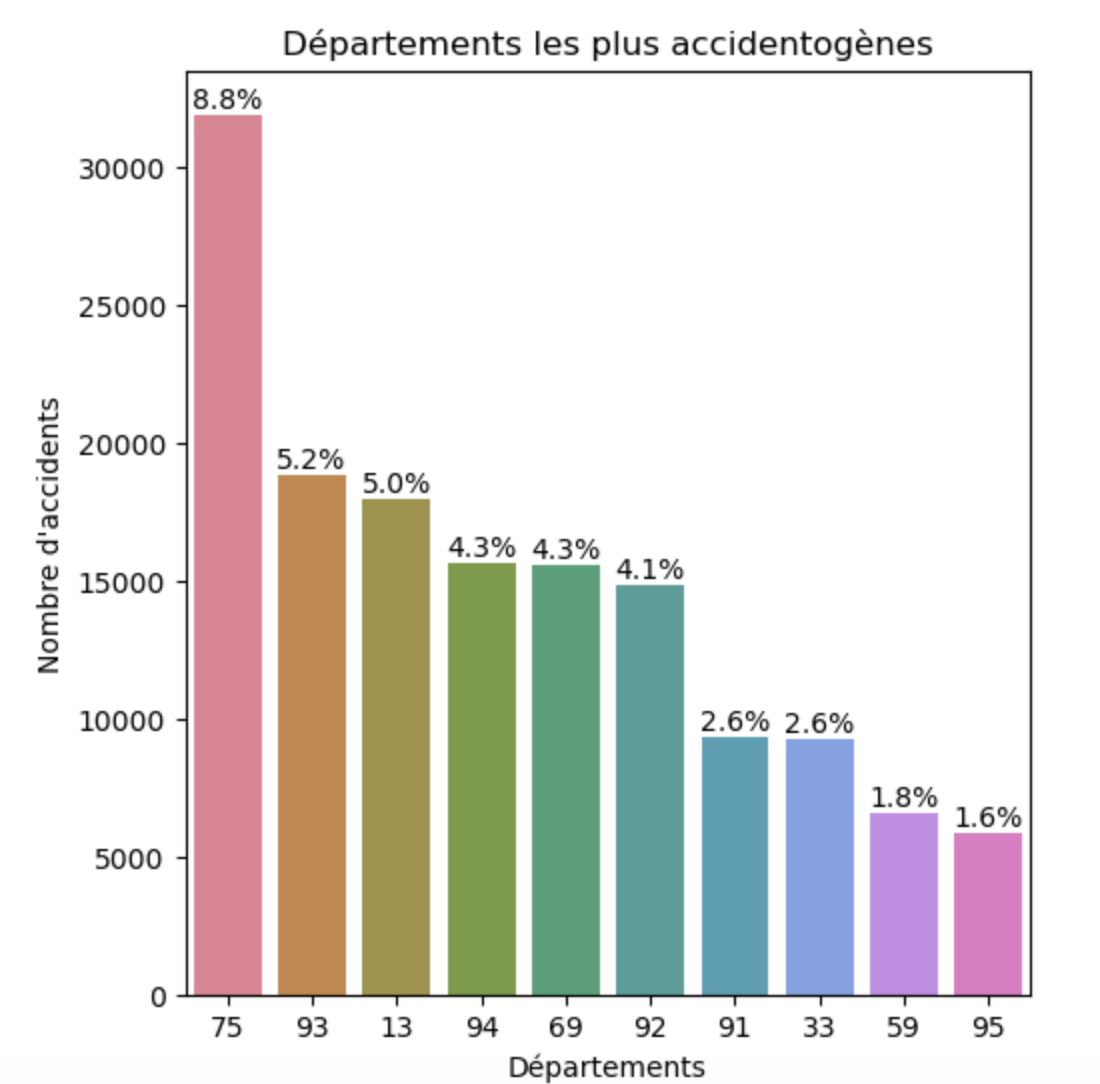
Si l’année 2022 a connue légèrement moins d’accident que l’année 2021, ils semblent avoir été plus violents que les années précédentes.

***Répartition de la gravité***

Cette répartition est relativement similaire d’une année sur l’autre.

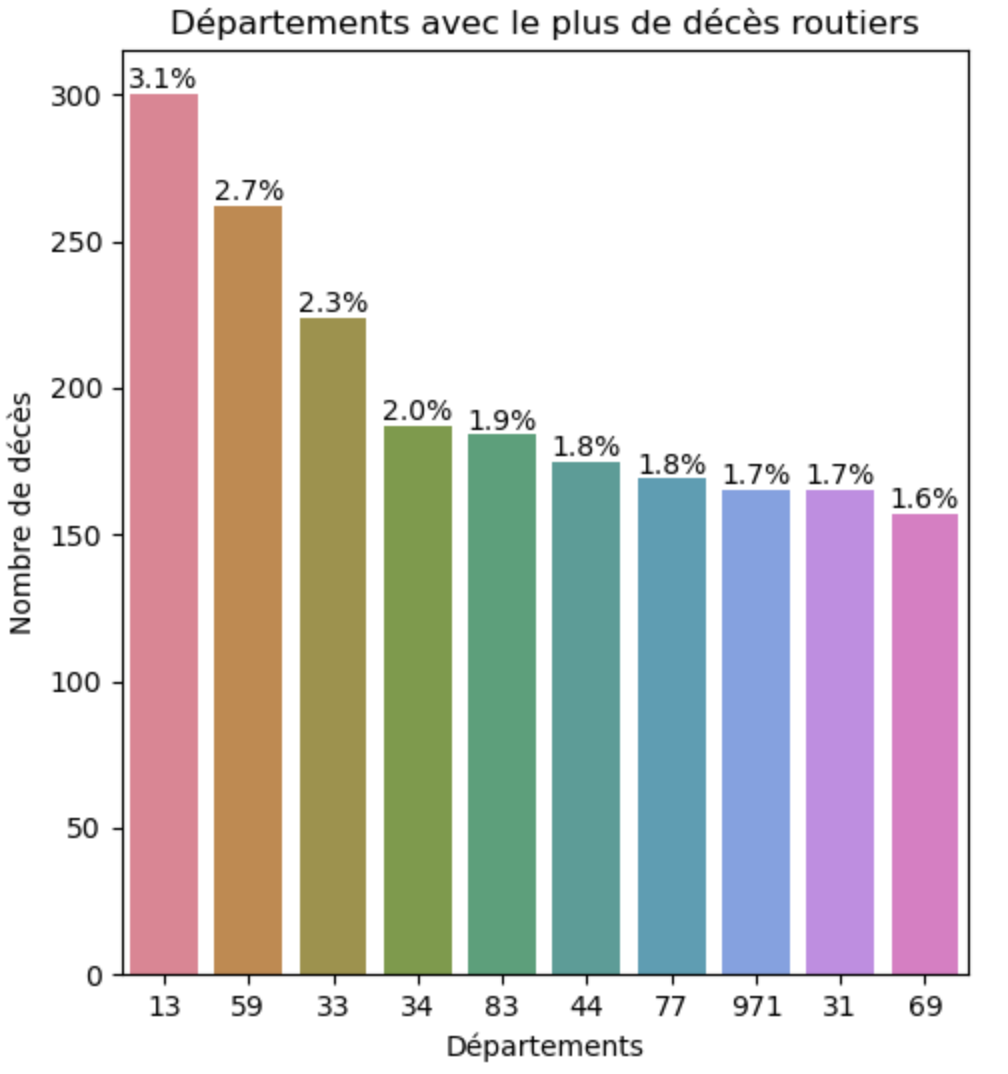
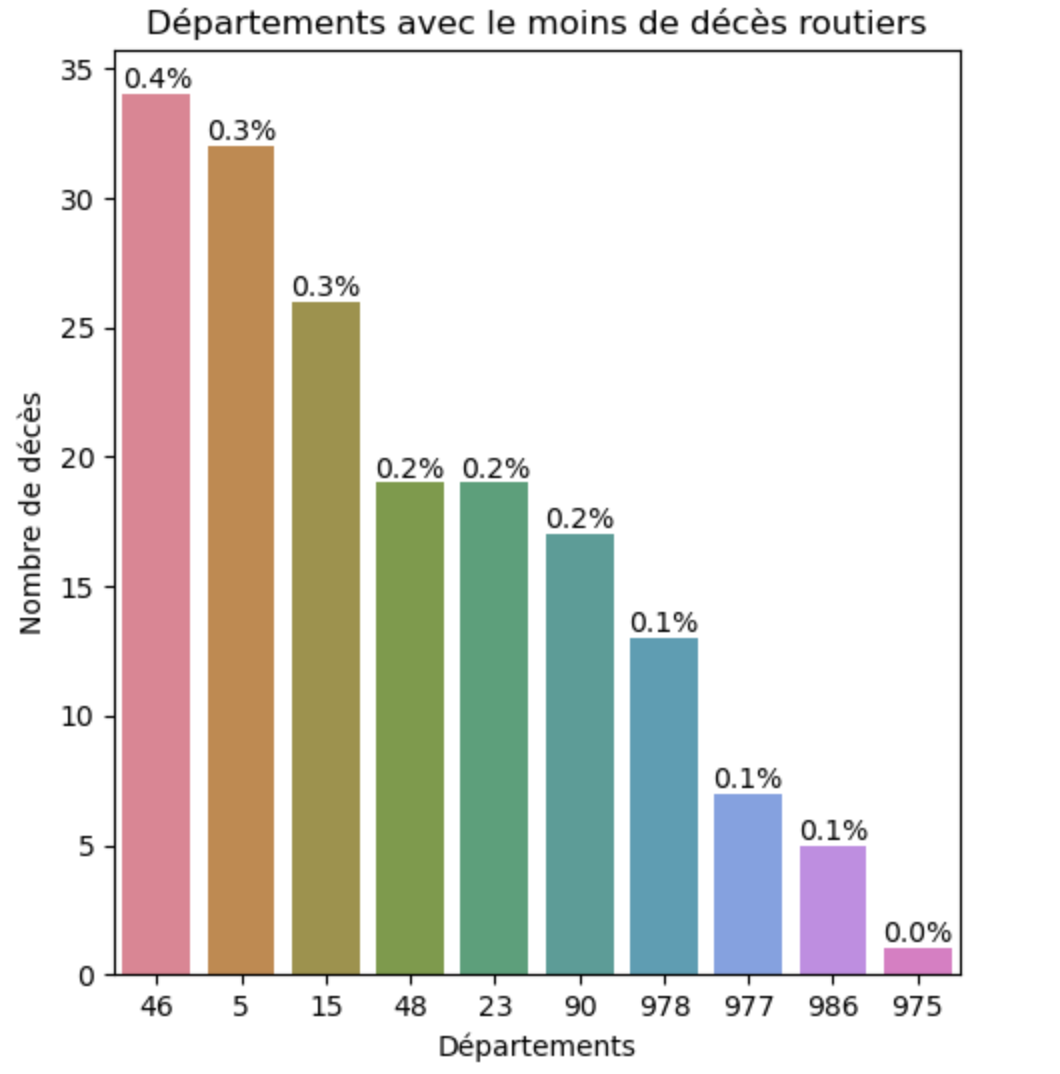
***Répartition des accidents par départements***



Les 10 départements les plus accidentogènes (dans lesquels nous retrouvons les métropoles les plus importantes tel que Paris, Marseille, Lyon ou Bordeaux) représentent à eux seul 40.3% des accidents (France + DOM/TOM). La région Parisienne porte à elle seule 26,6% du nombre d’accidents.

La population de la région île de France (sur la base de 13M habitants) représente 18% de la population totale pour 26,6% du nombre total d’accidents. Ceci doit s’expliquer par une densité de population en île de France 10 fois supérieure au reste du pays.

***Répartition des accidents mortels par départements***

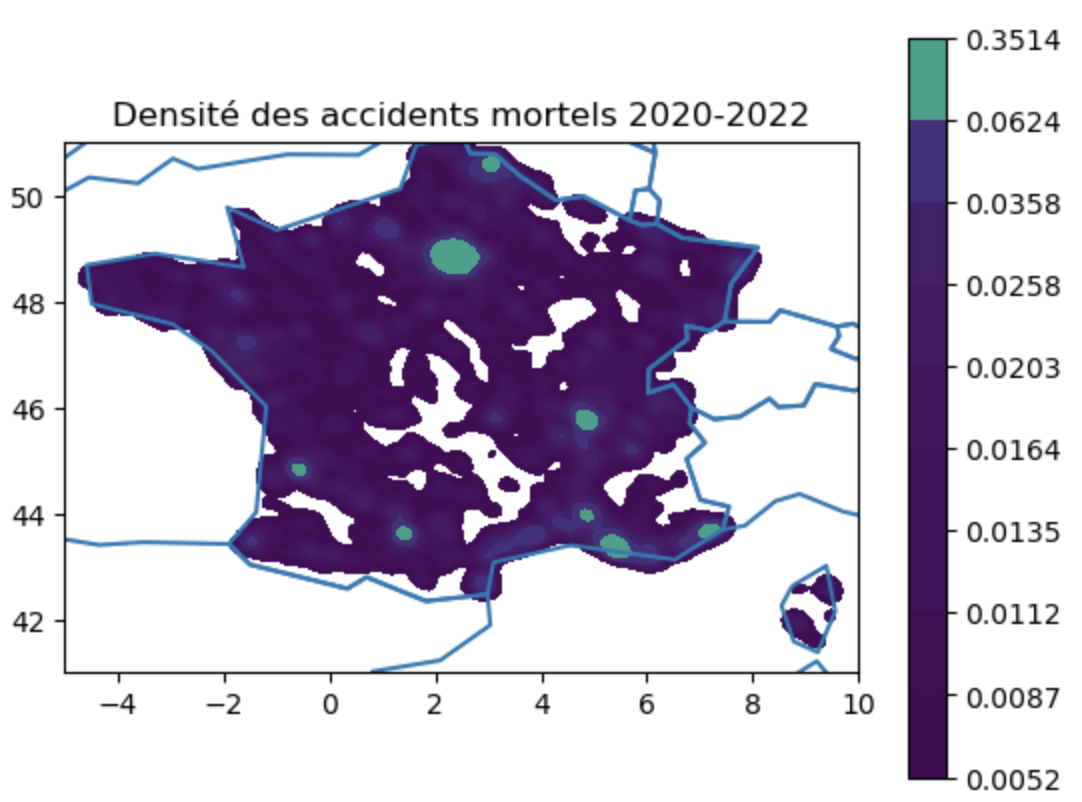
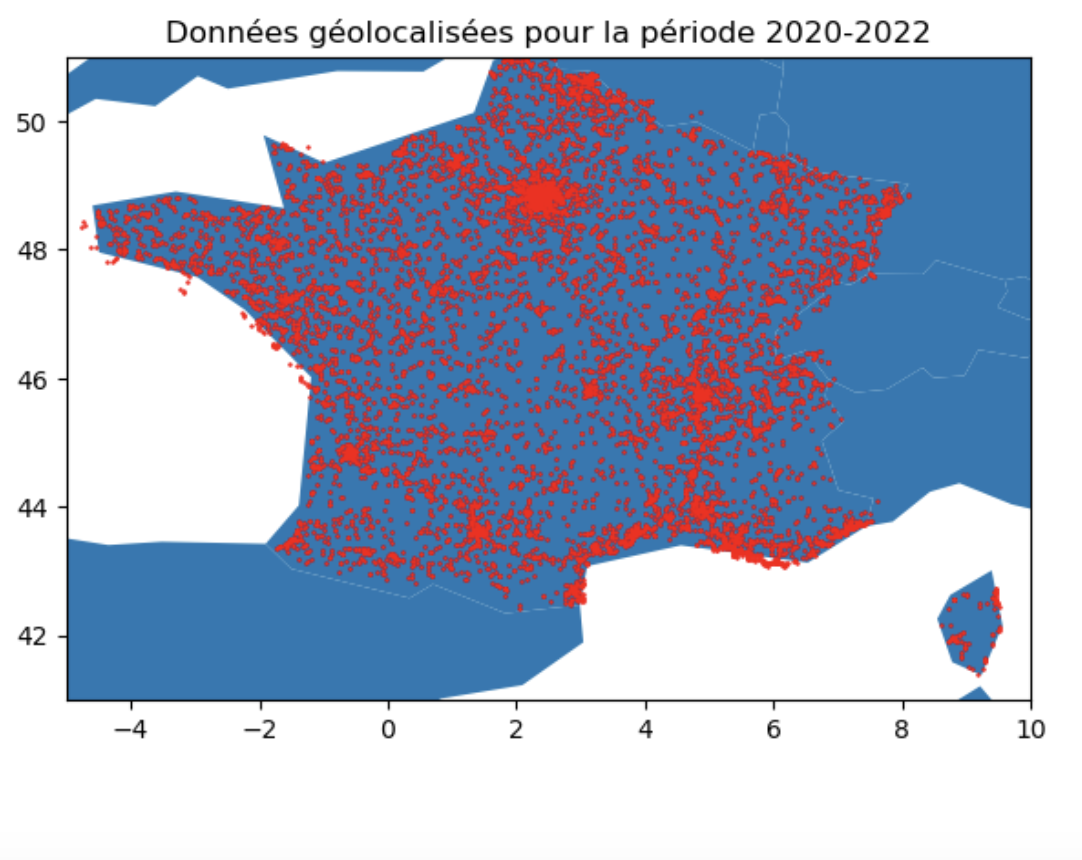
Si l’on retrouve majoritairement les mêmes départements en queue de classement que précédemment, il n’en est rien pour les départements les plus meurtriers où la région parisienne ne pèse plus que 1,8% des décès totaux sur les routes françaises avec d’ailleurs le département 77 qui ne faisait pas partie des 10 départements les plus accidentogènes.

A contrario, 4 départements du sud (bouches du Rhône en tête, la Gironde, l’Hérault et le Var) siègent dans le top 5 des départements les plus meurtriers sur les routes de France (9,3% des décès).

Le département du Nord vient compléter ce top 5 avec 2,7% des décès à lui seul (2ème département).

On constate que les accidents mortels sont globalement mieux répartis sur le territoire que l’ensemble des accidents.

***Géolocalisation des accidents mortels période 2020-2022***



Cette représentation géographique et notamment la courbe de densité de droite confirme les constatations que nous avons pu faire précédemment.

***Type de problématique :***

Deux approches possibles :

* Problème de classification supervisée sur la variable cible ‘user\_gravite’
* Problème de régression linéaire sur un scoring de dangerosité des accidents

Un indice de gravité calculé selon le coût normalisé des atteintes aux victimes par valeurs tutélaires est proposée dans une documentation gouvernementale (descriptif\_des\_variables\_pour\_le\_fichier\_2005\_à\_2010.pdf) :

* + - Tué = 100
    - Hospitalisé=10,8
    - Blessé léger=0,44

Pour la suite, sans demandeur pour challenger l’approche à privilégier, nous décidons de privilégier le problème de classification.

### Séparation de la base et méthode d’évaluation du modèle

Avant de démarrer l’une des parties les plus importantes d’un projet de Data Science : l’exploration des données, nous allons définir la méthode d’évaluation du modèle que nous souhaitons implémenter.

Nous décidons d’implémenter la technique de cross validation qui permet d’obtenir de meilleurs résultats :

1. *Utilisation efficace des données* : chaque observation est utilisée à la fois pour l’entraînement et la validation ce qui permet de mieux exploiter les données
2. *Estimation plus fiable des performances :* En moyennant les performances sur plusieurs plis, on obtient une estimation plus fiable
3. *Évaluation de la robustesse du modèle :* Grace à la mesure de la variance des performances sur différents ensembles de données : cela peut permettre d’identifier si notre modèle est sensible à la distribution des données et à la variable aléatoire
4. *Détection de problème de surapprentissage :* la comparaison des performances obtenues sur les ensembles de différents plis peut aider à détecter de l’overfitting en indiquant si le modèle généralise bien les données qu’il n’a pas vues pendant l’entraînement
5. *Facilite le paramétrage des hyperparamètres :* et donc aide à l’optimisation du modèle

Dans notre cas, nous avons un ensemble de données qui est très important. Nous pouvons en profiter pour retenir, en complément de la cross validation, la méthode d’évaluation de Holdout qui consiste à scinder l’ensemble de données en deux ensembles utilisés pour l’un pour la phase d’entraînement et pour l’autre pour la phase de validation finale.

Nous pouvons ainsi profiter des avantages des deux méthodes d’évaluation.

A ce stade, nous décidons de couper l’ensemble de données en deux sur la base d’un découpage aléatoire avec une répartition de 75% pour l’ensemble d’entraînement et 25% pour l’ensemble de test.

Remarque : Pour être cohérent avec la réalité, nous décidons de tenir compte de l’identifiant accident lors de ce découpage afin d’avoir des ensembles de données composés de l’intégralité des informations d’un accident (chaque accident sera intégralement soit dans le dataset train, soit dans le dataset test).

Les prochaines étapes d’entraînement du modèle ne seront réalisées que sur l’ensemble ‘train’. Nos pipelines de transformation de données seront lancés en fit uniquement sur le périmètre train afin de transformer les échantillons train et test sur cette base.

L’ensemble test ne sera utilisé que pour évaluer notre modèle.

### Regroupement des variables par typologie

Nous allons appliquer des méthodes d’exploration différentes selon la typologie de chacune des variables.

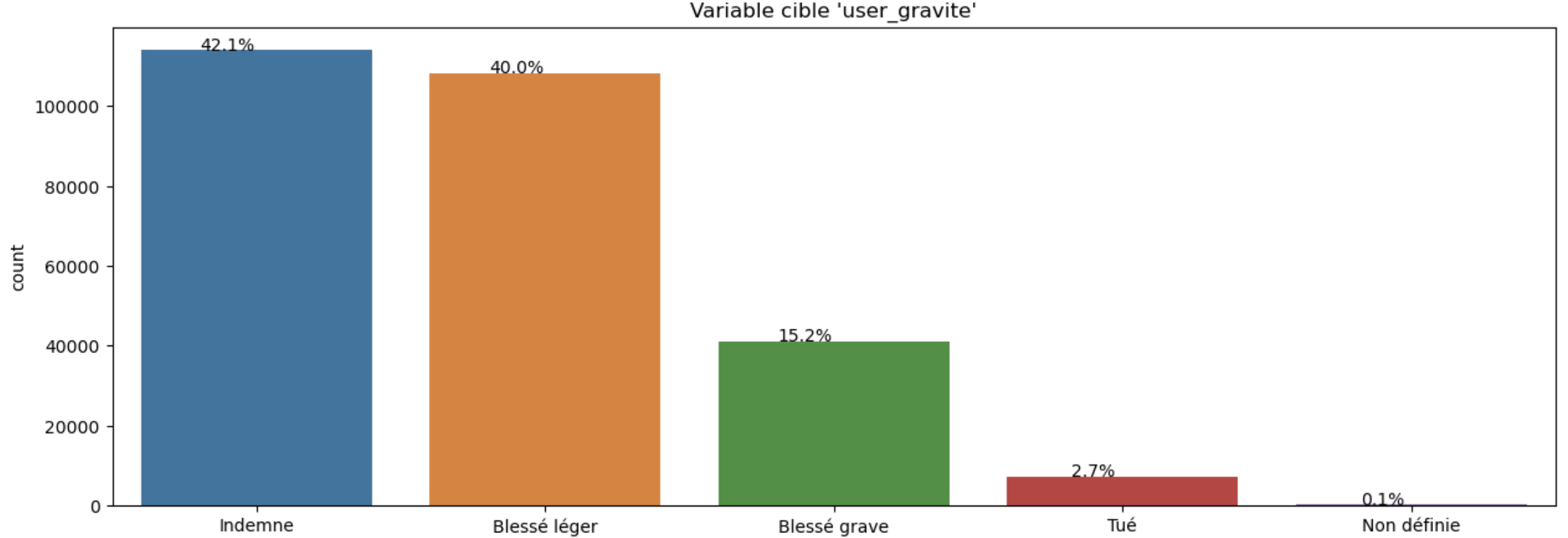
Typologie des variables explicatives identifiées :

* Index : id\_accident, id\_vehicule, num\_veh
* Date Time : carac\_an, carac\_mois, carac\_jour, carac\_hrmn
* Qualitatives binaires : carac\_agg, user\_sexe, agg\_is\_conducteur\_vehicule, agg\_is\_conductrice\_vehicule
* Qualitatives non binaires : carac\_atm, carac\_col, carac\_com, carac\_dept, carac\_int, carac\_lum, lieu\_catr, lieu\_circ, lieu\_infra, lieu\_plan, lieu\_prof, lieu\_situ, lieu\_surf, lieu\_vosp, vehi\_catv, vehi\_choc, vehi\_manv, vehi\_motor, vehi\_obs, vehi\_obsm, vehi\_occutc, vehi\_senc, user\_catu, user\_actp, user\_etatp, user\_locp, user\_place, user\_secu1, user\_secu2, user\_secu3, user\_trajet, agg\_catv\_perso
* Quantitatives : carac\_gps\_lat, carac\_gps\_long, lieu\_larrout, lieu\_lartpc, lieu\_nbv, lieu\_vma, vehi\_occutc, user\_vehi\_nais, agg\_nb\_pieton\_vehicule, agg\_nb\_passager\_vehicule, agg\_nb\_total\_vehicule, agg\_nb\_total\_velo, agg\_nb\_total\_vsp, agg\_nb\_total\_moto, agg\_nb\_total\_vl, agg\_nb\_total\_pl, agg\_nb\_total\_va, agg\_nb\_total\_conducteur, agg\_nb\_total\_conductrice, agg\_nb\_total\_pieton, agg\_nb\_total\_passager

## Analyse de la variable cible ‘user\_gravite’

La variable est une donnée catégorielle composée de 4 modalités ordonnées de 0 à 3 et d’une modalité -1 qui correspond aux valeurs manquantes (215 enregistrements).

Sur notre ensemble d’entraînement, il subsiste des missing values (valeurs ‘-1’) qui correspondent aux personnes en délit de fuite lors d’accident, personnes pour lesquels aucun suivi n’est possible.



**Pour la suite, les enregistrements sans variable cible seront écartés.**

Nous pouvons constater un déséquilibre important entre les différentes modalités et notamment les deux modalités les plus intéressantes pour nous :

* Blessé grave
* Tué

**Nous sommes face à un problème de classification supervisée déséquilibrée qu’il faudra traiter comme tel.**

## Analyse des variables explicatives

### Les variables d’index

Les variables d’index risqueraient de biaiser le modèle : nous ferons le nécessaire pour les exclure du périmètre des variables explicatives via une fonction de transformation qui sera appelée dans le pipeline de pre-processing.

**Pour la suite, les 3 variables index seront supprimées.**

### Les variables date/time

***Analyse date de l’accident***

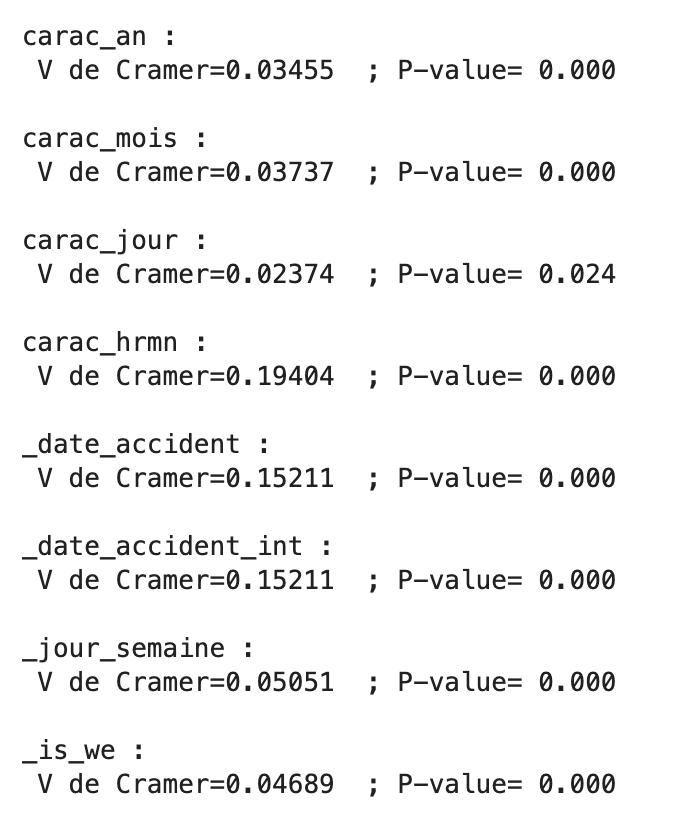
Nous avons à disposition 4 variables date/time. Nous pouvons les utiliser afin de générer de nouvelles variables qui pourraient nous aider par la suite :

* Date : date\_accident
* Jour de la semaine : jour\_sem
* WE : O/N, : we

Si la variable ‘date\_accident’ (format datetime) apporte une valeur ajoutée pour calculer ‘jour\_semaine’ et ‘is\_we’, nous utiliserons l’information au format int (YYYYMMDD) pour le test de Cramer : si la variable doit être conservée, elle ne pourra l’être au format datetime, incompatible avec de nombreux modèle de ML.

Pour chacune des variables nous réaliserons un test de chi2 ainsi qu’un test de Cramer qui nous confirme la corrélation avec la variable cible. Les résultats de ces tests seront archivés dans un dataframe ‘test\_stat’. Il permettra par la suite de rechallenger le périmètre des variables réellement maintenues.

Le test de Cramer nous montre que cette corrélation avec la variable cible est plus forte avec la variable date reconstituée qu’avec les trois variables à l’origine de cette reconstitution. Il en est de même avec les variables ‘is\_we’ et ‘jour\_semaine’ qui n’apportent, statistiquement, rien de plus que la date reconstituée.



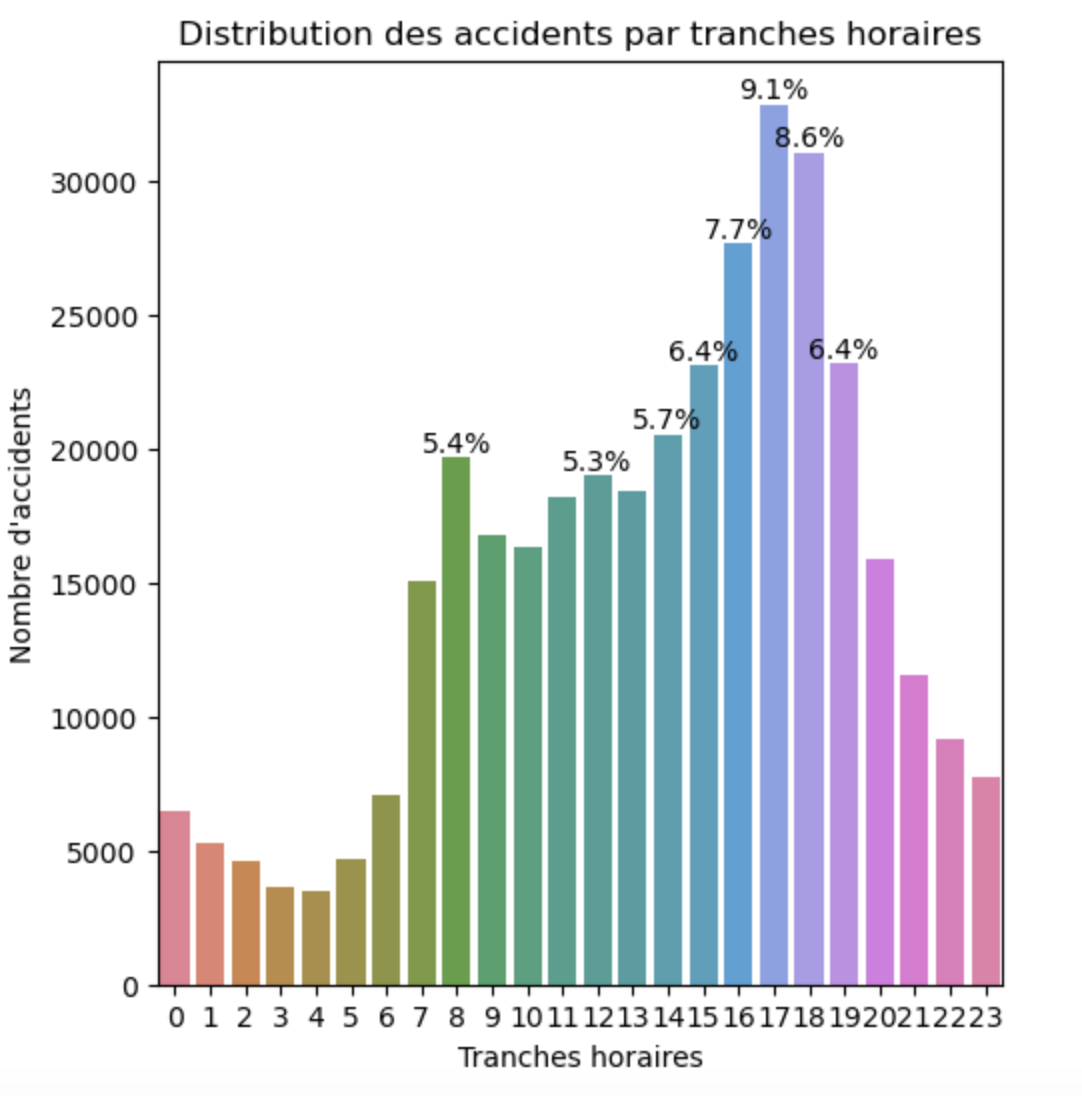
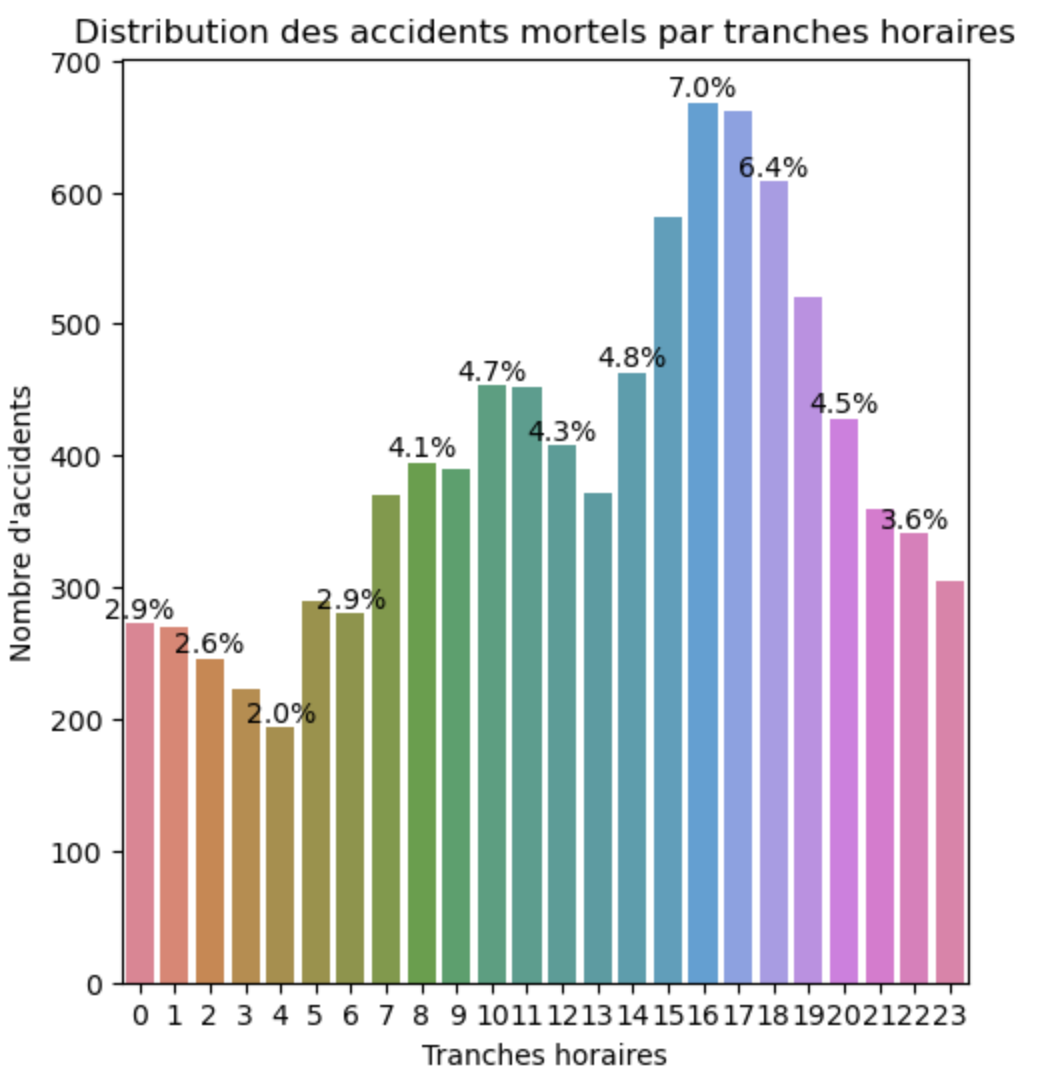
La variable date\_accident obtient un indice de V Cramer plus élevé que les variables ‘carac\_an’, ‘carac\_mois’, ‘carac\_jour’,.

A noter que dans notre approche, nous mesurons l’adhérence d’une variable avec la variable cible alors que le modèle pourrait retrouver une adhérence plus forte par le croisement de plusieurs variables explicatives.

En l’état, difficile de définir si une variable composée de 3 autres sera plus efficace que le maintien des 3 variables initiales.

La variable ‘carac\_hrmn’ semble quant à elle être plus corrélée que les autres.

La répartition des accidents (à gauche) et des mortels (à droite) semblent aller dans ce sens :

**Nous décidons pour la suite de ne conserver que ‘date\_accident’ (format int) et ‘carac\_hrmn’.**

*Cette position n’est qu’une position ‘à priori’ qui pourra être amenée à évoluer par la suite. Il ne faut le voir que comme une tendance : difficile par le biais de ces tests statistiques d’avoir un avis tranché sur le devenir de chaque variable.*

***Feature Engineering, période de vacances***

Nous avons à disposition un fichier contenant le détail des périodes scolaires. Nous allons essayer d’estimer l’intérêt de rajouter à notre dataset deux indications binaires :

* Période de congés sur lieu de l’accident : 0/1
* Période de congés hors lieu de l’accident : 0/1

*Cette étape n’a pas été réalisée pour le jalon de notre projet fil rouge.*

### Analyse détaillée des variables catégorielles binaires

La méthode retenue sera identique pour les 4 variables concernées :

* Analyse des missing values : Trois approches de remplacement seront testées :
  + Identification de tendances (valeur des autres variables pour chaque modalité de la variable en cours d’analyse) afin de les appliquer pour réaliser les remplacements des valeurs manquantes
  + Remplacement par la modalité la plus fréquente
  + Maintien de la catégorie par défaut (-1 dans notre cas)

Un test statistique de chi2 sera réalisé pour chaque approche afin de retenir celle qui maximise les résultats.

* Analyse de la répartition des différentes modalités sous forme de countplot :
  + Au global
  + En fonction de la variable cible
* Modalités ordonnées ou non (nécessité d’encoder la variable pour la suite si les modalités ne sont pas ordonnées)

Ces traitements ont été automatisés via deux fonctions :

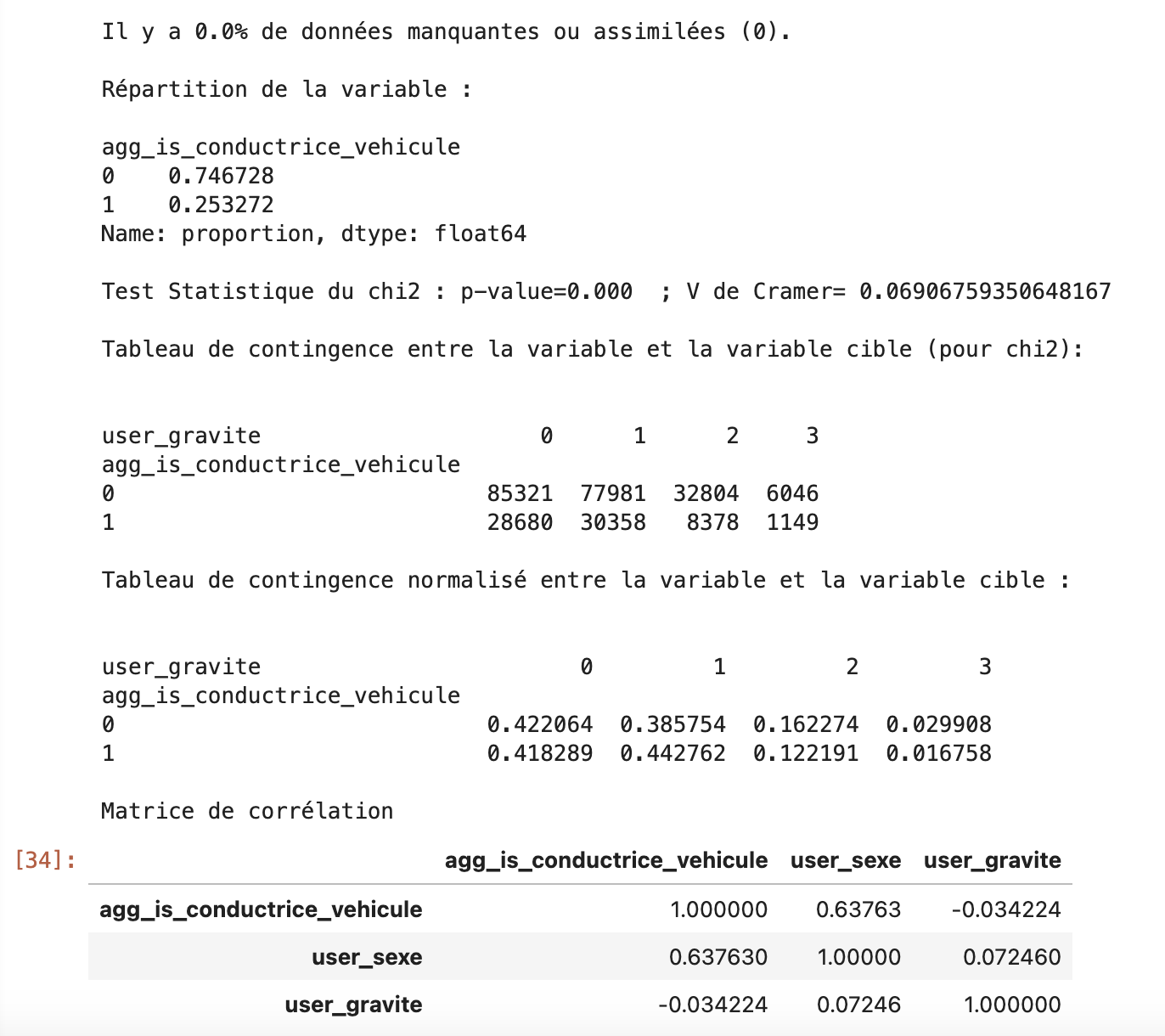
* Analyse\_variable\_binaire()
* Analyse\_valeurs\_manquantes()

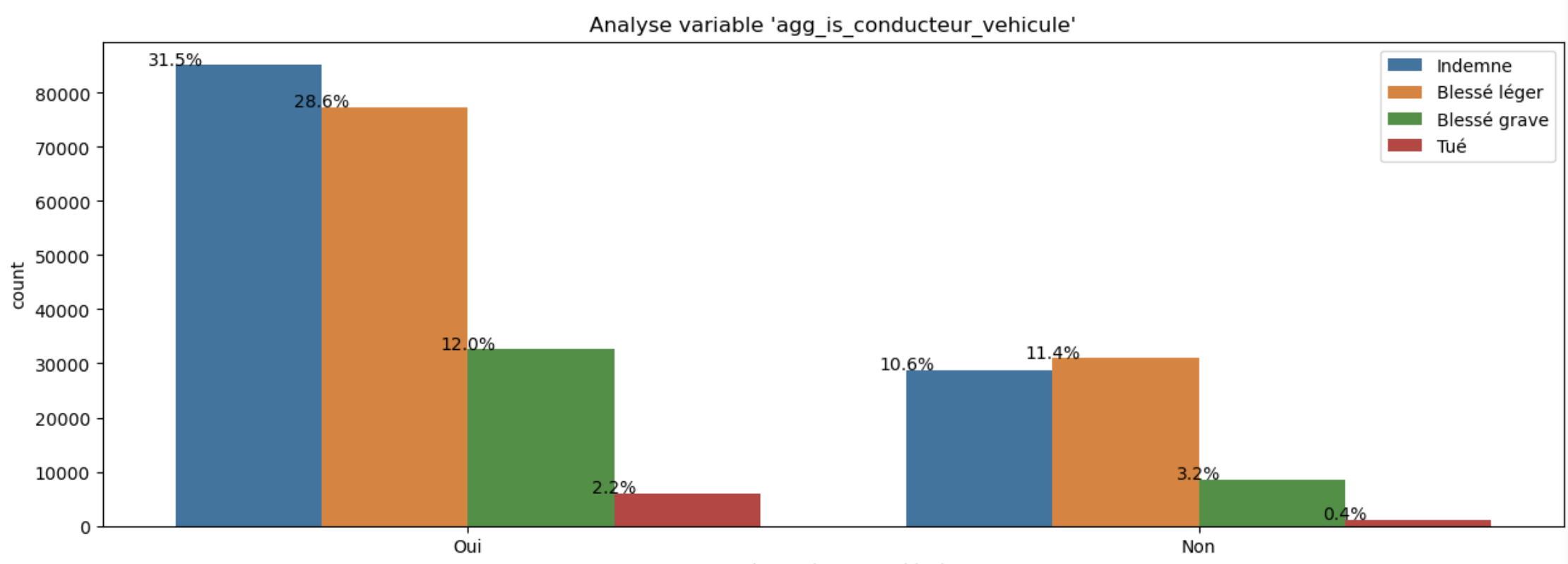
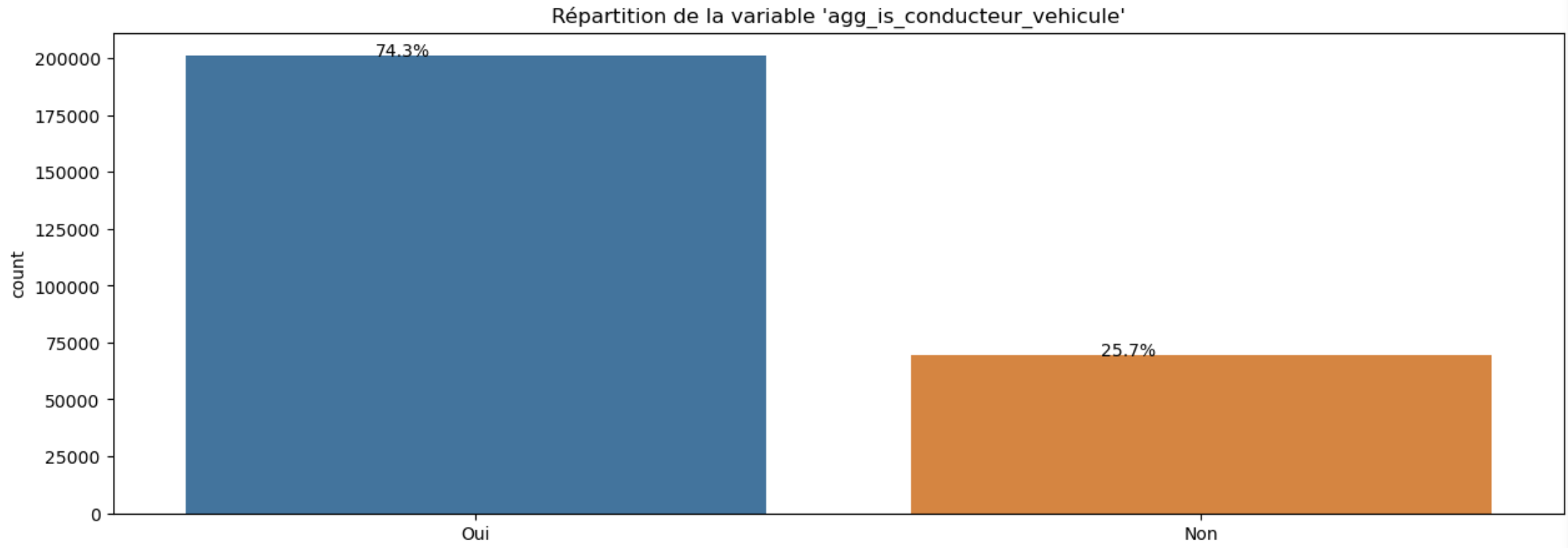
Cette analyse doit permettre d’identifier l’adhérence statistique avec la variable cible afin :

* De retenir la variable ou non comme variable explicative
* De définir la/les transformation(s) à privilégier pour la suite

L’ensemble des transformations envisageables pour chaque variable est référencé afin de les intégrer dans un pipeline. Plutôt que de décider immédiatement de privilégier arbitrairement une de ces transformations, nous prévoyons de les sélectionner dans la fonction de transformation de manière à les tester via une GridSearch.

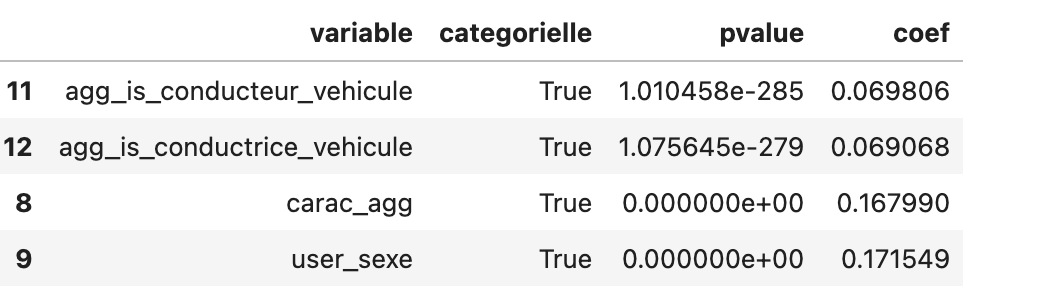
***Exemple de résultats obtenu (variable ‘agg\_is\_conducteur\_vehicule’)***





***Résultats obtenus***

Le test de chi2 montre qu’il existe une probabilité forte qu’il existe une dépendance entre chacune de ces variables et la variable cible mais les indices de Carmer semblent identifier une faible adhérence de ces données avec la variable cible.



Nous détaillerons plus tard dans ce document comment nous avons décidé du périmètre des variables à conserver. En regard des résultats obtenus, seuls les variables ‘carac\_agg’ et ‘user\_sexe’ pourraient être conservées.

### Analyse détaillée des variables catégorielles non binaire

L’analyse de ces variables non binaires sera très similaire aux variables binaires avec en complément, une étude de la corrélation de chaque modalité avec la variable cible afin d’identifier de possible regroupement de modalité qui pourraient améliorer la corrélation entre variable non binaire et variable cible.

Pour chaque variable non binaire nous réalisons :

* Un test de chi2 et de V Cramer avec les modalités existantes
* Un remplacement de la variable par une variable catégorielle binaire : pour réaliser ce regroupement de modalités, nous privilégions le taux de valeurs positives sur la modalité cible qui nous intéresse le plus (3=Tué) : si la modalité dépasse un seuil x, sa modalité de regroupement sera positionnée à 1 sinon 0.

Nous rechercherons la valeur de x la plus performante permettant de maximiser le V Cramer.

* Un second test de regroupement non binaire cette fois sera réalisé. Le principe sera le même : les modalités dont le pourcentage de tué dépasse un taux x seront maintenu en l’état (et non plus regroupé dans une modalité 1), les autres seront regroupées dans une nouvelle modalité (autre).

La valeur de x sera également recherchée de manière à maximiser l’indice de Cramer.

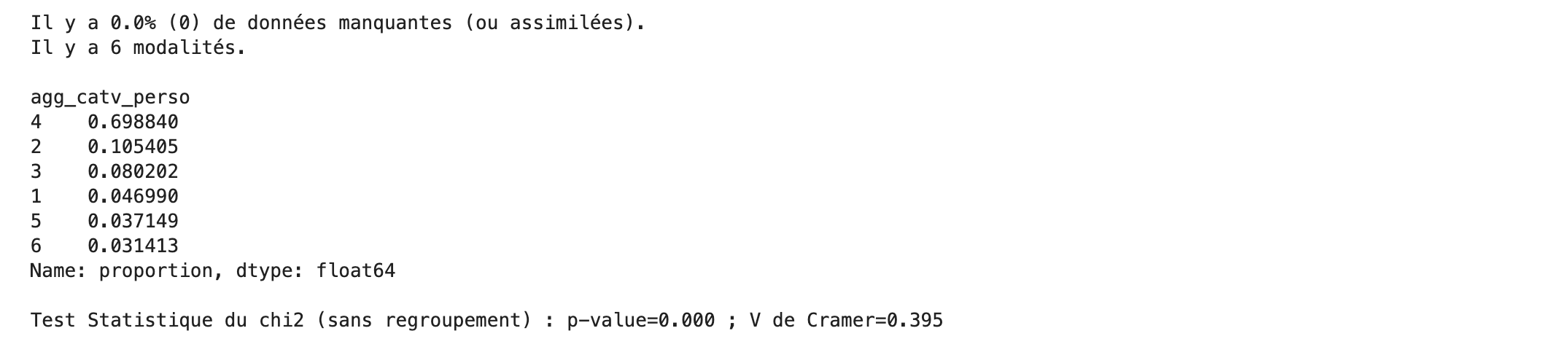
Ces tests seront réalisés :

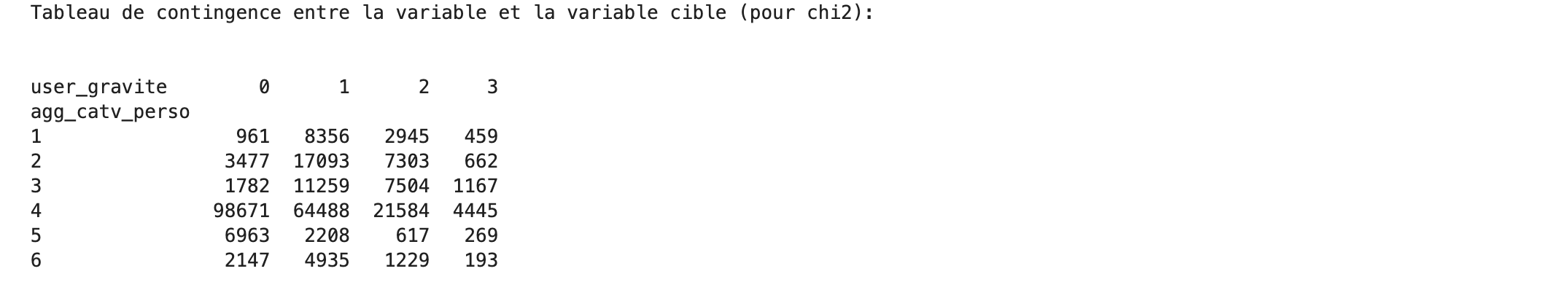
* Une première fois avec les missing values remplacées par la valeur -1
* Une seconde fois en privilégiant la modalité la plus fréquente

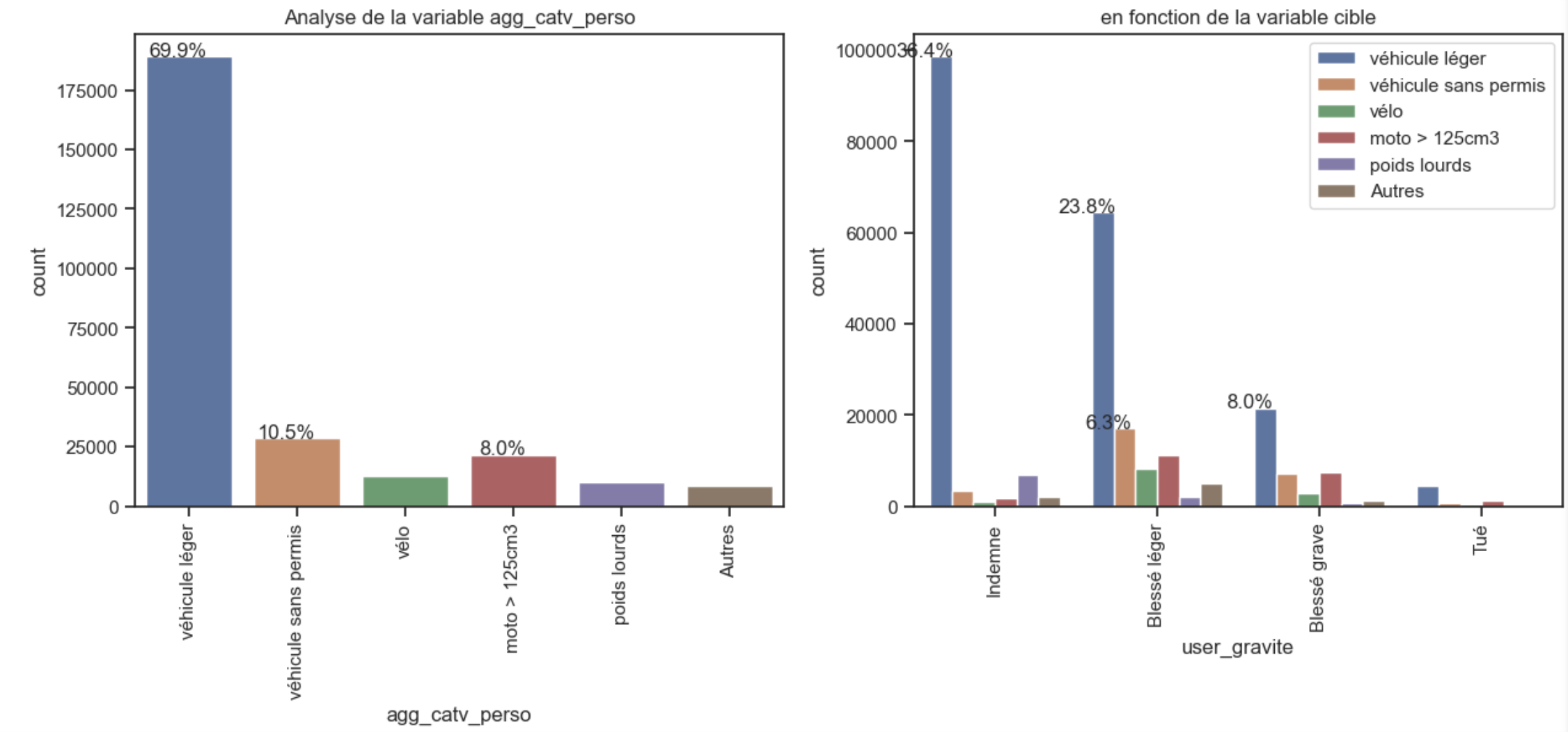
In fine, nous privilégierons la solution qui semble donner les meilleurs résultats.

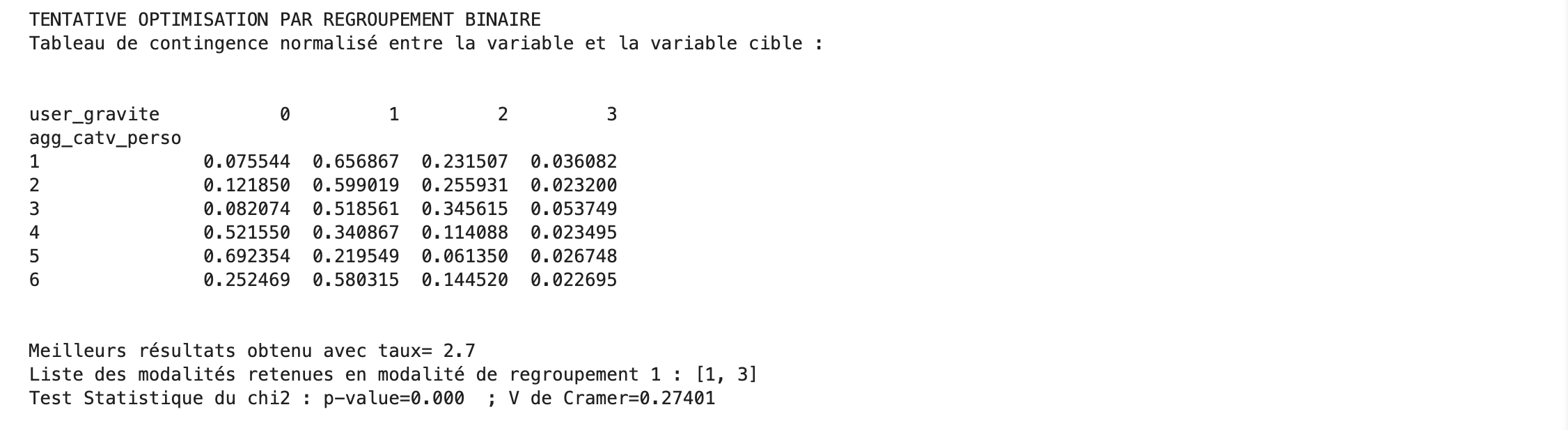
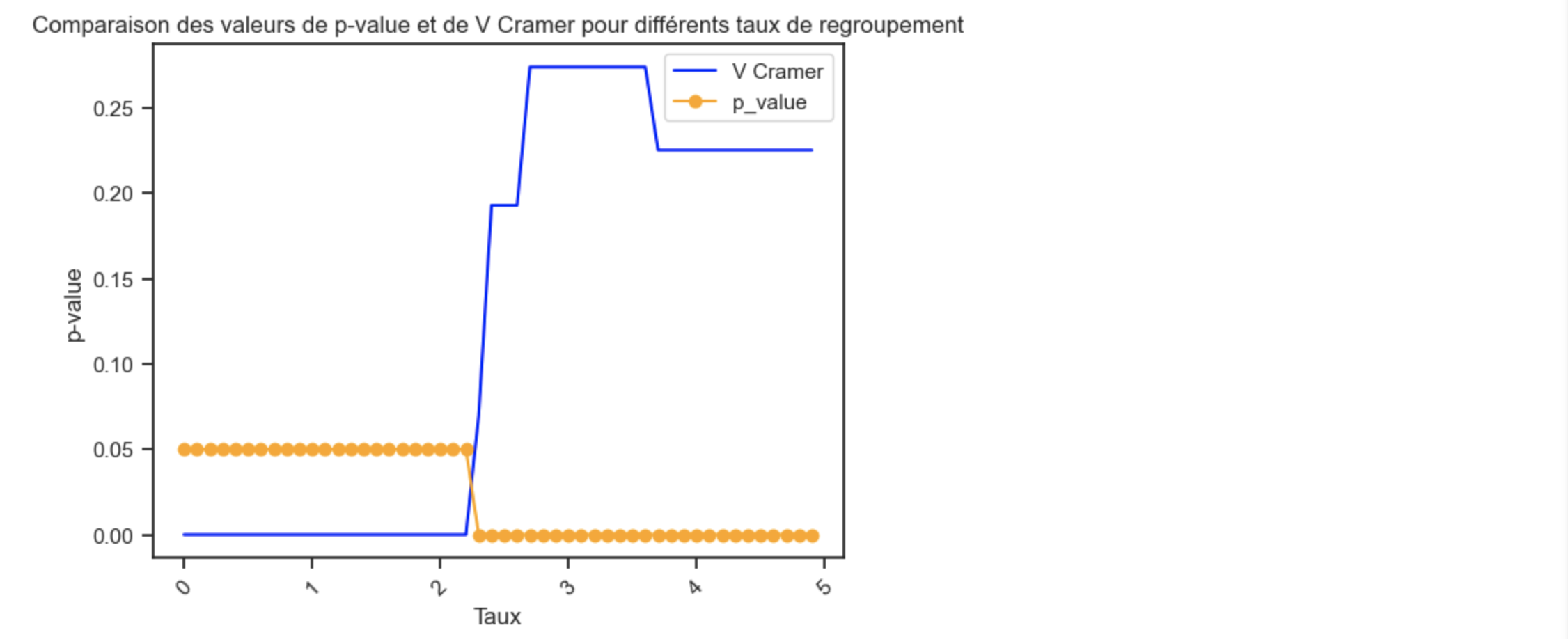
Là encore, les solutions de transformation envisagées seront codées dans une fonction dédiée de manière à pouvoir être exécutées via un GridSearch / pipeline afin d’infirmer/confirmer la tendance que nous venons d’identifier.

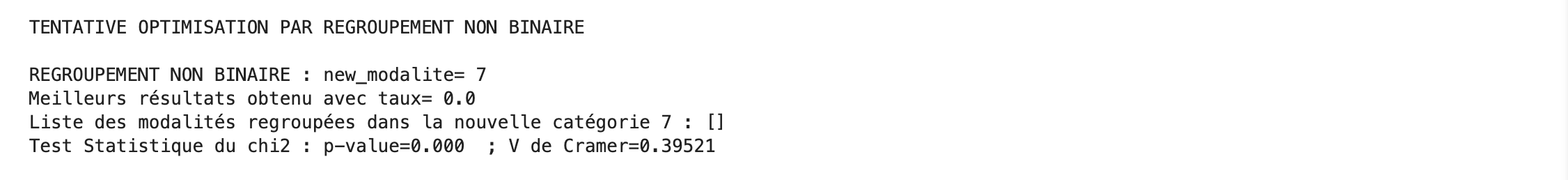
***Exemple de résultats obtenus (agg\_catv\_perso)***

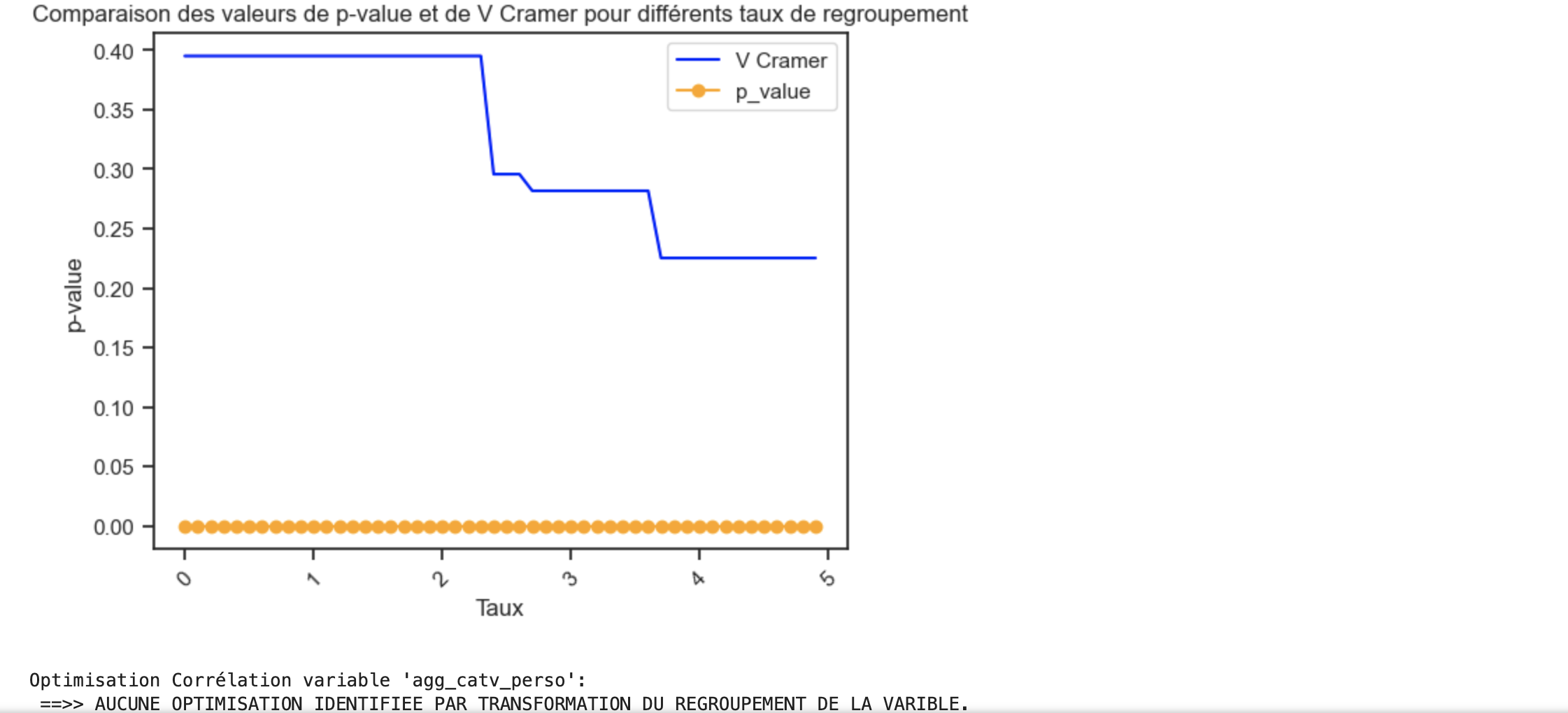










***Interprétation des résultats***

Si le test de Cramer permet d’estimer l’adhérence entre la variable étudiée et la variable cible, les opérations réalisées ne semblent pas concluantes :

* La variable ‘vehi\_senc’ qui est très vraisemblablement une donnée purement statistique obtient une p\_value inférieure à 0,05%. Seul Le test de Cramer semble confirmer cette faible adhérence (Cramer=0.022) : est-ce à dire qu’avec un score aussi bas, il est préférable d’exclure la variable des variables explicatives.

**Cette interrogation reste entière et devra être traitée ultérieurement.**

* Aucune optimisation n’a été identifiée par le biais de la méthode retenue.
* Nous constatons que la majorité des variables, bien que statistiquement dépendantes de la variables cible semblent que très peu liée : quelle valeur apporter au test de Cramer : **à ce stade, cela reste une inconnue.**
* Deux variables particulièrement liées à la variable cible : ‘vehi\_catu’ et ‘user\_secu1’ (V Cramer =0.483).

Plusieurs milliers de missing values pour la seconde : nous apportons une vigilance particulière au remplacement de ces valeurs afin de minimiser le biais qui pourrait en découler.

* + - Remplacement par la modalité la plus représentée
    - Prise en compte des catégories de véhicules pour choisir la modalité la plus représentée mais en lien avec ce type de véhicule (ex. on va exclure le port de la ceinture pour les 2 roues).

### Analyse détaillée des variables quantitatives

Chaque variable quantitative sera analysée unitairement afin d’identifier sa distribution, les missing values ainsi que les Outliers afin de corriger au mieux les données du dataset.

Un test Anova ainsi que le coefficient de corrélation seront systématiquement calculés afin d’estimer la corrélation entre chaque variable et la variable cible.

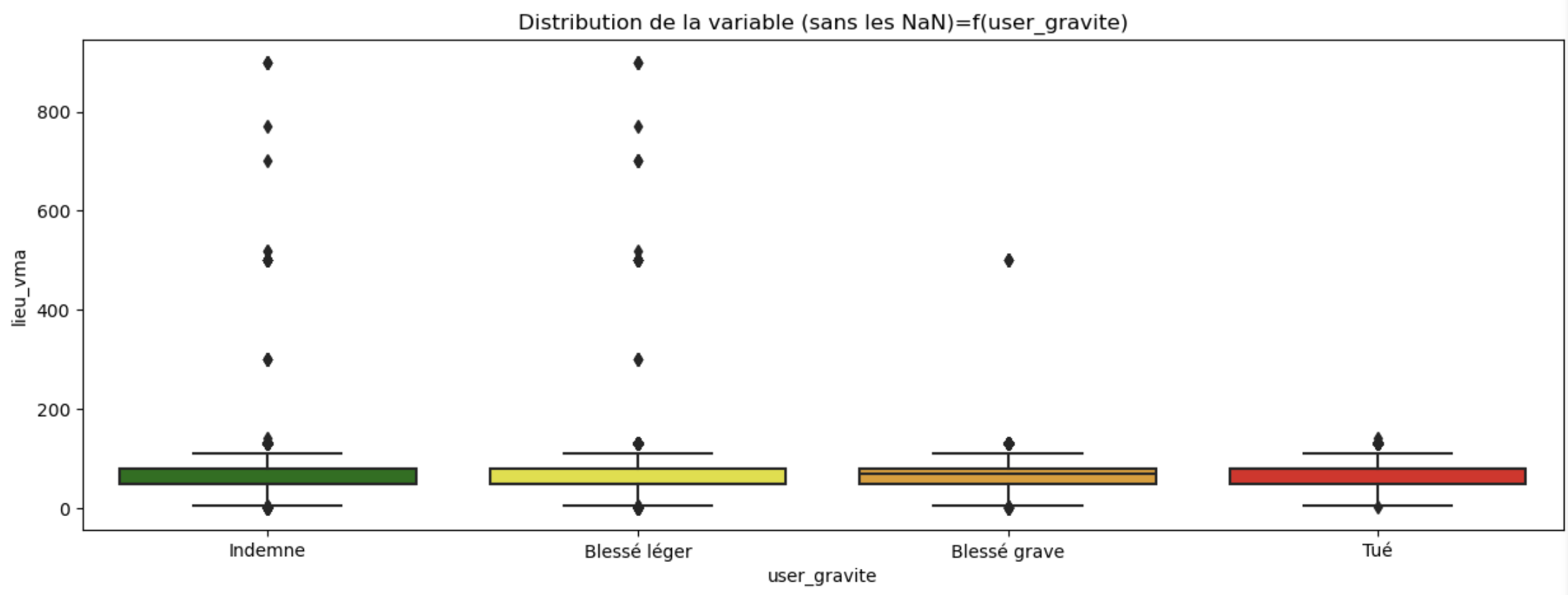
Enfin, dans le cas où des variables quantitatives seraient éligibles, un dernier contrôle de corrélation entre ces variables explicatives sera réalisé et tout particulièrement entre les variables agrégées qui ont été construites à partir d’autres variables du dataset.

***Résultats obtenus sur les variables quantitatives***

1. Sur les 13 variables d’agrégation de type quantitative, seules 2 variables semblent avoir une corrélation suffisante ‘agg\_nb\_total\_conducteur’ et ‘agg\_nb\_total\_vehicule’
2. Hormis ces deux variables, aucune ne semblent suffisamment corrélée avec la variable cible pour être utile par la suite.
3. De l’analyse des valeurs extrêmes (Outliers), il ressort qu’elles restent toutes cohérentes et plausibles.

Une exception malgré tout pour la variable ‘lieu\_vma’ correspondant à la vitesse maximale autorisée sur le lieu de l’accident.

Voici la distribution de cette variable :



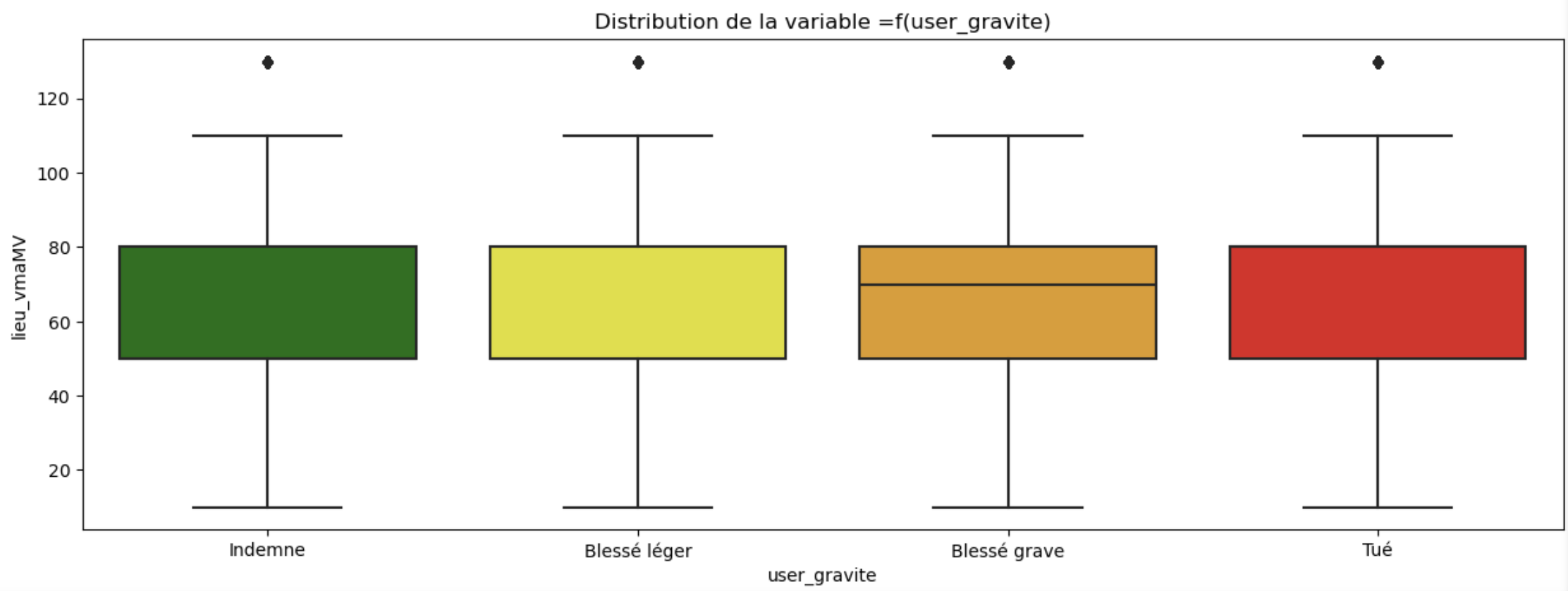
Après analyse détaillée, nous pouvons largement supposer que les 7128 Outliers positionnés au-dessus de l’interquantile Q3 et 125 en dessous de Q1 correspondent à des erreurs de saisies que nous nous proposons de corriger comme suit :

* Les valeurs inférieures à 10 seront multipliées par 10
* Les valeurs supérieures à 130 seront divisées par 10

Ces règles permettent de corriger 99% des outliers. Il subsiste 15 enregistrements associés à une vitesse de 130km/h et non associées à la catégorie autoroute.

* Si l’accident est hors agglo : on positionne la vitesse à 110 (on suppose une erreur de saisie de 1 caractère 130  110)
* Si l’accident est en agglo (c’est le cas de 4 accidents sur les 15) et après un contrôle par géolocalisation qui a confirmé le bien fondée de la règle, la vitesse est forcée à 50.

Après correction de ces outliers, nous obtenons le graphe suivant qui contient toujours des Outliers mais cohérents :



### Matrice de corrélation

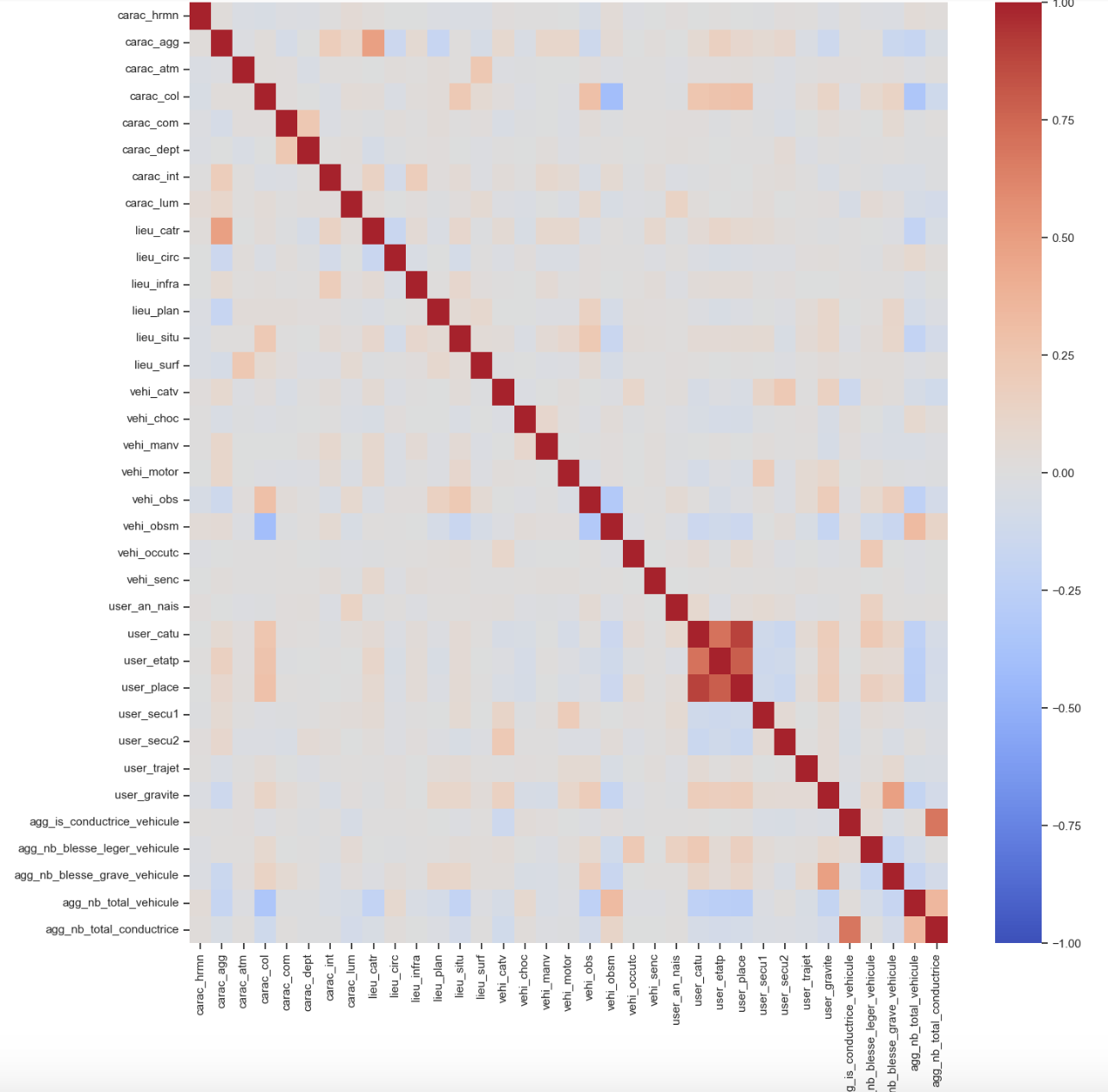
Représentation d’une matrice de corrélation, visuellement proposée sous forme d’un heatmap afin :

* De confirmer visuellement la corrélation entre variables explicatives et variable cible
* D’identifier de possible corrélation entre 2 variables explicatives qui n’auraient pas déjà été détectée.

Nous privilégierons dans ce cas celle ayant une meilleure corrélation avec la variable cible. L’autre sera écartée afin d’éviter le phénomène de multi colinéarité

***Matrice de corrélation obtenue***

Nous constatons une forte colinéarité entre les variables en lien avec les informations liées exclusivement aux piétons. Nous conserverons pour la suite la variable ayant le meilleur indice de Cramer.



## Analyse des résultats statistiques

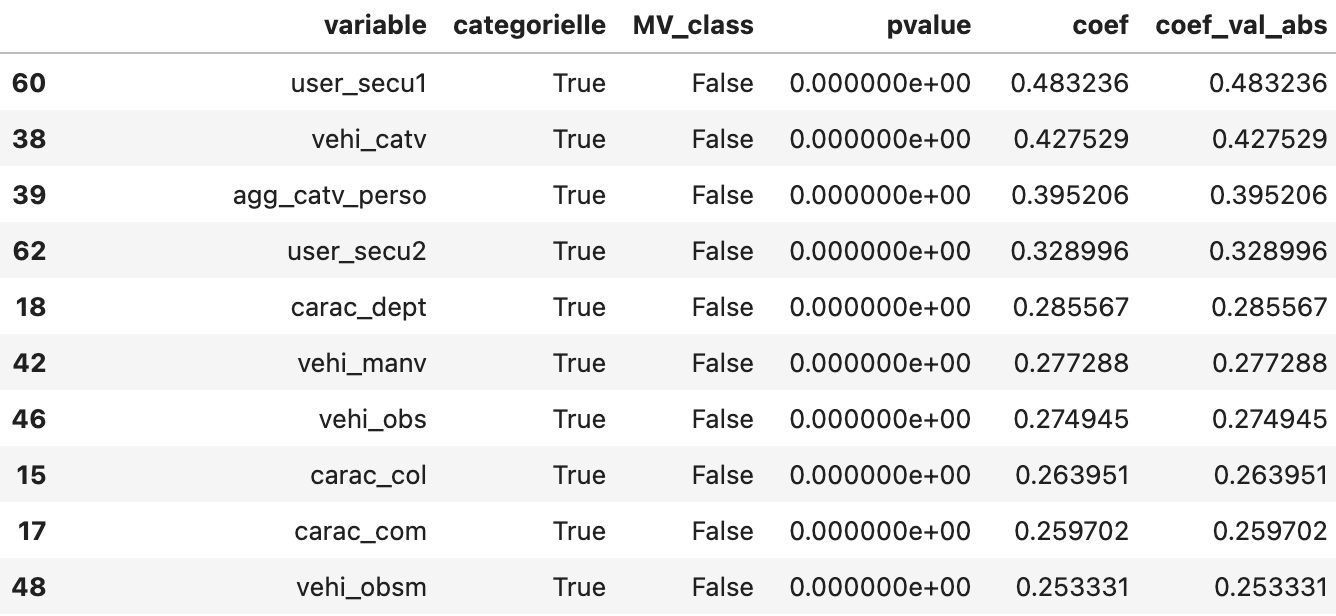
Nous avons conservé dans un dataframe ‘test\_stat’, le détail des tests statistiques réalisés. Nous y retrouvons :

* Le nom de la variable analysée,
* Un indicateur « catégorielle » : O/N
* p\_value (Chi2/Anova) selon la valeur de « catégorielle »
* coef. (indice de Cramer / coef de corrélation) selon l’indicateur de « catégorielle »
* Un indicateur MV\_class O/N : à O si la méthode de remplacement des missing values privilégie la modalité la plus fréquente, à N si remplacement par une catégorie ‘-1’.

Ce dataframe a été travaillé de manière à ne conserver qu’une seule entrée par nom de variable en privilégiant la valeur absolue du coefficient de test statistique le plus élevé. Nous avons ainsi l’information :

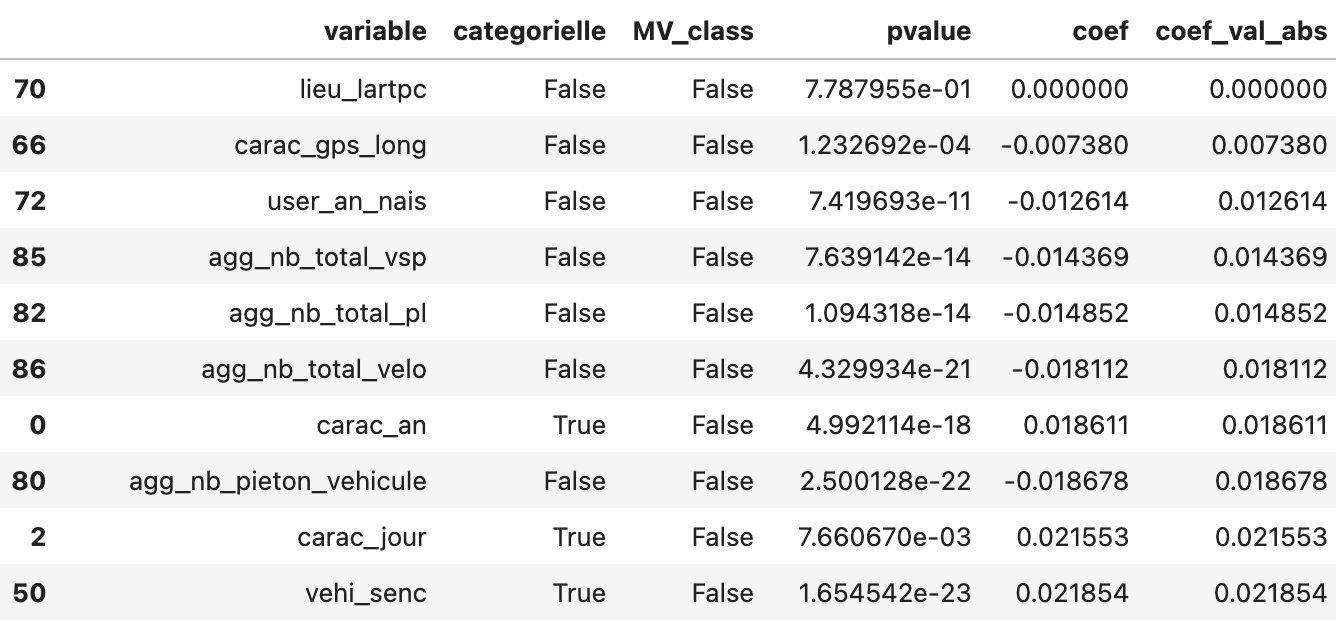
* Du meilleur test obtenu pour chaque variable et à partir de quelle règle de substitution des missing values.

***Top 10 des variables les plus fortement corrélées statistiquement à la variable cible***

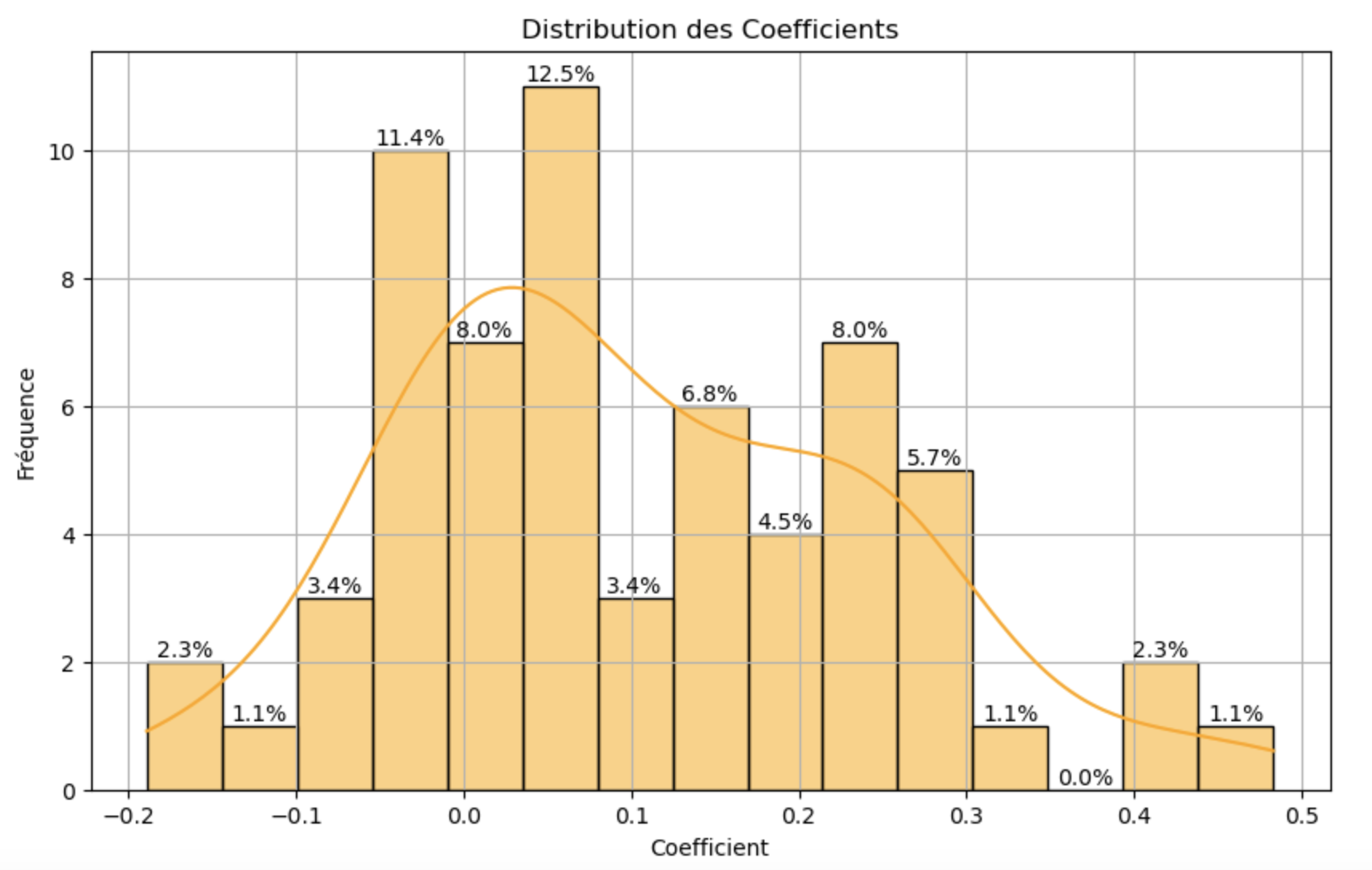


On constate que pour les 10 variables ayant obtenue la meilleure adhérence avec la variable cible sont toutes catégorielles. La méthode de gestion des missing values la plus efficace se retrouve systématiquement être l’affectation à la catégorie ‘-1’

***Flop 10 des variables les moins corrélées***



***Répartition des résultats statistiques***



Pour la suite, nous exclurons les variables ayant un indice de Cramer/coefficient de corrélation compris dans l’intervalle [-0.1 :0.1]. Ce seuil est positionné « à priori ». Il nous permet de conserver approximativement 50% des variables à disposition.

Dans celles retenues, certaines seront écartées malgré leur résultat de test statistique dès lors qu’il subsiste un risque de colinéarité.

Nous privilégierons dans ce cas (selon la même logique), la variable ayant une adhérence statistique la plus importante.

***Exception de ‘vehi\_catv’ et ‘agg\_catv\_perso’***

Ces deux variables que l’on retrouve dans le top 10 (encadrées en jaune) présentent toutes deux une adhérence importante avec la variable cible mais une corrélation forte l’une avec l’autre qui s’explique facilement par le fait que nous avons construit ‘agg\_catv\_perso’ par regroupement de modalités de ‘vehi\_catv’.

Dans la mesure où ‘vehi\_catv’ propose une adhérence (légèrement) plus forte que notre variable créée, nous devrions la privilégier pour la suite. Nous décidons exceptionnellement de ne pas respecter notre règle et ce afin de privilégier la variable ayant beaucoup moins de modalités (6 au lieu de 31) gardant à l’esprit que cette variable devra être encodée.

Sur cette base, nous obtenons le résultat suivant :

|  |  |  |  |
| --- | --- | --- | --- |
| Variable |  | Maintenue | RAISON |
| id\_xxxxx |  | NON | Identifiants |
| carac\_an |  | NON | DATE\_ACCIDENT (format int et non datetime) |
| carac\_mois |  | NON |
| CArac\_jour |  | NON |
| CARAC\_HRMN | Heure | OUI |  |
| CARAC\_ADR | Adresse | NON | String |
| CARAC\_AGG | Agglo 0/1 | OUI |  |
| CARAC\_ATM | Condition atmosphériques | NON | Faible corrélation |
| CARAC\_COL | Type de collision | OUI |  |
| carac\_COM | Commune | OUI |  |
| carac\_DEPT | Département | OUI |  |
| CARAC\_GPS | Indicateur localisation | NON | Donnée Obsolète |
| CARAC\_GPS\_LAT | Coordonnées GPS | NON | Faible corrélation |
| CARAC\_GPS\_LONG | Coordonnées GPS | NON | Faible corrélation |
| carac\_INT | Type intersection | NON | Faible corrélation |
| carac\_lum | Luminosité | OUI |  |
| LIEU\_CATR | Catégorie de route | OUI |  |
| LIEU\_CIRC | Régime circulation | OUI |  |
| LIEU\_ENV1 | Indicateur proximité école | NON | Donnée Obsolète |
| LIEU\_INFRA | Aménagement - infrastructure | NON | Faible corrélation |
| lieu\_LARROUT | Largeur chaussée | NON | Faible corrélation |
| LIEU\_LARTPC | Largeur terre-plein central | NON | Faible corrélation |
| LIEU\_NBV | Nombre de voies | NON | Faible corrélation |
| lieu\_PLAN | Tracé en plan | OUI |  |
| LIEU\_PR | Numéro du PR rattachement | NON | Faible corrélation |
| Lieu\_PR1 | Distance du PR rattachement | NON | Faible corrélation |
| LIEU\_PROF | Déclivité de la route | NON | Faible corrélation |
| LIEU\_SITU | Localisation (sur la chaussée) | OUI | Faible corrélation |
| lieu\_SURF | État de la surface | NON | Faible corrélation |
| LIEU\_V1 | Indice numérique no route | NON | Faible corrélation |
| LIEU\_V2 | Indice alpha no route | NON | Faible corrélation |
| LIEU\_VOIE | No de voie | NON | Faible corrélation |
| lieu\_VOSP | Voie réservée | NON | Faible corrélation |
| LIEU\_VMA | Vitesse maximale autorisée | NON | Faible corrélation |
| VEHI\_CATV | Catégorie de véhicule | NON | Multi colinéarité |
| AGG\_CATV\_PERSO | Catégorie personnalisée | OUI | Exception |
| VEHI\_CHOC | Point de choc initial | OUI |  |
| vehi\_manv | Manœuvre avant accident | OUI |  |
| vehi\_MOTOR | Motorisation | OUI |  |
| VEHI\_OBS | Obstacle fixe heurté | OUI |  |
| VEHI\_OBSM | Obstacle mobile heurté | OUI |  |
| VEHI\_OCCUTC | Nb passagers transport en C. | NON | Faible corrélation |
| vehi\_SENC | Sens de circulation | NON | Faible corrélation |
| user\_AN\_NAIS | Année de naissance | NON | Faible corrélation |
| USER\_CATU | Catégorie d’usager | OUI |  |
| user\_ACTP | Action du piéton | NON | Multi colinéarité |
| USER\_ETATP | Piéton seul ou accompagné | NON | Multi colinéarité |
| USER\_LOCP | Localisation du piéton | NON | Multi colinéarité |
| user\_PLACE | Emplacement dans véhicule | OUI |  |
| user\_SECU1 | Utilisation équipement sécu. | OUI |  |
| USER\_SECU2 | Utilisation équipement sécu. | OUI |  |
| USER\_SECU3 | Utilisation équipement sécu. | OUI |  |
| user\_SEXE | Genre de l’usager | NON | Multi colinéarité |
| user\_TRAJET | Type de trajet | OUI |  |
| AGG\_IS… | Véhicule est un … (vélo, etc.) | NON | Faible corrélation |
| AGG\_IS\_CONDUCteur… | Véhi. conduit par un h. | NON | Faible corrélation |
| AGG\_IS\_CONDUCtRICE… | Véhi. Conduit par une femme | NON | Faible corrélation |
| AGG\_NB\_... | Nb par catégorie d’usager | NON | Faible corrélation |
| AGG\_NB\_TOTAL\_VEHICULE | Nb total de véhicules | OUI |  |
| AGG\_NB\_TOTAL\_CONDUC… | Nb total de conductrices | NON | Faible corrélation |
|  |  |  |  |

Sur 81 variables, ce sont 23 variables qui se retrouvent sélectionnées. Attention, la majorité des variables catégorielles étant non ordonnées, le nombre de variables va significativement augmenter une fois l’encodage réalisé.

Nous nous réservons la possibilité de revenir sur cette pré sélection en fonction des contraintes et constatations qui pourraient ressortir plus tard.

# PARTIE 3 - Automatisation des opérations de préprocessing

## Déclaration des fonctions de transformation

Le principe retenu est d’écrire une classe (Tranformer) qui hérite des deux classes BaseEstimator et TransformerMixin afin de préparer une Classe générique qui sera en mesure :

* De gérer les missing values selon différentes règles pré définies (ne rien faire, remplacer par la modalité la plus récente, etc.)
* De gérer les outliers
* D’activer ou non un encoder (dichotommisation pandas, OneHotEncoder)
* D’activer ou non un scaler (StandardScaler, MinMaxScaler)
* De conserver ou supprimer la variable.

Cette classe générique sera utilisée (via héritage) pour créer autant de classes de transformation que de variables disponibles. Des valeurs par défaut seront définies pour chaque argument de manière à accéder par défaut au comportement attendu pour chacune des variables (activation d’un One hot encoder pour la variable département, maintenir la catégorie -1 pour les missing values de la variable infrastructure ou règle spécifique de gestion des missing values pour la variable vma.

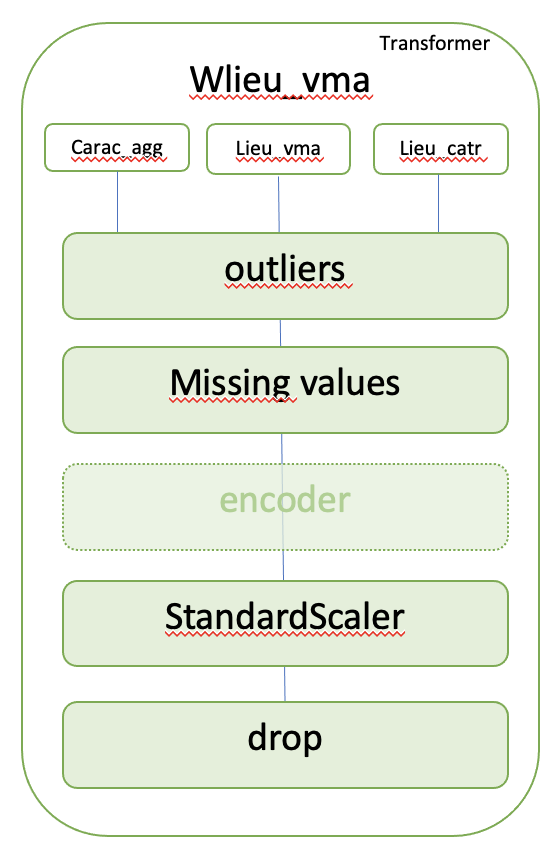
***Exemple transformation ‘lieu\_vma’***

Cette transformation a la particularité de surcharger les fonctions natives de la classe mère Transformer afin de gérer les spécificités de missing values et d’Outliers de cette variable tel que vu précédemment.

Autre particularité, cette variable n’a pas été retenue de par sa trop faible adhérence avec la variable cible. La valeur par défaut de drop sera positionnée à True afin de supprimer cette colonne.

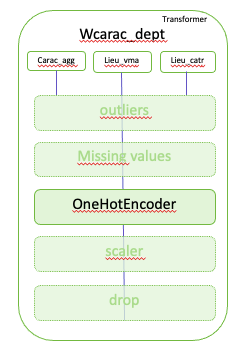
Nous serions en droit de remarquer le faible intérêt de surcharger les fonctions de gestion des missing values et d’Outliers dans la mesure où la variable sera écartée.

Ceci étant, nous avons paramétré le transformeur ‘Wlieu\_vma’ pour effectivement supprimer la variable mais si l’on souhaite valider cette hypothèse par des tests en conservant la variable, il nous suffira de modifier la valeur de l’argument drop en le passant de True à False pour conserver tel qu’il se doit la variable ‘lieu\_vma’.



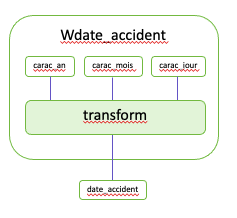
***Exemple transformation ‘carac\_dept’***

Cette transformation se résume à un simple encodage. Nous aurions pu utiliser un pipeline simple plutôt qu’une utilisation de notre classe Transformer mais l’utilisation de cette dernière nous permet de tester facilement différentes configurations : avec ou sans encoder, privilégier un scaler à l’encoder, ou simplement exclure la variable du périmètre de learning. Pour ce faire, il nous suffit de modifier les paramètres d’appel du pipeline.



***Exemple transformation ‘date\_accident’***

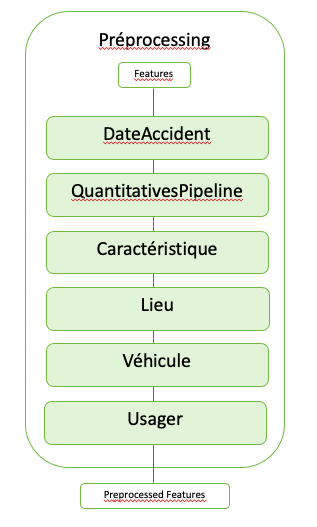
Construction d’une date au format datetime à partir des variables ‘carac\_an’, ‘carac\_mois’ et ‘carac\_jour’ puis suppression de ces colonnes pour ne conserver que ‘date\_accident’.



## Déclaration des pipelines de préprocessing

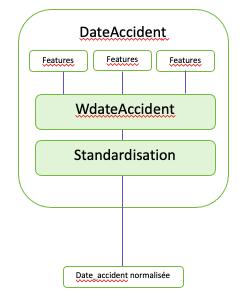
D’avoir privilégié l’utilisation de classe de transformation nous permet de leur donner vie au sein d’une pipeline de préprocessing. Ce pipeline sera composé de 6 étapes :

* Recomposition de date\_accident
* Préparation des données Quantitatives
* Préparation des données Qualitatives de Caractéristique
* Préparation des données Qualitatives de Lieu
* Préparation des données Qualitatives de Véhicule
* Préparation des données Qualitatives d’Usager

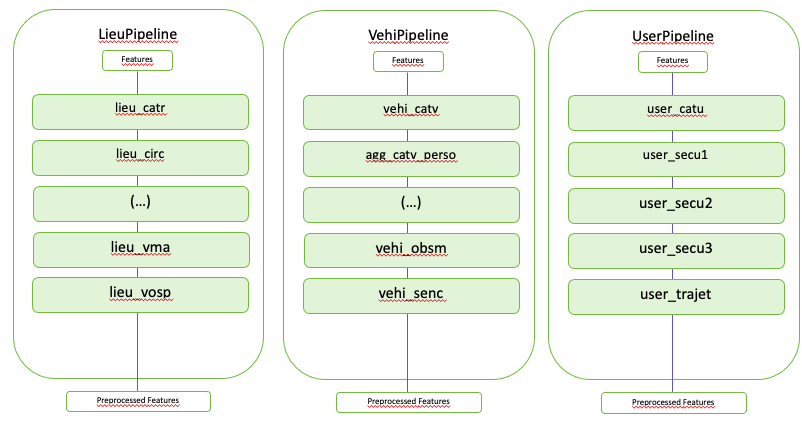


***Etape 1 – DateAccident***

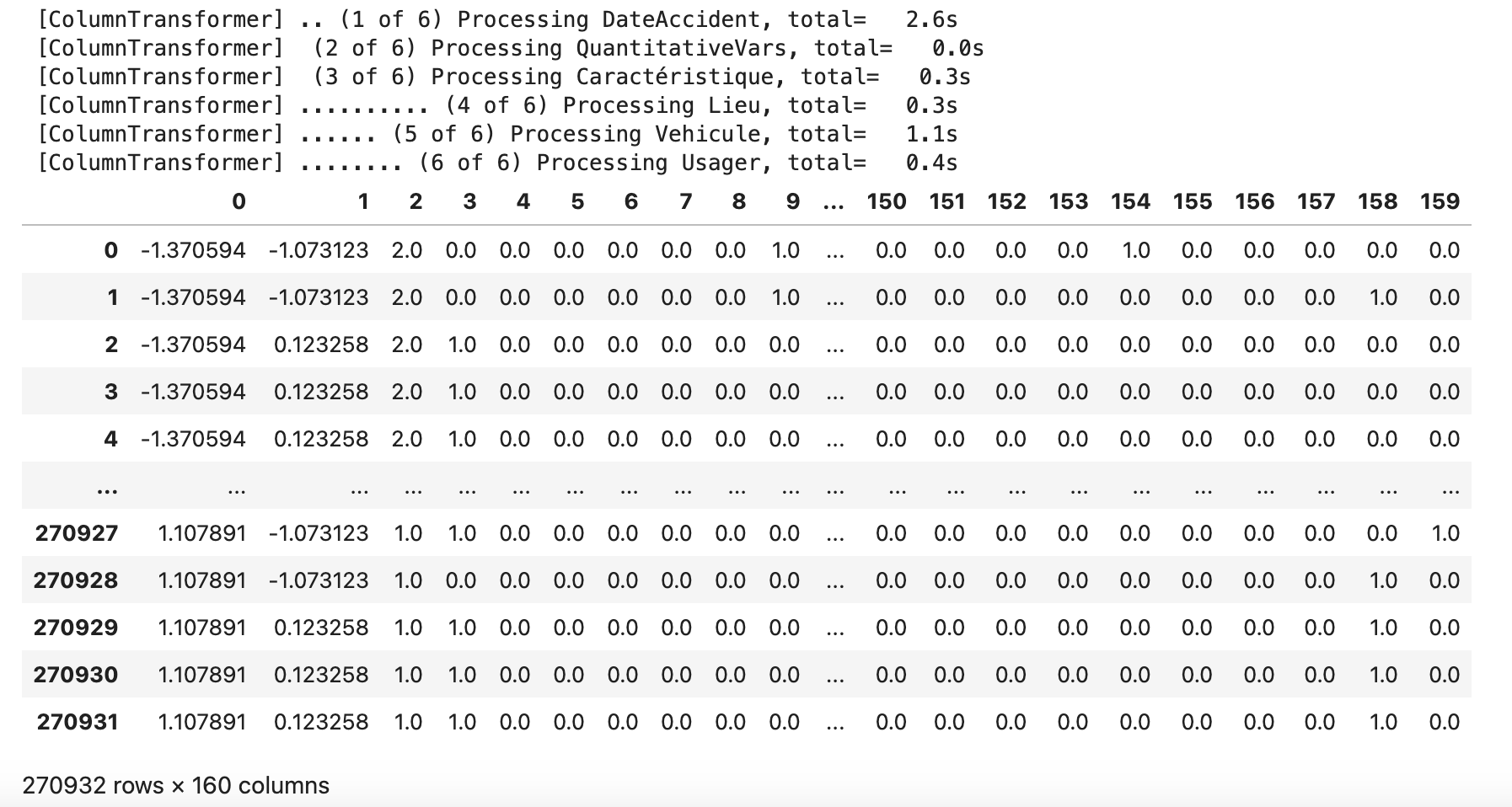
Ce pipeline va, à partir des données ‘carac\_an’, ‘carac\_mois’ et ‘carac\_jour’ reconstituer une variable construite comme suit : AAAAMMJJ ou AAAA= année, MM=mois, JJ=jour et standardiser sa valeur.



***Etape 2 à 6 – Traitement des variables qualitatives et quantitatives***



***Exécution pipeline préprocessing***



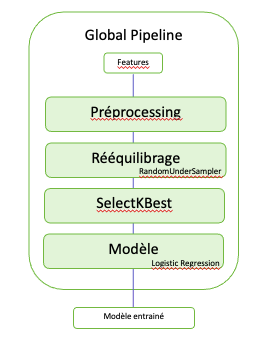
*Inconvénients de l’utilisation du make\_column\_transformer, l’objet retourné est de type numpy.ndarray ce qui implique la perte des noms de colonnes.*

Le pipeline s’exécute en fit et transform sur l’échantillon d’entrainement (270932 enregistrements) en quelques secondes.

## Déclaration du pipeline global

Dernière étape, lancer en automatique à la suite de la phase de préprocessing les étapes nécessaires à l’apprentissage du modèle.

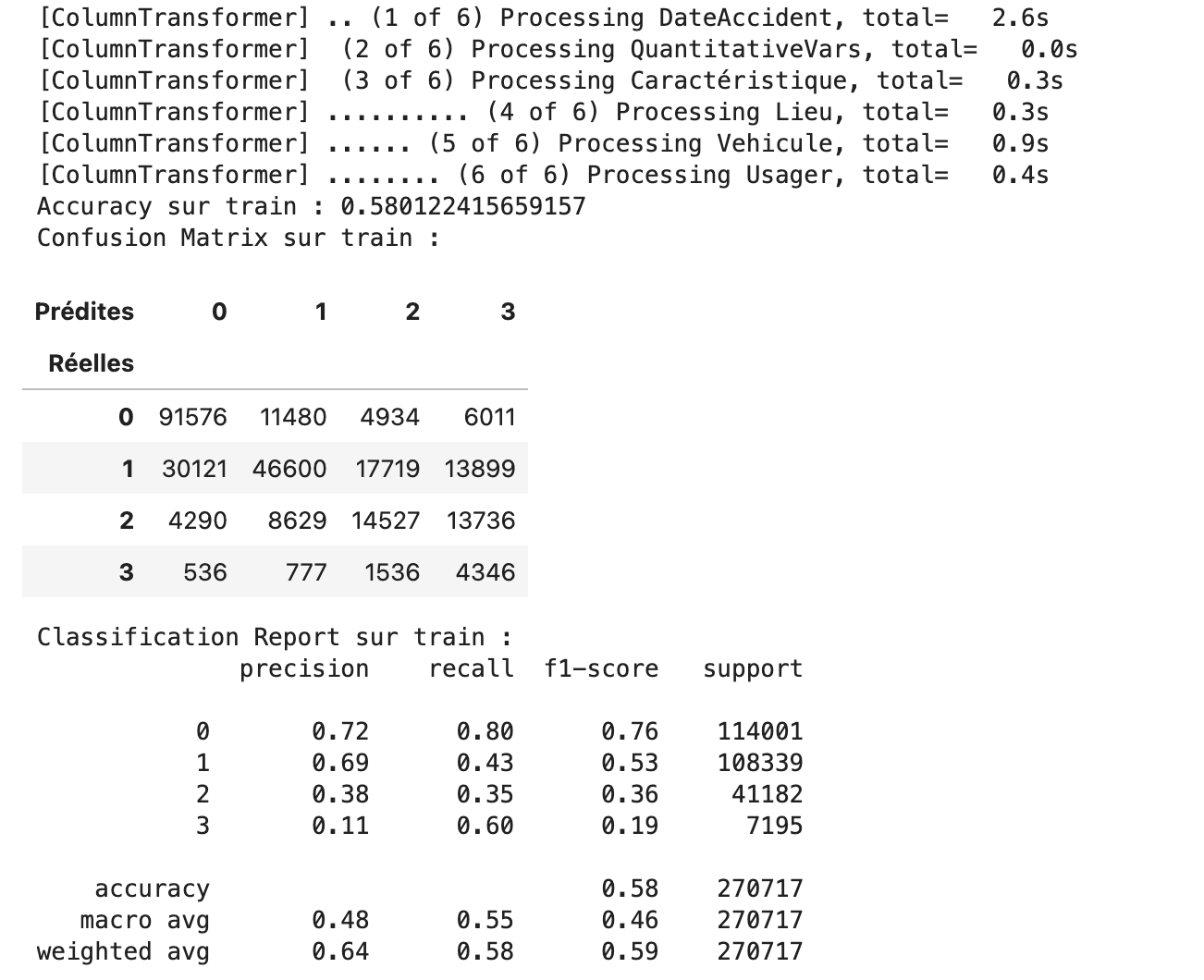
* Rééquilibrage de l’échantillon d’entrainement
* SelectKBest
* L’apprentissage de notre premier modèle : nous commencerons avec un modèle de régression logistique.



***Première exécution du pipeline***

Sur la base de l’ensemble des décisions que nous avons pu prendre jusque-là, nous lançons notre première phase d’apprentissage.

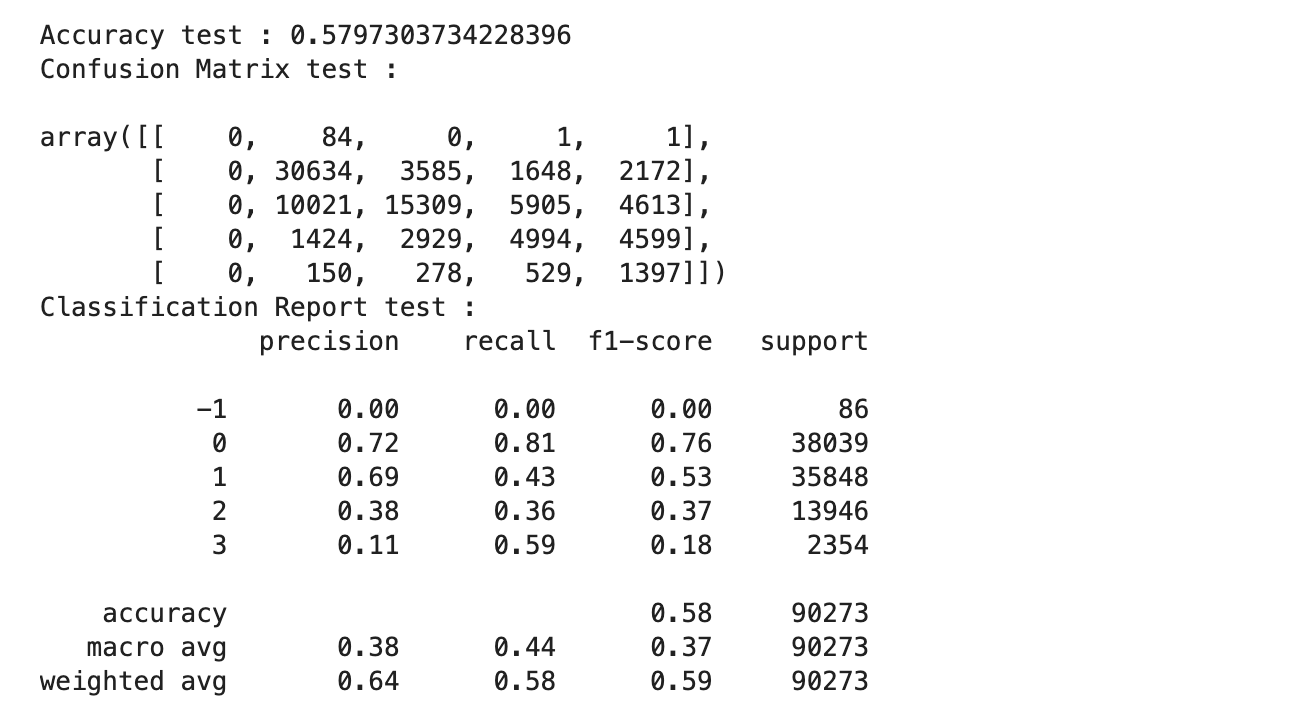
Il s’agit d’une première exécution (sans Cross validation) à partir d’un rééquilibrage de type Random Under Sampler et une régression logistique (hyperparamètres par défaut).



Les premiers résultats sont perfectibles : nous avons une accuracy d’à peine 0.58% ce qui n’est pas énorme. Sans surprise, les scores sont meilleurs pour les deux modalités les plus représentées (« indemne » et « blessé léger »).

La précision sur la modalité « tué » est très faible (11%) ce qui signifie qu’il détecte beaucoup trop souvent des « décès » à tort (faux positifs). Par contre, nous pouvons noter un recall qui est meilleur (60%). En clair, il détecte mieux les vrais positifs mais avec un taux d’erreur sur les faux.

***Évaluation de notre modèle sur test***



Les résultats sont similaires sur test et sur train. C’est peut-être un signe d’underfitting : le modèle ne semble pas avoir identifié de relation pertinente lui permettant de bien étiqueter la cible.

## Exécution de notre pipeline via un GridSearchCV

A partir de maintenant nous lancerons notre pipeline à partir d’un GridSearchCV afin :

* De pouvoir tuner nos hyperparamètres de préprocessing
* De pouvoir tuner l’algo de rééquilibrage ainsi que ses hyperparamètres
* De pouvoir tuner les hyperparamètres du SelectKBest
* De pouvoir essayer plusieurs modèles de ML

Pour commencer ce test, nous allons revenir sur la question de préprocessing que nous nous sommes posés en lien avec des colonnes ‘carac\_com’ et ‘carac\_dept’ :

* Incidence d’un one hot encoding sur des variables à fortes modalités (performances du pipeline et sur le score final)

Nous paramétrons notre Grid Search de manière à tester cette incidence lorsque :

* L’on exclue les deux variables
* On en exclue une alors que l’autre sera avec ou sans one hot encoding
* Pour les deux variables, on n’encode pas mais on normalise les données

Cela nous donne un paramétrage de ce type :

# Test paramétrage de com/dept

params\_com\_dept = [

{

# Suppression des 2 colonnes

'preprocessor\_\_Caractéristique\_\_wcarac\_com\_\_dropC' : [True],

'preprocessor\_\_Caractéristique\_\_wcarac\_dept\_\_dropC' : [True],

"Selector\_\_k" : ['all']

},

{

# Conservation que de com en encodage O/N

'preprocessor\_\_Caractéristique\_\_wcarac\_com\_\_encoder': [None, 'OneHotEncoder'],

'preprocessor\_\_Caractéristique\_\_wcarac\_com\_\_scaler' : [None],

'preprocessor\_\_Caractéristique\_\_wcarac\_com\_\_dropC' : [False],

'preprocessor\_\_Caractéristique\_\_wcarac\_dept\_\_dropC' : [True],

"Selector\_\_k" : ['all']

},

{

# Conservation que de dept en encodage O/N

'preprocessor\_\_Caractéristique\_\_wcarac\_com\_\_dropC' : [True],

'preprocessor\_\_Caractéristique\_\_wcarac\_dept\_\_encoder' : [None, 'OneHotEncoder'],

'preprocessor\_\_Caractéristique\_\_wcarac\_dept\_\_scaler' : [None],

'preprocessor\_\_Caractéristique\_\_wcarac\_dept\_\_dropC' : [False],

"Selector\_\_k" : ['all']

},

{

# Conservation des 2 colonnes en encodage O/N

'preprocessor\_\_Caractéristique\_\_wcarac\_com\_\_encoder' : [None, 'OneHotEncoder'],

'preprocessor\_\_Caractéristique\_\_wcarac\_com\_\_scaler' : [None],

'preprocessor\_\_Caractéristique\_\_wcarac\_com\_\_dropC' : [False],

'preprocessor\_\_Caractéristique\_\_wcarac\_dept\_\_encoder' : [None, 'OneHotEncoder'],

'preprocessor\_\_Caractéristique\_\_wcarac\_dept\_\_scaler' : [None],

'preprocessor\_\_Caractéristique\_\_wcarac\_dept\_\_dropC' : [False],

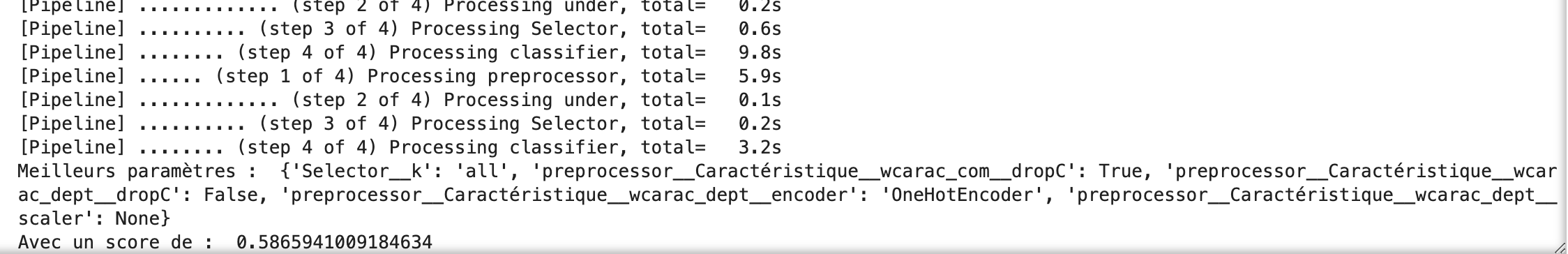
"Selector\_\_k" : ['all']

}

]

***Résultats obtenus***

En utilisant une Cross Validation=5, nous constatons (pour un modèle de régression logistique) une situation où, il semble préférable de privilégier la variable « département » et éliminer la variable « commune ». Cela va dans le sens des tests statistiques réalisés précédemment où le département obtenait un indice de Cramer de 0.285, légèrement plus élevé que celui de la variable « commune » (0.259).



Pour la suite des tests, nous privilégierons cette configuration.

## Fin étape automatisation du préprocessing

En l’état, nous sommes outillés pour améliorer notre modèle. Plusieurs pistes pour la suite du projet :

* Tuning de nos règles de préprocessing
* Tuning des hyperparamètres (cross validation, rééquilibrage, modèle de ML, etc.)
* Ajustement du seuil de décision
* Utilisation de modèles ensemble (Random Forest, boosting/Bagging)
* Réduction de dimension
* Etc.