Rapport mi-projet
Fairness pour l'IA

Erwan LEMATTRE, Yannis CHUPIN

7 avril 2024

Table des matières

1 Introduction

Ce projet a pour objectif d'analyser les accidents de la circulation routière afin de pouvoir déterminer à partir des données d'un véhicule accidenté si l'accident est mortel ou non. Les données sont des données libres mises à disposition par le *Ministère de l'Intérieur et des Outre-Mer*. Le jeu de données correspond aux accidents de 2005 à 2022 en France. Nous allons, dans une première partie, analyser ces données afin d'extraire les informations utiles à l'apprentissage. Puis nous essaierons de repérer d'éventuelles sources de biais affectant notre modèle.

Vous pouvez retrouver le code sur le GitHub du projet. Le fichier `main.ipynb` contient le code principal que nous allons suivre et contextualiser tout au long de ce rapport. de plus, le fichier `utils.py` contient toutes les fonctions auxiliaires que nous utilisons dans le fichier principal.

Première partie

Analyse

2 Découverte du jeu de données

2.1 La base de données

La base de données est composée de plusieurs tables : *usagers*, *véhicules*, *lieux* et *caractéristiques*. Nous avons joint ces quatre parties pour obtenir un dataframe contenant une cinquantaine de colonnes. On peut retrouver une partie des attributs dans l'annexe B.

2.2 Répartition des données

Afin de pouvoir conserver les données utiles pour l'apprentissage, nous avons analysé la répartition des différentes données dans notre dataframe. Nous avons ainsi pu faire différentes observations.
Voici quelques-unes d'entre elles qui nous sont ensuite utiles pour la préparation des données.

2.2.1 Catégories de véhicule

La base de données contient beaucoup de types de véhicules différents. Nous avons cependant pu remarquer que la majorité des véhicules sont dans seulement 5 catégories.

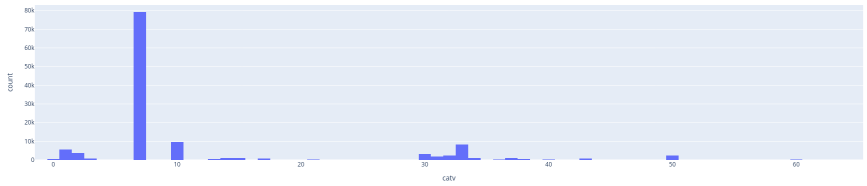


FIGURE 1 – Répartition des catégories de véhicules

2.2.2 Gravité de l'accident

En affichant l'effectif d'individus décédés dans un accident, nous avons pu remarquer qu'ils ne représentent qu'une infime partie des individus accidentés. Leur proportion est si fable que ça ne nous permet pas d'apprendre un modèle. C'est la raison pour laquelle nous avons décidé de nous intéresser non pas à la mortalité à l'échelle d'une personne, mais plutôt à l'échelle d'un accident. Nous nous mettons pour cela au niveau d'un véhicule car cela nous permet de conserver plus d'informations (à l'échelle d'un accident, on aurait dû enlever trop d'informations pour ne conserver que les attributs plus généraux à l'accident).

3 Préparation des données

À partir des observations précédentes, nous avons supprimé les attributs moins intéressants pour l'apprentissage et nous avons modifié certains attributs afin d'en extraire les informations inté-

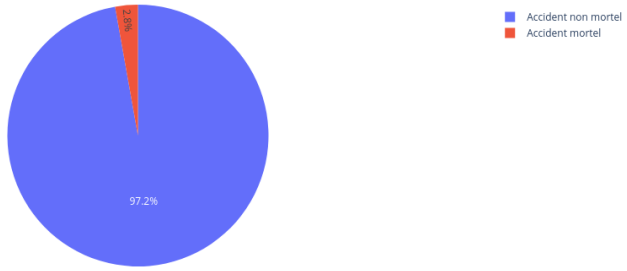


FIGURE 2 – Proportion d'accidents mortels

ressantes.
Les attributs supprimés sont : *voie*, *v1*, *v2*, *pr*, *pr1*, *lartpc*, *larrout*, *num_veh*, *occutc*, *adr*, *senc*, *etatp*, *actp*, *manv*, *jour*, *com*, *hrmn*, *motor*, *place*, *vosp*, *locp*.

Nous avons effectué les modifications suivantes :

- Création d'un attribut *mortal* qui vaut 1 si le véhicule est impliqué dans un accident mortel, 0 sinon.
- À partir de l'attribut *sexe*, nous avons créé un attribut *sexe_conducteur* qui garde seulement le sexe du conducteur du véhicule.
- Création d'un attribut *piéton* qui vaut 1 si un piéton est impliqué dans l'accident, sinon 0.
- Nous avons utilisé l'année de naissance et l'année de l'accident pour récupérer l'âge du conducteur.
- L'attribut *vma* a été découpé en 4 catégories de vitesse.
- Pour les attributs *catv* et *catr*, nous avons gardé les valeurs les plus représentées dans la base de données.

Nous avons également réduit les valeurs de certains attributs. Par exemple, pour des attributs avec des valeurs telles que *Non-renseigné*, *Autre*, ... nous avons regroupé ces valeurs en une seule valeur. L'objectif était ici de simplifier en réduisant les catégories mais également d'améliorer les performances de notre modèle.

4 Analyse des données

Une fois nos données préparées, nous avons pu les visualiser. Nous allons montrer dans les deux prochaines parties les observations intéressantes que nous avons pu faire lors de l'analyse de notre dataset.

4.1 Analyse univariée des données

4.1.1 Les accidents mortels

Une donnée intéressante à observer est la proportion de véhicules impliqués dans un accident mortel. C'est en effet la valeur que nous voulons prédire.

FIGURE 3 – Proportion des véhicules impliqués dans un accident mortel

Nous pouvons remarquer sur la figure 3 que le fait de s'intéresser aux véhicules impliqués dans un accident mortel et non plus aux personnes nous permet de doubler ce pourcentage. Même si cette proportion reste faible, cela va nous permettre d'avoir plus de données dans la catégorie mortelle lors de l'apprentissage et par conséquent d'avoir un meilleur modèle.

4.1.2 Les piétons

Nous nous sommes ensuite intéressés aux accidents dans lesquels un piéton est impliqué. La figure 4 nous montre qu'un peu moins de 10% des accidents impliquent un piéton.

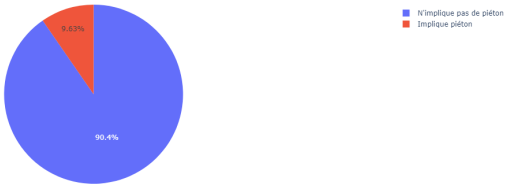


FIGURE 4 – Proportion des accidents avec piéton

4.1.3 L'âge

Nous pouvons visualiser l'âge des conducteurs via une boîte à moustache. La figure 5 nous montre la répartition de l'âge des conducteurs. Lors du prétraitement des données, les valeurs aberrantes ont été enlevées. On retrouve donc logiquement des âges contenus entre 0 et 100 ans. L'âge médian des conducteurs est 33 ans avec le premier quartile à 21 et le troisième quartile à 49 ans. Même si on peut imaginer que des valeurs sont fausses (il y a des conducteurs de moins de 16 ans), les valeurs sont tout de même assez cohérentes par rapport à ce que l'on pourrait imaginer de la répartition de l'âge des conducteurs.

4.1.4 Le genre des conducteurs

Le genre des conducteurs est assez intéressant à analyser. Sur la figure 6, nous pouvons remarquer une différence importante entre le nombre de femmes au volant d'un véhicule ayant eu un accident (indice 0) et le nombre d'hommes (indice 1). Cette différence pourrait être une source de biais pour notre modèle. En effet, le fait qu'il y ait beaucoup plus de données d'accident avec des hommes ne signifie pas qu'il y a plus de chances d'avoir un accident si on est un homme.

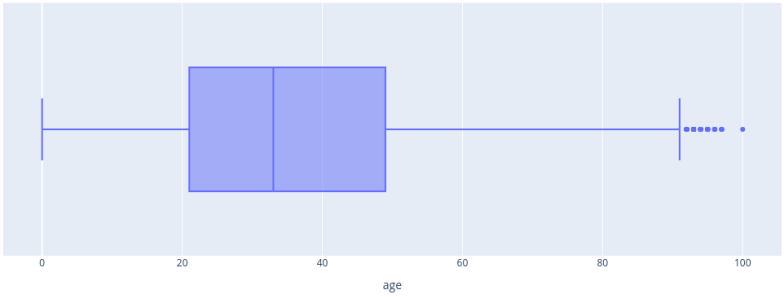


FIGURE 5 – Âge des conducteurs

Cela signifie peut-être que la proportion d'hommes au volant est plus élevée et donc qu'il y a plus d'accidents avec un homme au volant car il y a plus d'hommes au volant. Le risque ici est que moins de données avec des femmes conduisent à des prédictions plus marquées pour les femmes et par conséquent à des potentiels biais.

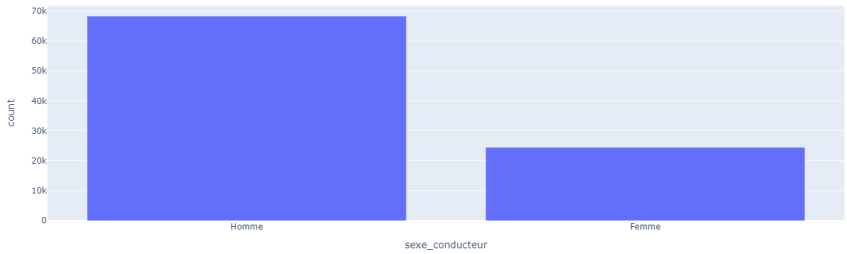


FIGURE 6 – Genre des conducteurs

4.1.5 Le type de collision

La figure 7 montre la répartition des différents types de collisions dans notre dataset. On peut remarquer que tous les types de collisions sont plutôt bien représentés dans notre dataset. C'est un attribut qui pourra être assez intéressant pour l'apprentissage.

4.2 Analyse bivariée des données

Nous allons dans cette partie donner quelques exemples intéressants obtenus lors de l'analyse bivariée. On peut retrouver l'ensemble des graphiques observés dans le fichier `main.ipynb`.

4.2.1 Le genre du conducteur

Un attribut qu'il est intéressant d'analyser est le genre du conducteur. En effet, il peut être source de biais s'il y a un déséquilibre entre hommes et femmes. On retrouve globalement la même proportion dans la corrélation que l'on soit homme ou femme. Les hommes étant beaucoup

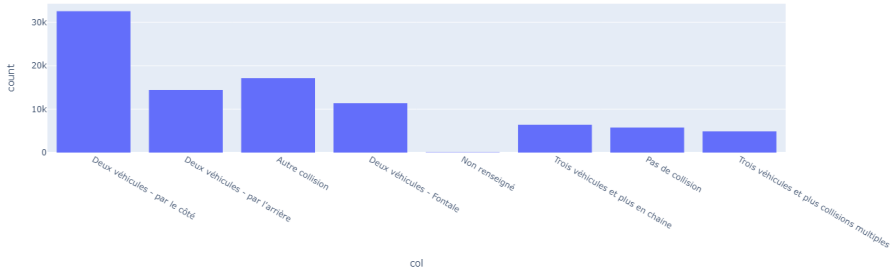
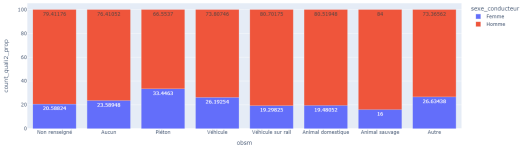


FIGURE 7 – Les types de collision

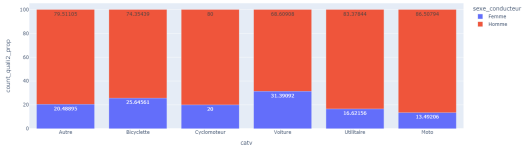
plus représentés dans le dataset, la proportion d'hommes est logiquement plus élevée. On peut cependant faire quelques remarques.

La figure 8a nous montre une proportion de femmes moins élevée quand *obsm* vaut *6*. La proportion de femmes est deux fois plus élevée quand *obsm* vaut *1*. Ceci pourrait biaiser notre modèle.

Sur la figure 8b, on remarque également une proportion différente de femmes en fonction du type de véhicule. On pourrait expliquer cela par le fait que certains véhicules sont dans la réalité plus utilisés par les hommes, par exemple on pourrait imaginer qu'il y a plus d'hommes qui conduisent des motos. Il faudra être vigilant car cela peut être source de biais. Il se peut que notre modèle associe une moto à un homme. Dans le cas d'une femme sur une moto le résultat pourrait être soit forcément un accident mortel, ou bien forcément un accident non mortel.



(a) Proportion femmes/hommes en fonction de la présence d'un obstacle dans l'accident



(b) Proportion femmes/hommes en fonction de la catégorie du véhicule

4.2.2 L'âge du conducteur

On peut remarquer sur la figure 9 que la probabilité d'accident mortel est beaucoup plus élevée à l'âge de 21 ans. Cela est dû en partie au fait que les conducteurs de 21 ans sont surreprésentés. Cette probabilité risque de poser problème pour notre modèle. En effet l'âge n'est pas un paramètre déterminant dans l'évaluation de la gravité de l'accident. Le problème est que notre modèle va probablement associer l'âge de 21 ans à un accident mortel, peu importe les autres paramètres de l'accident. L'âge serait donc notre **attribut sensible**.

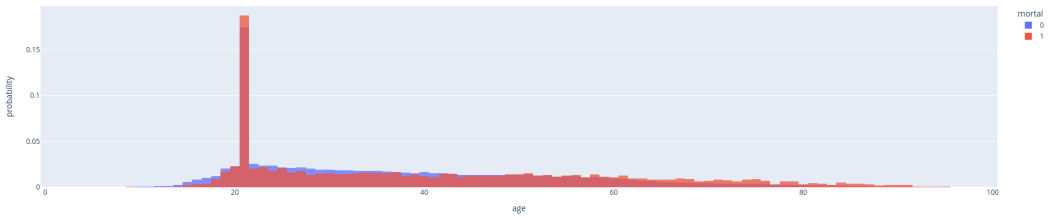


FIGURE 9 – Analyse bivariée âge – accident mortel

5 Apprentissage

5.1 Split

Séparer nos données en deux ensembles demandait de tenir compte d'une spécificité. Nous nous plaçons du point de vue d'un véhicule, mais bien souvent, plusieurs véhicules sont impliqués dans un même accident. Il nous a donc fallu adapter le split pour que les véhicules d'un même accident soient dans le même ensemble.

Par ailleurs, nous avons constaté que, si on ne le faisait pas, un surapprentissage lors du test se produisait. Typiquement, un trop grand nombre de morts étaient correctement prédits. Mais le modèle trouvait une partie de ces véhicules en les associant à ceux avec lesquels ils avaient eu un accident dans l'ensemble d'apprentissage.

5.2 Encodage One Hot

La base de données nous fournit, pour la plupart des attributs, des données qui peuvent être converties en entiers. Par exemple, une catégorie de véhicule (*catv*) est désignée par un chiffre. L'attribut reste cependant catégoriel. En pratique, les seuls attributs quantitatifs qui nous restent sont l'âge du conducteur, les mois et les départements. Le reste est soit binaire, soit qualitatif.

Tous les attributs qualitatifs ont donc été transformés en OneHot. Initialement, nous souhaitions optimiser le nombre de colonnes en réalisant les OneHot nous-mêmes. Par exemple, prenons "surf" (l'état de la surface de la route). Il est inutile de créer une colonne pour "Non-renseigné", "Autre" ou "normale". Nous sommes uniquement intéressés par les états spécifiques de la route.

Cependant, cette approche posait des problèmes car elle empêchait la réalisation de l'audit. Nous avons donc utilisé la méthode standard pour nos OneHot.

5.3 Arbre de décision

Une fois les données prêtes, nous avons utilisé le classifieur par arbres de décision pour obtenir un modèle. C'est le premier que nous avons utilisé. En ce qui concerne les résultats, le score d'accuracy est de *1* pour l'entraînement et de *0.9014* pour le test.

En ce qui concerne la matrice de confusion, on remarque que *27687* véhicules ont été classés correctement en tant qu'accidents non létaux. Tandis que *291* accidents mortels seulement ont été correctement identifiés. Cependant, *1413* accidents mortels ont été recensés à tort comme étant non létaux et *1648* véhicules ont subi le sort inverse.

FIGURE 10 – La matrice de confusion pour le classifieur d'arbres aléatoires

En résumé, on remarque que le modèle réussit très bien à classifier les accidents qui n'ont pas conduit à la mort de quelqu'un. Cependant, il lui apparaît bien plus difficile de trouver les véhicules impliqués dans des accidents mortels. On peut attribuer ceci au fait que ces accidents létaux ne représentent que *5.38%* des véhicules accidentés répertoriés dans la base.

5.4 GaussianNB

Avancés dans le projet, nous avons voulu tester plusieurs classifieurs, tous ont donnés des résultats différents, pour certains médiocres. Mais nous avons trouvé un autre classifieur qui donnait des résultats intéressants : GaussianNB.

Il ne maximise pas notre accuracy score puisqu'il n'est que de *0.8734* à l'entraînement et de *0.8677* pour les tests. Ce qui a toutefois retenu notre attention, c'est le nombre d'accidents létaux qu'il parvient à détecter : *508*. Réduisant ainsi l'erreur associée (les accidents mortels recensés à tort comme étant non létaux) à seulement *1196*. Ceci s'accompagne malheureusement une perte de performances pour ce qui est de la détection d'accidents non mortels comme on peut le voir ci-dessous.

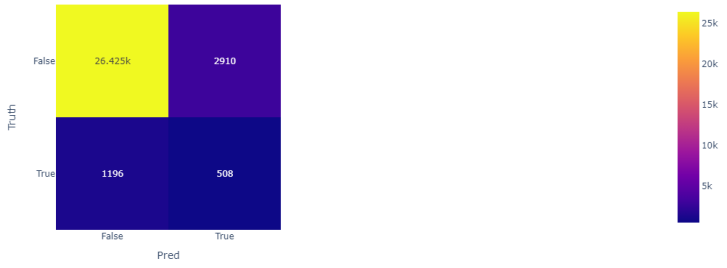


FIGURE 11 – La matrice de confusion avec GaussianNB

Un autre avantage qui peut ressembler à un inconvénient au départ est que ce classifieur demande des nombres et non des OneHot. Nos valeurs au départ sont en format numérique, rendant donc nos données compatibles avec cette approche. Ceci nous ouvre alors la possibilité de tester d'autres choses puisque le classifieur donne des probabilités et non des arbres. Au premier rang desquels *xplique*.

Par souci de concision cependant, nous prendrons le modèle issu du *tree classifier* pour tous les audits qui vont suivre, puisque c'est le premier avec lequel nous avons travaillé. On notera tout de même que lorsqu'on effectue ces audits avec *GaussianNB*, les résultats diffèrent en bien des points de ceux observés avec le *tree classifier*, preuve qu'ils ont une approche bien différente.

5.5 Base Rate

Nous avons voulu mesurer les résultats méritant potentiellement une protection : le fait qu'un piéton soit impliqué dans un accident et le sexe du conducteur. Voici les résultats :

- **Sexe du conducteur** 1 (un homme)
- **Disparate Impact** : 1.21767750298587 1.6331199049428085
- **P-rule disparate Impact** : 0.8212355057458954 0.6123249107266375
- **Démographie Parité** : 0.011747835864606336 0.02383701239031672

Ce qui est flagrant ici, c'est qu'il existe une disparité dans la prédiction d'accidents mortels pour les hommes. Allant par ailleurs dans le sens de ce qu'indique Disparate Impact par la suite. Notons cependant que la disparité démographique entre hommes et femmes existe mais est très faible.

- **Piéton** 1 (un piéton est impliqué)
- **Disparate Impact** : 1.0538191011165416 1.0968113720110815
- **P-rule Disparate Impact** : 0.9489294689576994 0.9117337999207913
- **Démographie Parité** : 0.003345154811069756 0.005266913173450724

Pour ce qui est de l'implication d'un piéton dans un accident, on constate ici aussi une légère disparité, soulignant leur plus grande implication dans des accidents mortels. Ceci concorde encore une fois avec le *disparate Impact*. Démographiquement, le cas est similaire au sexe, puisqu'on constate une légère disparité en leur faveur, bien que cette fois la disparité soit plus faible encore.

6 Audit du modèle

6.1 Génération des contrefactuels avec Dice

Dice nous a permis d'établir quels attributs influent sur la prédiction de notre modèle. Les résultats sont cependant assez variables d'une exécution à l'autre. On peut tout de même retrouver lesquels sont fréquemment impliqués dans le changement des exemples contrefactuels. Parmi eux, on peut noter que l'attribut *âge* est souvent représenté. Le taux d'accidents (notamment mortels) étant plus élevé chez les jeunes et les personnes âgées, cela paraît assez cohérent. L'attribut *obsm* est également souvent utilisé dans les exemples contrefactuels. Nous avons pu remarquer que l'attribut *col* est peu représenté dans les résultats, contrairement à l'hypothèse que nous avions pu faire lors de l'analyse.

Enfin, les attributs *dep* et *sexe_conducteur* sont également représentés. Ils pourraient être sources de biais, notamment l'attribut *dep* ; le modèle pourrait associer un département à une prédiction d'accident mortel.

6.2 BlackBoxAuditing

Les résultats que nous allons analyser sont issus du modèle généré par *tree classifier*.

Tout d'abord, l'audit a porté sur les 24 features que nous avons conservé afin d'élaborer ce modèle. On constate tout d'abord le rôle prédominant de l'âge pour l'accuracy score. À *0.84*,

ce dernier éclipse tous les autres. Une explication pourrait venir de ce que nous avons constaté dans notre analyse univariée : il y a une surreprésentation des jeunes de 20–21 ans dans ce set. Ceci est en réalité loin de surprendre puisqu'il s'aligne avec la politique de prix pratiquée par les assureurs envers les jeunes conducteurs.

Par la suite, on peut s'intéresser à l'évolution de l'accuracy en fonction du niveau de réparation appliqué à chaque attribut :

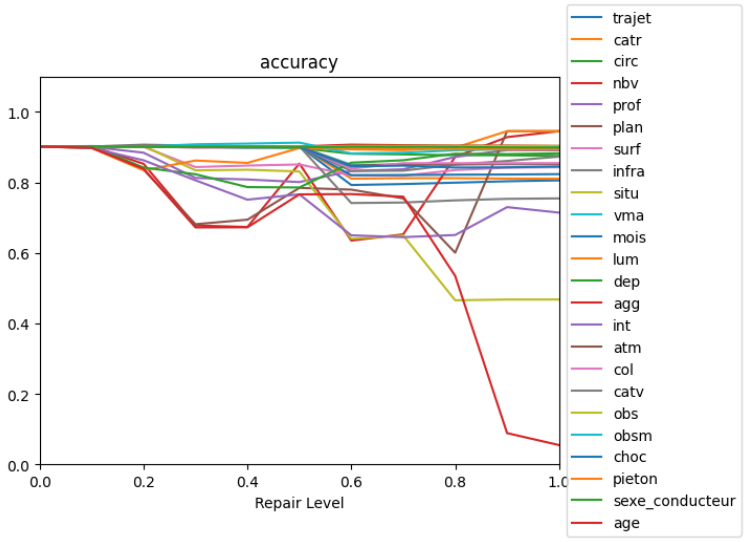


FIGURE 12 – L'accuracy en fonction du niveau de réparation pour chaque attribut.

On reconnaît toput de suite les attributs dont l'accuracy est au plus haut en regardant ceux qui s'effondrent le plus. En premier lieu, l'âge s'illustre à nouveau. À la lumière de ceci, il apparaît évident que le modèle discrimine fortement les véhicules en fonction de l'âge du conducteur. Dans une moindre mesure, on constate la même chose pour *situ*, qui fait référence à la localisation géographique de l'accident. Ce qui paraît sensé, étant donné que certains lieux sont plus dangereux que d'autres. Enfin, on observe que le type d'intersection et la catégorie du véhicule sont également sources de discriminations de la part de notre modèle. Encore une fois, cela semble cohérent.

Toutefois, notons l'absence de quelques attributs notables tels que le nombre de voies, le sexe du conducteur (chose intéressante : ce n'est pas le cas avec GaussianNB), le type de collision, le type d'obstacle heurté ou encore le fait qu'un piéton soit impliqué.

Maintenant regardons l'équilibre du modèle à l'aide du BCR.

On remarque ici que les attributs sont plutôt centrés autour de *0.5* et que la plupart reste rectiligne, mais que certains divergent quelque peu. Cet évasement est dû à des attributs déjà connus tels que *âge* ou *catv*. D'autres influent aussi mais ne s'étaient pas illustrés précédemment, à l'image du nombre de voies (*nbv*), du profil de la route (*plan*) ainsi que de la luminosité (*lum*).

Malgré cela, l'évasement assez faible laisse penser que le modèle est plutôt équilibré. De plus, comme mentionné précédemment dans la section dédiée au split, nous avons pu tester le cas où l'on acceptait que des véhicules impliqués dans le même accident se retrouvent dans des ensembles différents (entraînement / test). Ce qu'il ressortait alors du BCR était un évasement

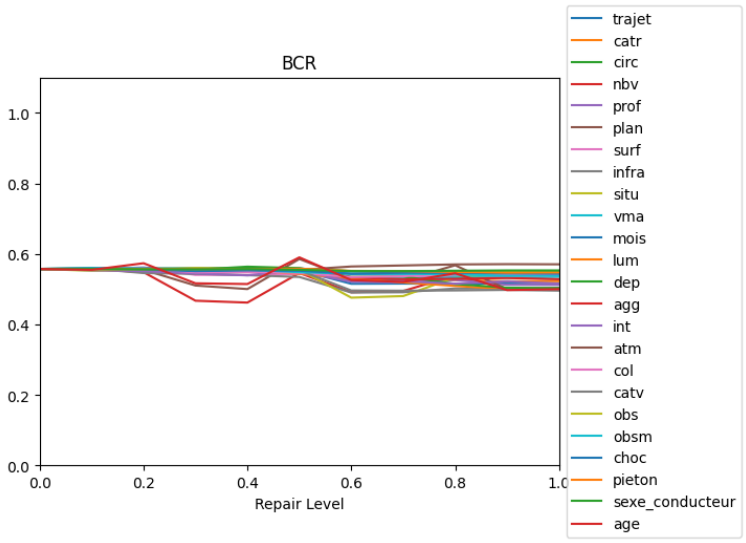


FIGURE 13 – Le BCR en fonction du niveau de réparation pour chaque attribut.

beaucoup plus important et une origine située vers 0.68. Ce qui laisse à penser que cette version était bel et bien trop modelée sur le train set, l'équilibre était rompu.

Pour conclure, BlackBoxAuditing nous a permis de comprendre quels étaient les attributs les plus discriminés. Âge est certainement le plus notable.

6.3 Expliquer le modèle avec les valeurs de Shapley

Afin d'analyser la contribution des différents attributs dans notre modèle, nous avons utilisé les valeurs de Shapley. Nous avons d'abord essayé de calculer la valeur exacte avec la fonction `ShapleyValues` du module *ShapKit*. Cependant, le nombre d'attributs de notre dataset est trop important et le calcul trop long. Il a donc fallu calculer une approximation de la valeur de Shapley avec la fonction `MonteCarloShapley`. Nous devons cependant être prudents car ce calcul de l'approximation nous donne une tendance pour notre modèle mais certainement pas une valeur exacte. On peut le constater en changeant le nombre d'itérations, qui produit souvent un résultat différent. Toutes les approximations ont ici été calculées avec `n_iter=1000`.

Le temps de calcul étant encore assez long nous avons parallélisé le calcul des valeurs de Shapley. Les calculs ont ensuite été séparés en deux, d'un côté pour les prédictions qui donnent 1 et de l'autre pour celles qui donnent 0. ON peut ainsi avoir une idée des attributs décisifs dans les deux cas.

Ce qui ressort des résultats obtenus est que les attributs *col*, *choc*, *obsm*, *int* et *agg* contribuent souvent au résultat. Ce résultat paraît plutôt cohérent. En revanche, on retrouve également assez souvent les attributs *mois*, *âge* et *dep*. Ces attributs ne devraient pas contribuer autant à la sortie de notre modèle. Il peuvent constituer un biais (ce n'est pas parce qu'on a un accident à l'âge de 21 ans que l'on va forcément mourir, d'autres attributs sont beaucoup plus importants).

Pour l'attribut *sexe_ conducteur* on nous donne une contribution nulle de cet attribut. Cela peut nous encourager dans l'idée que le sexe n'est pas un biais pour notre modèle conrairement à ce que nous pouvions penser au départ.

Ci-dessous un résultat obtenu du calcul des valeurs de Shapley. Vous pouvons retrouver un échantillon des résultats dans l'annexe ? de ce rapport.

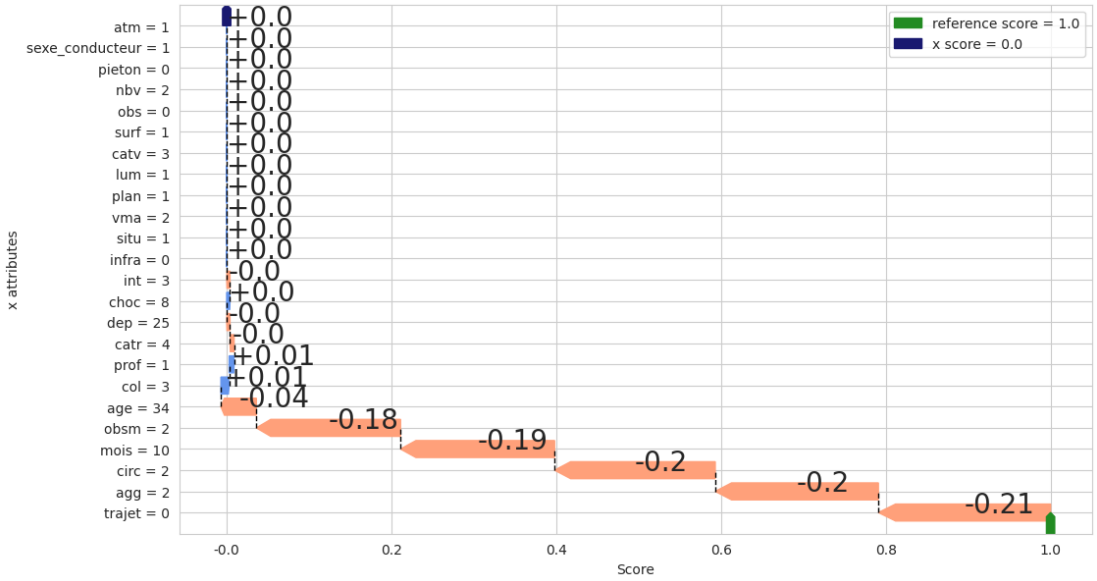


FIGURE 14 – Une approximation des valeurs de Shapley

6.4 Expliquer le modèle avec Lime

Nous avons également utilisé *Lime* afin d'avoir une idée des attributs qui contribuent le plus aux prédictions de notre modèle. On peut retrouver le code correspondant dans le fichier `lime.ipynb`.

Nous avons d'abord récupéré les indices des prédictions positives de notre modèle (avec lesquelles les entrées notre modèle prédit 1). Ainsi, on a pu regarder dans les cas où les prédictions sont 0 et les cas où les prédictions sont 1. On peut remarquer qu'on trouve des variables similaires à celles trouvées avec les valeurs de Shapley. On retrouve notamment assez fréquemment l'attribut *agg*. Cependant, contrairement au résultat que pouvait nous donner *ShapKit*, l'attribut *mois* est moins représenté. Les contributions sont également plus réparties entre les attributs. Enfin, il semble que, comme nous avions pu l'observer avec les valeurs de Shapley, l'attribut *sexe_ conducteur* ait un faible impact sur la prédiction.

7 Premières conclusions

Pour ce projet nous avons cherché un dataset qui nous intéressait, et nous avons choisi celui rapportant les accidents de la route. Pourvu de nombreux attributs, nous avons dû adapter leur format avant de pouvoir les analyser de manière univariée puis bivariée. Lors du premier essai, nous avons fait le choix d'utiliser des arbres de décision. Ce qui nous a conduit à transformer beaucoup de nos variables en OneHot. L'accuracy de ce modèle était élevée, mais il a du mal à trouver les accidents réellement mortels. Pour ce modèle, nous avons par la suite calculé le base rate puis effectué quelques audites tels que dice, BlackBoxAuditing et shapkit.

Par la suite nous avons essayé de trouver un classifieur qui avait de meilleures performances, c'est

ainsi que nous avons trouvé GaussianNB, qui est meilleur pour détecter les véhicules impliqués dans des accidents vraiment mortels. Pour ce modèle nous avons aussi effectué les audits évoqués précédemment, mais nous ne les avons pas analysés dans ce rapport. Ce qui conclut notre premier rendu.

L'analyse des données puis les audits que nous avons effextués après entrainement du modèle nous ont permis d'identifier plusieurs attributs sensibles. D'abord nous pensions que le genre du conducteur pourrait avoir un impact sur la prédiction. Cela aurait été un biais de notre modèle. Or nous avons pu observer via les différents outils utilisés que le genre du conducteur a un impact très faible sur la prédiction du modèle. En revanche, le travail effectué nous a permis d'identifier d'autres attributs sensibles qui eux ont réel impact sur les sorties du modèle. Les attributs *âge*, *dep* et *mois* sont des attributs que nous avons identifié comme sensibles pour notre modèle. Ils ont un impact assez important sur les prédictions et pourtant ne devraient pas être décisifs dans la classification d'un accident mortel ou non mortel.

8 Une réflexion pour la suite

Cette première partie nous a permis de nous familiariser avec notre jeu de données. L'objectif de classification a été une question qui nous a suivi du début de ce projet jusqu'à la rédaction de ce rapport. Les premières conclusions nous ont mené encore une fois à la réflexion sur une nouvelle direction à suivre pour la classification. Ne faudrait-il pas classifier l'ensemble des gravité ? Nous nous basons dans cette première partie sur des données dans lesquelles nous avons seulement 5% d'accidents mortels. Ne serait-il pas judicieux de prendre en compte l'ensemble de gravité et donc de travailler sur des proportions plus conséquentes ?

Ces questions seront déterminantes pour la suite de ce projet.

Deuxième partie

Correction des biais

9 Améliorations du modèle

Cette deuxième partie du projet a commencé par l'amélioration des premiers resultats. Nous allons parcourir dans cette section les différentes amélioration effectuées avant d'arriver à la correction des biais.

9.1 Nouvelle métrique de base : taux d'erreur et de justesse

Afin de trouver l'attribut à protèger et pour mieux diagnostiquer les effets des diférentes méthodes de réductions de biais, nous avons mis en place une nouvelle métrique de base. En tout il y a 4 caluls :

— Taux d'**erreur** pour l'attribut **sensible** :

$$P(\overline{Y} = 1|(Y = 0, Z = 1))$$

— Taux d'**erreur** pour l'attribut **privilégié** :

$$P(\overline{Y} = 1|(Y = 0, Z = 0))$$

— Taux de **justesse** pour l'attribut **sensible** :

$$P(\overline{Y} = 1|(Y = 1, Z = 1))$$

— Taux de **justesse** pour l'attribut **privilégié** :

$$P(\overline{Y} = 1|(Y = 1, Z = 0))$$

Seules, ces métriques n'ont que peu d'intérêt, c'est lorsqu'on les regarde ensemble qu'elles prennent tout leur intérêt pour la Fairness. En effet, l'équité est garantie si la différence entre ces taux pour l'attribut senseible et l'attribut privilégié est nulle. En effet on serait en droit de s'attendre à ce que la justesse et l'erreur se produise dans les même proportions. Notons pour finir que la liste des taux de succès est incomplète, on aurait pu ajouter le cas $\overline{Y} = 0|Y = 0$, mais ce n'est pas ce qui nous intéresse le plus dans notre cas. On souhaite se concentrer sur les prédictions d'accidents mortels.

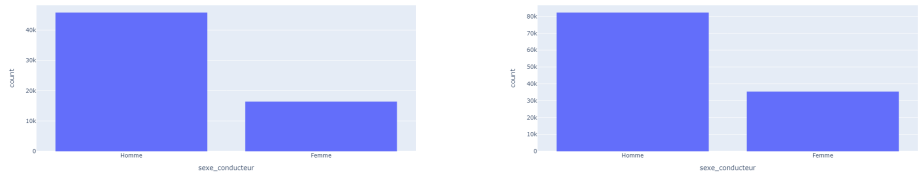
9.2 Sur-échantillonnage

Le pré-traitement des données a été amélioré avec l'utilisation de l'outil de sur-échantillonnage *SMOTE*. Nous avions pu remarquer lors de l'analyse de nos données que les accidents mortels sont minoritaires (environ 5%). Cet outil nous permet de rééquilibrer les classes afin d'avoir un meilleur apprentissage sur la classe des accidents mortels. Le *SMOTE* va générer de nouveaux individus minoritaires qui ressemblent à ceux déjà présents sans pour autant être identiques. Nous pouvons voir sur la figure 15 que le rééquilibrage nous donne 50% pour chaque classe tout en affectant peu les proportions sur les autres attributs comme nous le montre la figure 16 qui prend pour exemple l'attribut *sexe_conducteur*. Les résultats observés dans la suite du projet sont bien plus concluants en utilisant cette méthode. Tous les résultats obtenus dans la suite

utilisent donc cette méthode lors du prétraitement.

(a) Accidents mortels avant SMOTE (b) Accidents mortels après SMOTE

FIGURE 15 – Accidents mortels avant/après SMOTE



(a) Homme/Femme avant SMOTE (b) Homme/Femme après SMOTE

FIGURE 16 – Proportion Homme/Femme avant et après SMOTE

9.3 *Random Forest Classifier*

La première partie de ce projet utilisait un *Decision Tree Classifier*. Afin d'avoir un modèle plus robuste, nous avons décidé d'utiliser une forêt aléatoire. À première vue, les résultats étaient bien moins bons. En effet le nombre de vrais positifs est très faible. Cependant, en modifiant le seuil d'acceptation on peut faire évoluer la quantité de vrais positifs et obtenir des résultats bien plus intéressants. Une classe a été créée afin de surcharger la méthode `predict` de la classe *RandomForestClassifier*. Le seuil a été fixé dans un premier temps à 0.2. Nous verrons dans la suite comment régler ce seuil afin de réduire au maximum les biais.

9.4 Attributs sensibles

L'analyse du nouveau modèle nous a mené à de nouvelles observations. D'abord, nous avons remarqué une réduction du disparate impact pour le sexe féminin. De plus, les différents outils d'évaluation des biais nous montrent tous que le sexe féminin est plus disciminé sur ce modèle. L'utilisation de ce modèle a également eu pour effet de réduire l'influence de l'âge, le mois et le département, ce qui est un point positif. Nous pouvons notamment l'observer au travers des valeurs de Shapley obtenues avec ce nouveau modèle.

Dans la suite de ce projet nous nous penchons donc sur la correction des biais de l'attribut *sexe_conducteur*.

10 Pre-processing

[1]

10.1 Repondération

Dans le contexte de la Frainess, la repondération a pour objectif de rééquilibrer le set d'entrainement en attribuant des poids différents à chaque instance. Ce poid est calculé en fonction de l'impact que cette instance a sur l'attribut sensible. Après apprentissage et observation des résultats de notre modèle, nous pouvons faire deux observations. Premièrement les métriques de fairness sont très positives. Le disparate impact passe de 0.50 à 0.94 pour les femmes. La seconde observation est que l'accuracy de notre modèle a en revanche diminuée de 0.87 à 0.75. Cela peut s'expliquer par une augmentation du nombre de faux positifs. En revanche un point intéressant est que le nombre de vrais positifs est ici plus élevé que le nombre de positifs prédits faux, ce qui n'était pas le cas sur les modèles testés jusqu'ici. La matrice de confusion (figure 17) nous montre les résultats obtenus après repondération. Un effet intéressant de cette intervention sur le set d'entrainement est que le modèle parvient mieux à trouver les morts vraiment mort qu'auparavant. La baisse en accuracy s'explique par l'augmentation d'erreur de détection des accidents qui auraient du être labelisé accidents non-mortels. En résumé ce nouveau modèle catégorise trop d'accidents comme étant mortels. Mais parmi les accidents détectés comme étant mortels seul un sur huit l'est vraiment. Dans notre cas c'est intéressant car dans un objectif de prévention routière, on veut écarter les causes d'accidents non mortels des campagnes de sécurité pour augmenter leur efficacité.

10.2 Disparate Impact Remover

Les résultats obtenus avec le *Disparate impact remover* sont similaires à ceux obtenus avec la repondération. On obtient une accuracy de 0.75 et un disparate impact de 0.89. Le disparate impact est cependant un peu moins bon que celui obtenu avec la repondération pour une accuracy équivalente. On peut remarquer que tout comme avec la repondération, la matrice de confusion (figure 18a) nous montre une tendance allant vers plus de faux positifs. On remarque également encore une amélioration des prédictions de vrais positifs qui est très proche de celle obtenue par la méthode précédente. Le disparate impact remover peut être réglé avec un paramètre *level*. La figure 18b nous montre que le disparate impact reste plutôt stable sauf avec les paramètres 0.6 et 0.7. La valeur la plus élevée est de 0.89 à 0.9. Les résultats obtenus avec ce modèle étant très proches de ceux avec la repondération, on préférera utiliser la repondération dont l'entrainement est bien plus rapide.

1. Le pre-processing a été effectué sur un modèle avec seuil

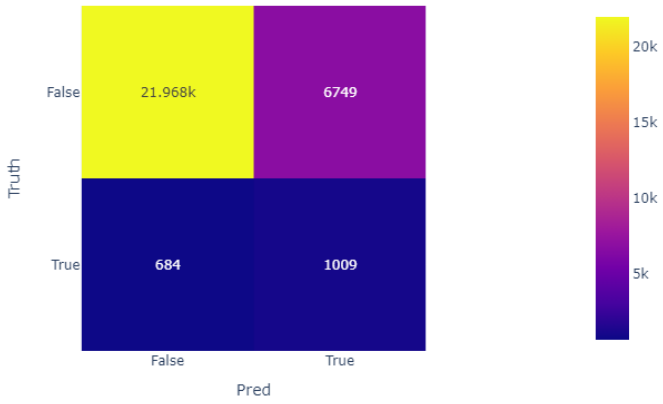
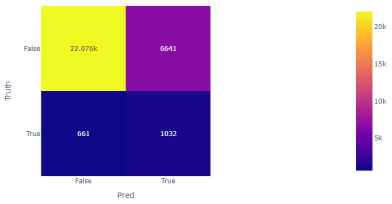
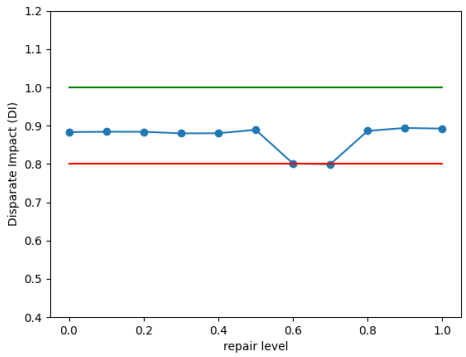


FIGURE 17 – Matrice de confusion avec repondération



(a) Matrice de confusion avec Disparate Impact Remover



(b) Disparate impact en fonction du paramètre *level*

FIGURE 18 – Résultats disparate impact remover

10.3 Représentation latente fair

11 In-processing

11.1 Adversarial debiasing

11.1.1 Explications

Cette méthode vise à entrainer un classifieur à maximiser l'accuracy tout en réduisant la possibilité pour un seconde modèle de prédire la sortie en fonction de l'attribut sensible. En réduisant la possibilité de prédire la sortie en fonction de l'attribut sensible, on réduit les biais sur cet attribut. La spécificité de cette méthode par rapport aux autres est donc qu'elle n'utilise plus le modèle que nous avons définit auparavant mais un autre modèle.

11.1.2 Résultats

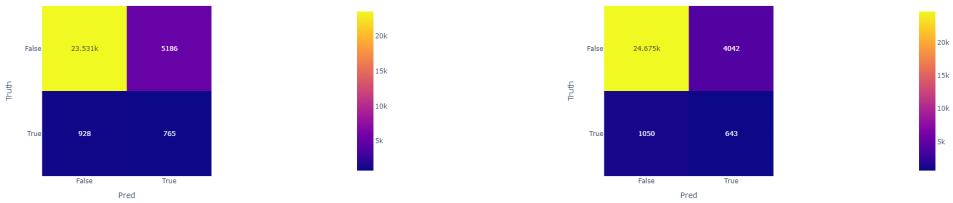
Les résultats obtenus sont plutôt intéressant que ce soit sans ou avec la réduction de biais. En effet, que ce soit avec ou sans la réduction des biais le disparate impact est supérieur à 1. Nous pouvons expliquer cela par le fait que cette méthode n'utilise pas de `RandomForestClassifier`, ce qui donner des résultats très différents. L'accuracy dimunue légèrement avec la réduction des biais mais reste aux alentours de 0.80. On note que l'accuracy sans réduction des biais est elle moins élevée qu'avec notre modèle. La figure 19a nous montre les résultats obtenus avec ce modèle. Nous pouvons remarquer que le nombre de prédictions vraies est plutôt intéressant mais plus faible que le nombre de prédictions faussements négatives.

Les résultats obtenus avec cette méthode sont tout de même très intéressants car nous conservons une accuracy acceptable tout en ayant un très bon disarate impact. En revanche, l'objectif étant d'améliorer notre modèle (une forêt aléatoire) en réduisant ses biais, l'utilisation d'un nouveau modèle que nous connaissons peu est discutable.

11.1.3 Un peu de pre-processing

Les résultats obtenus avec ce modèle sont intéressants c'est pourquoi nous avons voulu ajouter du pre-processing afin d'essayer de l'améliorer d'avantage et notamment au niveau de l'accuracy. Nous avons choisi d'utiliser la repondération qui a donné d'excellents résultats sur notre modèle utilisant une forêt aléatoire. Les résultats obtenus sont encore meilleur que sans repondération. Sur la matrice de confusion figure 19b on peut voir que les bonnes prédictions vrai un légèrement diminuées. Cependant on remarque que le nombre de prédictions vraies qui sont fausses a fortement diminué au profit des prédictions fausses prédites fausses.

On obtient avec ce modèle des métriques de fairness aussi bonne qu'avant avec un disparate impact légèrement supérieur à 1, une equal opportunity difference proche de 0 tout en ne diminuant pas l'accuracy qui est maintenant de 0.83.



(a) Sans pre-processing

(b) Avec repondération des données

FIGURE 19 – Matrices de confusions de modèles Adversarial debiasing avec réduction de biais

12 Post-processing

12.1 Reject option classification

12.1.1 Un nouveau seuil

Le but de cette méthode de correction de biais post-preprocessing, est de faire varier le seuil de décision de manière à trouver celui qui a la meilleure fairness, pour le comparer au seuil ayant la meilleure accuracy.

Toujours à l'aide d'AIF 360, nous avons utiliser ROC en paramétrant le groupe privilégié à 1 (Hommes) et les lésés à 0 (Femmes). Les résultat sont les suivant :

— Sans fairness : *0.1189*
— Avec fairness : *0.1289*

De ces résultats on peut sortir deux observations. Tout d'abord, ROC à resenti le besoin de déplacer le seuil de sorte à corriger le biais. Ensuite, la correction effectuée est minime dans son ampleur.

12.1.2 Les nouvelles performances du modèle

Les répercussions sur l'accuracy quand à elles sont notables mais acceptables. On passe en effet de *0,876* à *0,790*. Cependant, si on regarde un peu plus en détail la matrice de confusion, il parrait intéressant de noter que ce modèle détecte mieux les accidents mortels véritablement mortels. Pour être précis, ce nombre passe de *660* à *1044* ce qui constitue *62%* des accidents mortels répertoriés dans l'ensemble. Ceci se fait au prix d'un augmentation des accidents non motrtels faussement détectés. Malgré tout, le taux de détection de ce type d'accident reste élevé à *80%*. Voilà pourquoi la qualification d'acceptable pour la baisse en accuracy est parfaitement adaptée.

12.1.3 Les performances pour la Fairness

La première observation que l'on peut faire, c'est que le disparate impact se voit corrigé pour les femmes. Pour être précis, avec le nouveau seuil, le disparate impact augmente à *0.640* là ou celui des hommes est à *0.636*. Il s'est donc à la fois presque aligné à celui des hommes tout en se rapprochant de 1.

On remarque aussi que les taux d'erreur

$$P(\overline{Y} = 1 | (Y = 1, Z = 0))$$

$$P(\overline{Y} = 1|(Y = 0, Z = 0))$$
$$P(\overline{Y} = 1|(Y = 1, Z = 1))$$
$$P(\overline{Y} = 1|(Y = 0, Z = 1))$$

chez les femmes se sont rapprochés de ceux des hommes. Cela qui atteste d'une correction de biais substentielle.

En revanche, il faut sousligner que la correction de biais n'est pas complète. On peut même dire qu'elle est inférieure à d'autres méthodes telles que reweighing.

12.2 Calibrated equalized odds

12.2.1 Explications

Cette méthode va modifier les sorties du modèle afin d'augmenter l'égalité entre les valeurs de l'attribut sensible. Cette méthode est testée sur différents seuils de prédiction afin de trouver le modèle le plus efficace. On note cependant que les résultats son assez dépendants de l'aléatoire. Ils peuvent être variables d'une exécution à l'autre et peuvent même donner de moins bons résultats que sans le post-processing, ce nous allons voir par la suite.

12.2.2 Les résultats obtenus

La figure 20 nous montre l'évolution de l'accuracy et de l'égalité de l'attribut sensible (ici homme/femme) dans les prédictions en fonction du seuil. L'objectif est d'avoir l'accuracy la plus élevée tout en ayant le plus d'égalité sur l'attribut sensible (donc *equal opportunity difference* le plus proche de 0). Le résultat obtenu nous montre que l'application de cette méthode a ici eu un effet négatif. En effet nous pouvons remarquer que le meilleur résultat obtenu est avant post-processing avec un seuil légèrement en dessous de 0.15. Comme dit précédemment, les résultats varient avec l'aléatoire et on arrive à avoir des cas où les résultats sont meilleurs avec le post-processing.

12.2.3 Mise en perspective

Comme nous avons pu le voir ce résultat n'est pas vraiment satisfaisant. Cependant cette méthode reste très intéressante pour avoir une vu globale du modèle et des résultats obtenus. Le `RejectOptionClassifier` vu précédemment nous indique directement un seuil optimal sans qu'on ait une idée de comment évoluent les biais en fonction du seuil. Ici nous avons des résultats selon les différents seuils. De plus une donnée intéressante ici est la visulisation de l'accuracy. Cela nous permet de réduire les biais tout en ayant le contrôle de l'accuracy (que ce soit avec ou sans post-processing), ce qui n'est pas forcément évident avec les autres méthodes.

13 Résultats

14 Conclusion

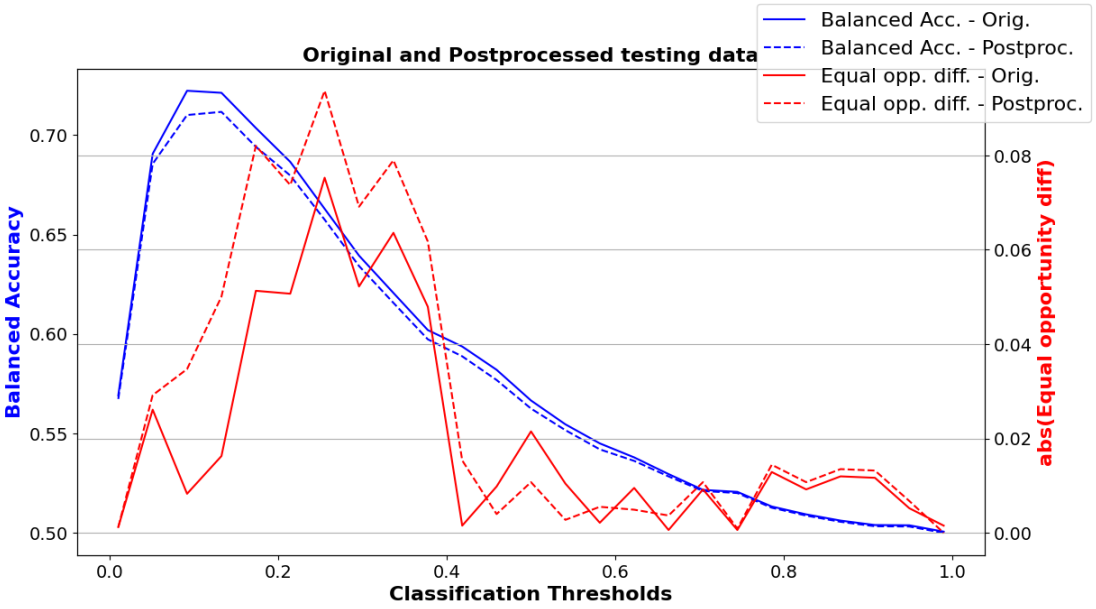


FIGURE 20 – Résultats avec et sans post-processing en fonction du seuil

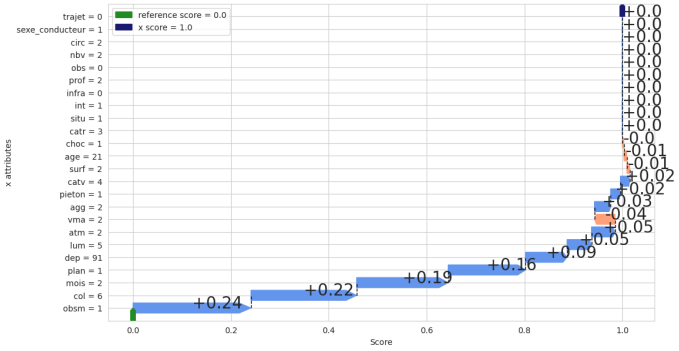
A Les données conservées pour le modèle.

Attribut	Description
Num_Acc	Numéro d'identifiant de l'accident
jour mois	Jour de l'accident, mois de l'accident
an	Année de l'accident
hrmn	Heure et minutes de l'accident
lum	Conditions d'éclairage dans lesquelles l'accident s'est produit
dep	Code INSEE du département
agg	Localisation en agglomération
int	Type d'intersection
atm	Conditions atmosphériques
col	Type de collision
adr	Adresse postale (pour les accidents en agglomération)
lat	Latitude
long	Longitude
catr	Catégorie de route
voie	Numéro de la route
circ	Régime de circulation
nbv	Nombre total de voies de circulation
vosp	Présence d'une voie réservée
prof	Profil en long de la route
plan	Tracé en plan de la route
surf	État de la surface de la route
infra	Présence d'aménagements ou d'infrastructures
situ	Situation géographique de l'accident
vma	Vitesses maximale autorisées
id_vehicule	Identifiant du véhicule (clé étrangère)
catv	Catégorie du véhicule impliqué dans l'accident
obs	Type d'obstacle heurté
obsm	Type d'obstacle mobile heurté
choc	Point de choc initial
manv	Manœuvre principale avant l'accident
catu	Catégorie d'usager (conducteur, passager, piéton)
grav	Gravité de l'accident pour l'usager
sexe	Sexe du conducteur
trajet	Motif du déplacement au moment de l'accident
mortal	Indique si le véhicule est impliqué dans un accident mortel (calculé dans le notebook)

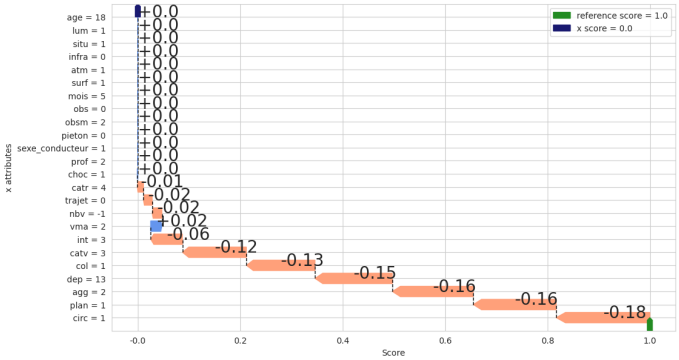
B Les données abandonées et la raison de leur abandon

Attribut	Description	Raison de l'abandon
id_vehicule	Identifiant numérique unique du véhicule	information administrative, une fois les tables jointes, cette information sans intéret, n'est plus utile
num_veh	Identifiant du véhicule pour associer les passagers du même vehicule	Information administrative, une fois les tables jointes, cette inforamtion sans intêret, n'est plus utile
id_usager	Identifiant des usagers dans la base	Utilisé lors des jointures, inutile par la suite
com	Numéro de commune (code INSEE)	Commune trop spécifique, ça donne trop de catégories
adr	Adresse de l'accident	Trop spécifique, ne donne rien de pertinent
lat	Latitude de l'accident	Ces informations positionelles ne reflètent pas le type d'endroit où l'accident a lieu
Long	Longitude de l'accident	Ces informations positionelles ne reflètent pas le type d'endroit où l'accident a lieu
Num_Acc	Numéro d'identifiant de l'accident	Information administrative, une fois les tables jointes, cette inforamtion sans intêret, n'est plus utile
voie	Numéro de la route	Pas pertinent, plusieurs routes ont le même numéro, certaines routes ont des sections plus dangereuses que d'autres
V1	Numéro de la route	Pas pertinent, plusieurs routes ont le même numéro, certaines routes ont des sections plus dangereuses que d'autres
V2	Numéro de la route	Pas pertinent, plusieurs routes ont le même numéro, certaines routes ont des sections plus dangereuses que d'autres
vosp	Signale l'existence d'une voie réservée, indépendamment du fait que l'accident ait eu lieu ou non sur cette voie.	Attribut trop peu renseigné, peu d'accidents concernés
pr	Numéro de la borne (routière) en amont	Pas pertinent, beaucoup de routes sont non bornées, n'indique rien d'intéressant sur l'accident
pr1	Distance en mètres à la borne en amont	Pas pertinent, beaucoup de routes sont non bornées, n'indique rien d'intéressant sur l'accident
lartpc	Largeur terplein central en mètres	Trop peu renseigné
larrout	Largeur de la chaussée en mètres	Trop peu renseigné
senc	Indique si le véhicule allait vers une borne supérieure ou inférieure à la précédente.	Information administrative, pas utile
manv	Indique la manoeuvre en cours au moment de l'accident	Trop souvent inconnue ou non renseignée
motor	Motorisation du véhicule	Trop peu renseignée
place	Place du passager dans l'habitacle, 10 indique un piéton	Utilisée pour trouver les conducteurs et les piétons, devient inutile à l'échelle de l'accident ensuite
grav	Donne l'état de gravité de l'usager accidenté	Utilisée pour trouver les accidents mortels, devient inutile à l'échelle de l'accident ensuite
sexe	Sexe de l'usager (binaire)	Utilisé pour déterminer le sexe du conducteur, devient inutile à l'échelle de l'accident ensuite
secu1, secu2, secu3	Trois attributs renseignant tous les 3 sur l'utilisation d'équipements de sécurité	Attribut difficilement implémentable et peu fiable car non renseigné apparait beaucoup
locp	Localisation du piéton par rapport au véhicule	Le point de vue de l'accident rend difficile l'implémentation de cet attribut peu renseigné
actp	Action piéton lors de l'accident	Le point de vue de l'accident rend difficile l'implémentation de cet attribut peu renseigné
etatp	Précise si le pieton était accompagné	Le point de vue de l'accident rend difficile l'implémentation de cet attribut peu renseigné
occutc	Nombre d'occupants du transport en commun	Cet attribut est trop peu renseigné
jour	Numéro du jour de l'accident	Peu utile, il aurait été plus utile de connaitre le jour de la semaine
hrmn	Heures : minutes	Format non adapté, redondant de luminosité (lum) qui est plus précis
an	Année de l'accident	Utilisée pour déterminer l'âge du conducteur, n'est plus utile par la suite
an_nais	Année de naissance de l'usager	Utilisée pour déterminer l'âge du conducteur, n'est plus utile par la suite, pas adapté au point de vue de l'accident

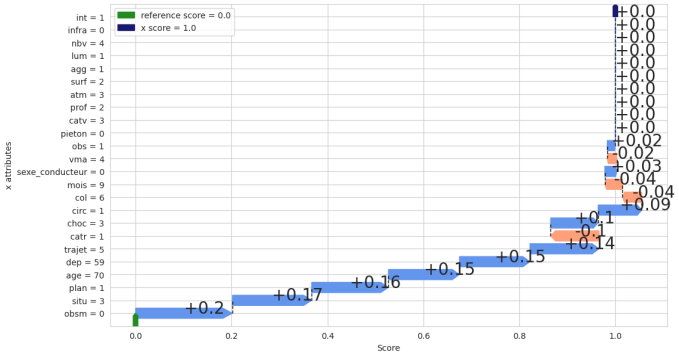
C Diagrammes en cascade des valeurs de Shapley



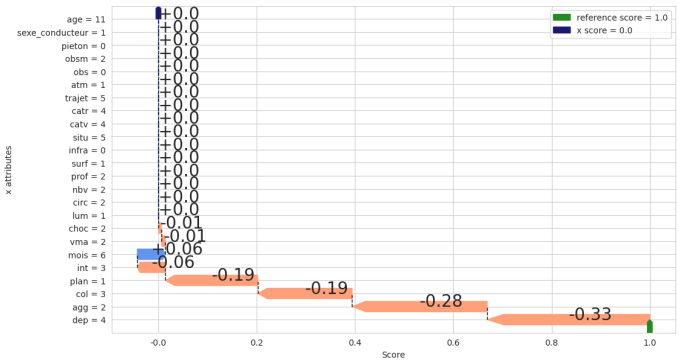
(a)



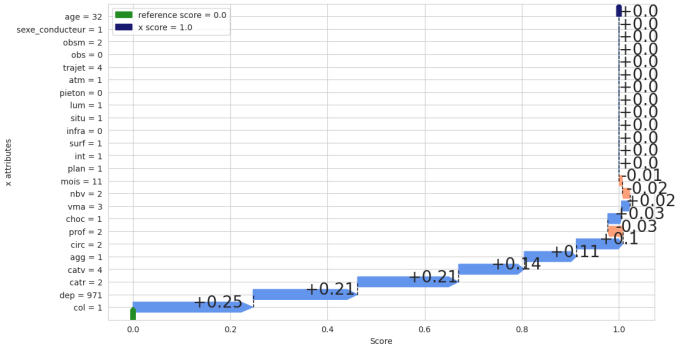
(b)



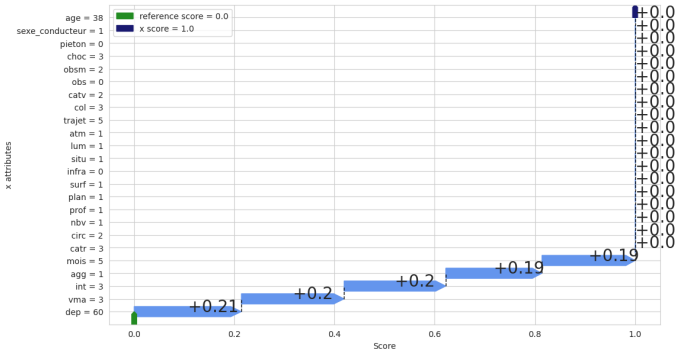
(c)


reference score = 1.0
x score = 0.0
x attributes
Score
-0.19
-0.19
-0.28
-0.33

(a)


reference score = 0.0
x score = 1.0
x attributes
Score
+0.25
+0.21
+0.21
+0.14
+0.11
+0.1

(b)


reference score = 0.0
x score = 1.0
x attributes
Score
+0.21
+0.2
+0.2
+0.19
+0.19

(c)