Projet Émission de CO2 par les véhicules

Analyse de la pollution en CO2 des véhicules commercialisés en France en 2023

Membres de l'équipe : Ndèye Oumy DIAGNE , Justine GAUTERO, Lucie MONTAGNE

Sommaire

Introduction

Ce projet représente une opportunité de contribuer à un enjeu environnemental majeur tout en mettant en pratique les connaissances acquises lors de notre formation.

Dans le cadre de cette étude, nous avons pour objectif principal l'analyse de l'émission de CO2 au niveau national d'un échantillon de véhicules variés.

D'un point de vue **technique**, le projet implique :

- **la compréhension des normes environnementales** pour sélectionner les variables pertinentes et identifier les caractéristiques techniques liées aux émissions du CO2. Elle a un rôle important dans la construction du dataset ;
- l'utilisation des **outils d'analyse de données et de prédiction** pour résoudre la problématique.

D'un point de vue **économique et écologique**, le projet a pour but :

- de **réduire l'empreinte carbone** en réduisant les émissions de CO2 des véhicules;

- de proposer aux constructeurs des solutions permettant de **répondre à des besoins clients** soucieux de l'environnement mais aussi de **réduire les taxes** en optimisant le Malus écologique des véhicules concernés.

Pour atteindre ces objectifs nous allons essayer de répondre à la problématique en suivant les étapes suivantes :

- ☑ Identifier les modèles de véhicules qui émettent le plus de CO2 (on se focalise sur ce type de pollution, le dioxyde de carbone, sachant qu'il existe aussi la pollution aux particules fines, dioxyde d'azote (NO) et les hydrocarbures non brûlés. Le CO2 > joue sur le changement climatique)
- ☑ Identifier les caractéristiques techniques qui influencent le plus les émissions de CO2
- ☑ Estimer et anticiper la pollution future des nouveaux véhicules
- ☑ Déterminer comment réduire l'impact carbone des voitures

- ☑ Faire des recommandations aux constructeurs pour réduire les émissions de CO2 des véhicules

Ce domaine d'activité nous est totalement inconnu, nous avons effectué des recherches et consulté les documentations existantes pour nous imprégner du sujet.

Compréhension et manipulation des données

Choix des sources de données

Nous avions à disposition deux sources à partir desquelles trouver les données concernant l'émission de CO2 des véhicules. **Ces données sont toutes open sources et commentées** :

- Dataset de l'Etat français (données issues de l'ADEME) datant de 2014 : **"Émissions de CO2 et de polluants des véhicules commercialisés en France"**
- Dataset de l'Union Européenne à travers l'agence Européenne de l'environnement (EEA) datant de 2023 (mais dont les données ne sont pas encore finales) : **"Monitoring of CO2 emissions from passenger cars, 2023 - Provisional data"**

En parallèle, nous avons trouvé un troisième dataset pour le territoire français (données également issues de l'ADEME) mais plus récent puisque mis à jour en septembre 2024 : **"ADEME-CarLabelling".** Il contient donc les derniers véhicules commercialisés dotés des dernières technologies .

Nous avons fait le choix de sélectionner ce dernier dataset, car au-delà du fait d'être le plus récent, il s'avère être beaucoup plus complet en termes de variables que le dataset français de 2014.

Par ailleurs, il est important de préciser qu'**une nouvelle norme d'essai d'homologation des véhicules légers est entrée en vigueur en 2017 au niveau européen** : la norme WLTP (WorldWide harmonized Light vehicle Test Procedures). Celle-ci remplace la norme européenne NEDC (Nouveau cycle Européen De Conduite) et intègre désormais la mesure de la consommation de carburant (à différentes vitesses), l'autonomie électrique et les rejets de CO2 (à différentes vitesses) et de polluants (tels que les particules fines, dioxyde d'azote (NO) et hydrocarbures non brûlés), **d'où la plus grande précision du dataset de 2024.**

A contrario, nous avons décidé de ne pas nous servir du dataset européen :

- Dans un premier temps, nous n'avons pas identifié de clé primaire pour réaliser une éventuelle jointure entre les deux dataset,
- Nous avons donc envisagé, dans un second temps, de réaliser une jointure à l'aide d'une clé créée à partir de variables communes que nous avions identifiées ('Marque', 'Modèle', 'Groupe'). Nous y avons renoncé pour les raisons suivantes :
    - Le nombre de modèles enregistrés entre les 2 dataset diffère énormément (par exemple : le modèle 'CADDY' de Wolkwagen concerne 376 enregistrements pour le dataset européen (filtré sur le territoire français) contre 52 enregistrements pour le dataset français).
    - Certains modèles ne sont pas présents pour le dataset français (par exemple : le modèle 'DUSTER' de Dacia).
    - **Une jointure aurait eu donc comme conséquence un nombre très important de valeurs manquantes que nous n'aurions pu combler.**

Voici la volumétrie du dataset sélectionné :

	ADEME-CarLabelling
Territoire	France
Taille totale du jeu de données	1,1 Mo
Nombre lignes	3604 enregistrements
Nombre colonnes	52 variables
Types de données	object, float, interger
Temporalité	Données de 2023 mises à jour en septembre 2024
Qualité des données Valeurs manquantes Doublons avec la méthode ProfileReport*	 19,39% NAN 165 lignes identiques au minimum

** Pour être considérée comme étant un doublon, une ligne de notre df doit au moins avoir les mêmes valeurs pour les variables allant de 'Marque' à 'Puissance fiscale'*

La pertinence

La variable cible

Nous avons identifié **la variable 'CO2 vitesse mixte Max'** pour deux raisons :

- Cette variable provient d'un test du **cycle WLTP** dont l'objectif est de mesurer les émissions polluantes en circulation réelle. La **'vitesse mixte'** permet donc de simuler l'émission de CO2 avec une combinaison de différents types de conduite pour mieux refléter les conditions réelles du véhicule (vitesse basse, moyenne, haute et très haute).

- Nous avons préféré sélectionner la variable **'CO2 vitesse mixte Max'** plutôt que la variable **'CO2 vitesse mixte Min'** car elle contient moins de valeurs manquantes. La plupart des marques semble considérer cette première mesure comme étant celle de référence. La différence est que la variable 'CO2 vitesse mixte Max' est testée dans les pires conditions possibles de consommation de carburant et d'émissions de CO2 et à l'inverse la variable 'CO2 vitesse mixte Min' est testée dans les conditions les plus favorables.

<u>Les particularités du jeu de données</u>

<table>
<tr><th colspan="2">Au niveau des variables</th></tr>
<tr>
<td>Nous avons décidé de conserver 6 variables sur les 52 variables totales.

Les variables conservées sont :

-Energie ;

-Carrosserie ;

-Puissance Fiscale ;

-Type de boîte ;

-Masse OM Max ;

-CO2 vitesse mixte max (variable cible).</td>
<td>Les variables supprimées sont :

- les variables mesurant les autres types de pollution qui ne concernent donc pas notre problématique ;
- les variables concernant les mesures propres aux véhicules électriques, non concernées par la problématique ;
- les variables de type ‘administratives’ : ‘Marques’, ‘Modèle’, ‘Libellé modèle’, ‘Description Commerciale’, ‘Bonus-Malus, ‘Barème Bonus-Malus’, ‘Prix’, ‘Nombre de rapport’, qui ne nous semblent pas pertinentes au regard de notre problématique ou inutiles à l'entraînement du modèle de machine learning ;
- des variables fortement corrélées entre elles que nous avons validées avec des tests statistiques (cf ci-dessous) : ‘Cylindrée’, ‘ Nombre de rapports’, ‘Gamme’, ‘Poids à vide’, ‘Rapport poids puissance’ et les variables concernant la consommation de carburant. Ces dernières présentent une forte corrélation avec d’autres variables similaires (cf Heatmap cf ci-dessous), ce qui pourrait introduire de la redondance dans le modèle et ainsi nuire à sa performance. Nous avons opté pour leur exclusion afin d’éviter un problème de multicolinéarité.</td>
</tr>
</table>

Au niveau des lignes	
Nous devrons également **supprimer du jeu de données, tous les enregistrements qui concernent les voitures électriques** (véhicules non émetteurs de CO2), car elles ne sont pas concernées par notre problématique. Cela représente 134 enregistrements.	Nous avons remarqué qu'une grande partie des valeurs manquantes concernaient les véhicules électriques (puisqu'ils ne sont pas émetteurs de CO2, nous n'avions pas de données pour les variables dédiées aux mesures de CO2). Les supprimer implique une baisse du nombre de valeurs manquantes. Il sera d'ailleurs intéressant de comparer le pourcentage de valeurs manquantes avant et après le pré-processing. Les garder pourrait fausser nos analyses futures.
Une attention doit être portée aux **véhicules hybrides**.	Les tests sont fait uniquement en conso vitesse Min / Max (pas de données sur les autres variables conso car données similaires). Choix variables conso vitesse mixte Max et mixte Min // Vitesse mixte Max et mixte Min.
Nous constatons que **beaucoup de valeurs sont présentes sous le type "object".**	Le type a été modifié pour pouvoir réaliser les manipulations nécessaires.

Description des variables conservées

Nom de la variable	Description
Energie	Type d'énergie de la voiture (Essence, gazole...)
Carrosserie	Type de carrosserie (Berline, Suv...)
Puissance Fiscale	Puissance en CV fiscaux
Type de boîte	Boite de vitesse (Auto, mécanique...)
Masse OM Max	Masse en ordre de marche maxi (en Kg), ou masse maximale d'un véhicule prêt à circuler (poids du véhicule, conducteur, passagers, cargaison, fluides)
CO2 Vitesse mixte Max	CO2 mixte combiné - maximum (en g/Km)

> Voir en détail l'analyse du dataset <

Visualisation du jeu de données

Nous utilisons tout d'abord la data visualisation dans le but d'obtenir une description complète du jeu de données. Elle nous permettra d'avoir connaissance des valeurs nulles, manquantes et éventuellement aberrantes ou extrêmes du dataset, des marques de voitures les plus distribuées, des énergies et des types de boîtes les plus représentés ainsi que des modèles et gammes les plus vendues par exemple.
Enfin, elle nous permettra de voir les relations entre les variables et de déterminer des corrélations. Cette analyse va nous aider dans le choix des caractéristiques de notre jeu de données final.

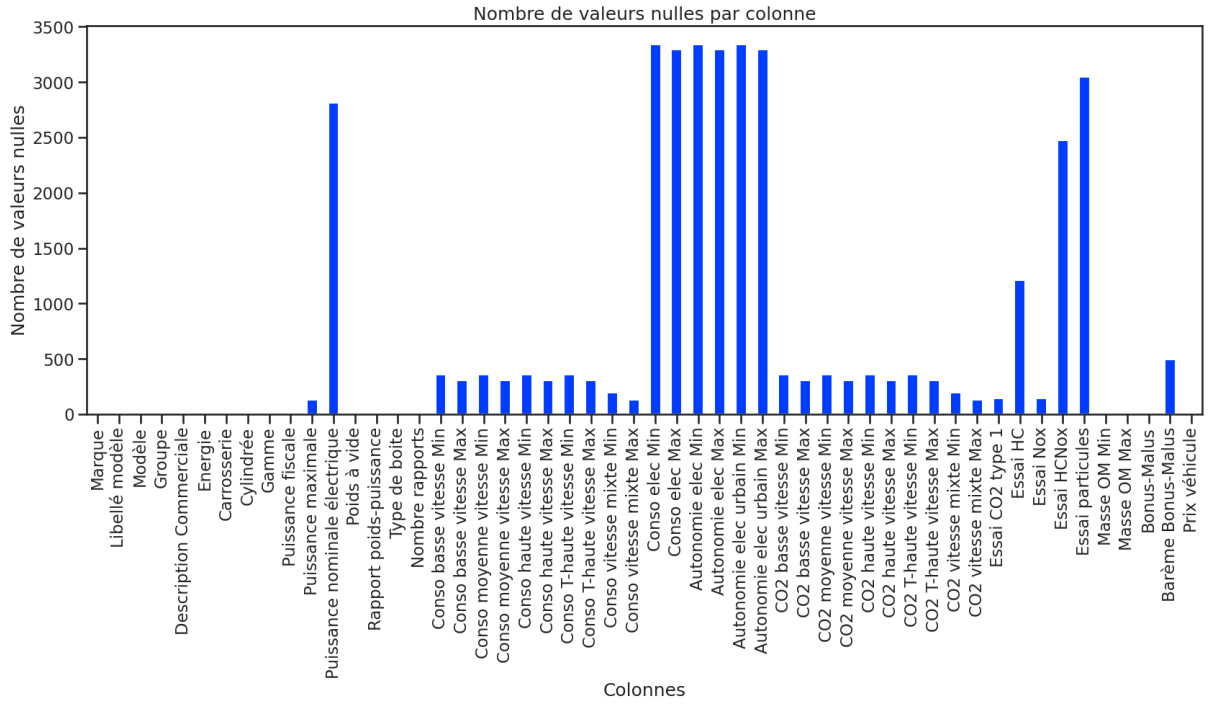
Distribution des valeurs nulles dans le jeu de données

Mise en évidence des valeurs nulles par colonne

Pour évaluer la qualité de notre base de données, il faudrait se retrouver avec le moins de valeurs nulles possible. Nous constatons qu'il y a beaucoup de valeurs nulles sur les conso / Autonomie / Puissance électriques, ce qui est normal par rapport à la représentativité des voitures électriques dans le jeu de données.

Nous remarquons également beaucoup de données manquantes pour les variables relatives à la pollution (Nox, HC, Particules) , Nous n'allons pas utiliser ces variables dans notre étude. Nous nous intéressons qu'à l'émission de CO2 pour ce cas pratique.

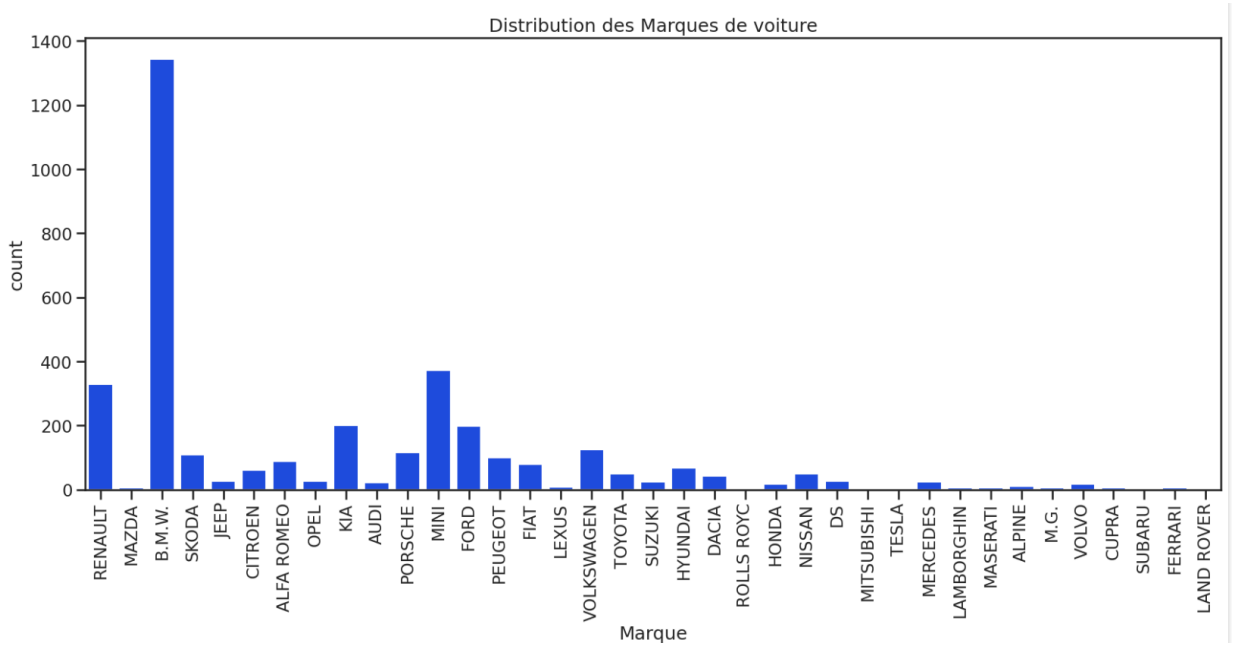
Ce graphique nous a également aidé à choisir la variable “CO2 Vitesse Mixte Max” et “ Conso vitesse mixte max” comme variables d’étude car elles contiennent le moins de valeurs manquantes que nous remplaceront par la médiane pour les besoins de l’étude.



Distribution des caractéristiques des véhicules commercialisées en France

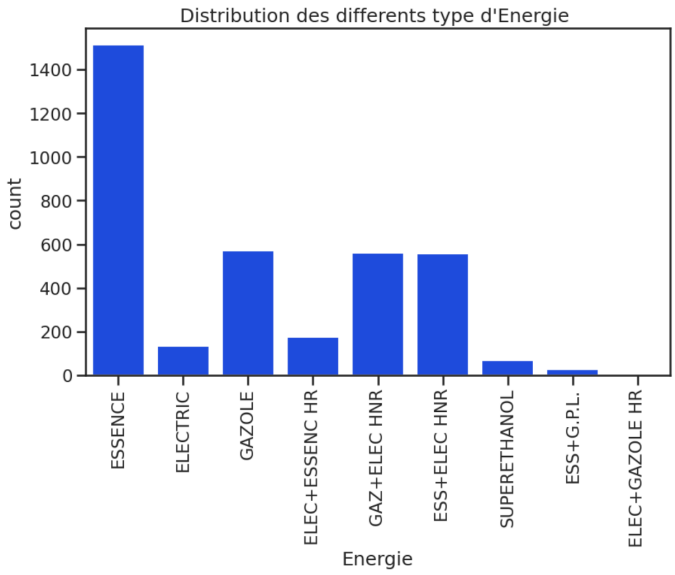
Distribution des Marques de voiture

Nous constatons que dans notre base de données les véhicules de la marque BMW sont les plus représentés suivis des véhicules Renault et Mini. La distribution des marques est disparate, nous risquons d’avoir des valeurs extrêmes avec des voitures Ferrari ou LandRover par exemple.



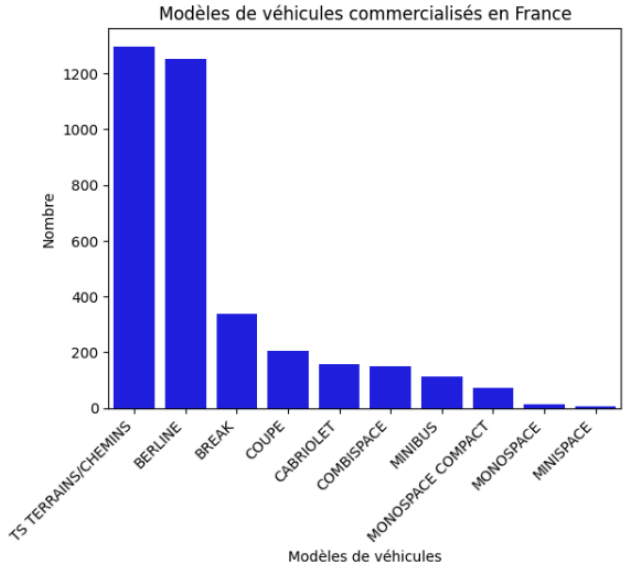
Distribution des énergies

Les véhicules de type Essence sont majoritairement représentés dans le jeu de données avec plus de 1400 véhicules. S'en suivent les voitures Gazole et les hybrides non rechargeables qui sont chacune autour de 600 voitures. Nous remarquons la présence des voitures de dernières générations : les voitures hybrides rechargeables et les électriques.



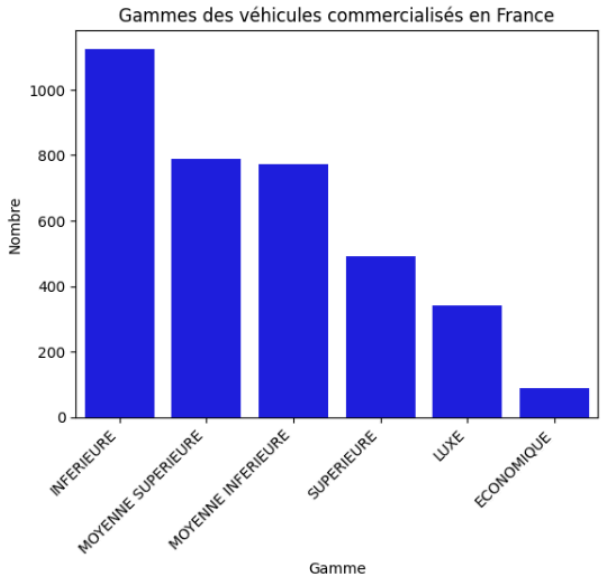
Distribution des modèles

Les modèles de véhicules les plus commercialisés aujourd'hui en France sont les tous terrains/chemins et les berlines.



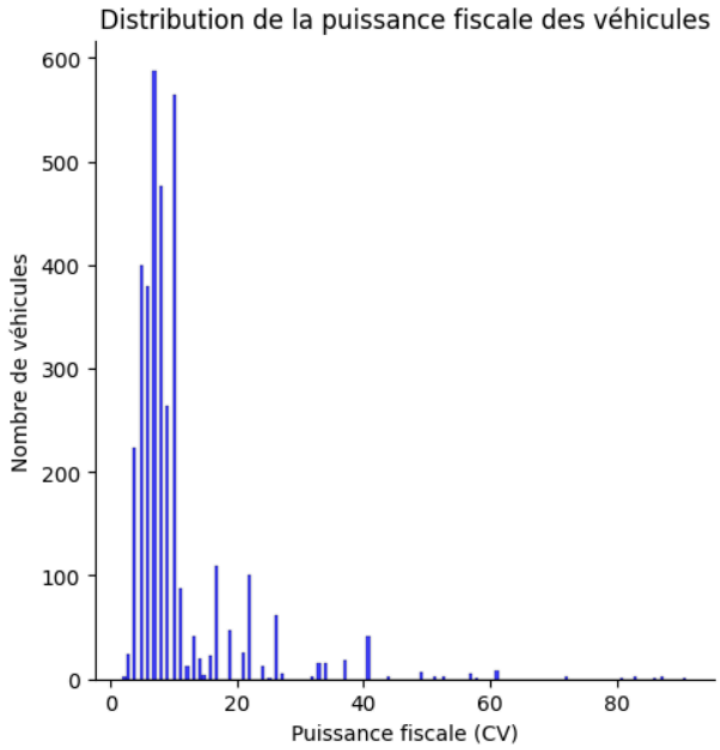
Distribution des gammes

La gamme la plus commercialisée aujourd'hui en France est la catégorie inférieure.



Distribution de la puissance fiscale des véhicules

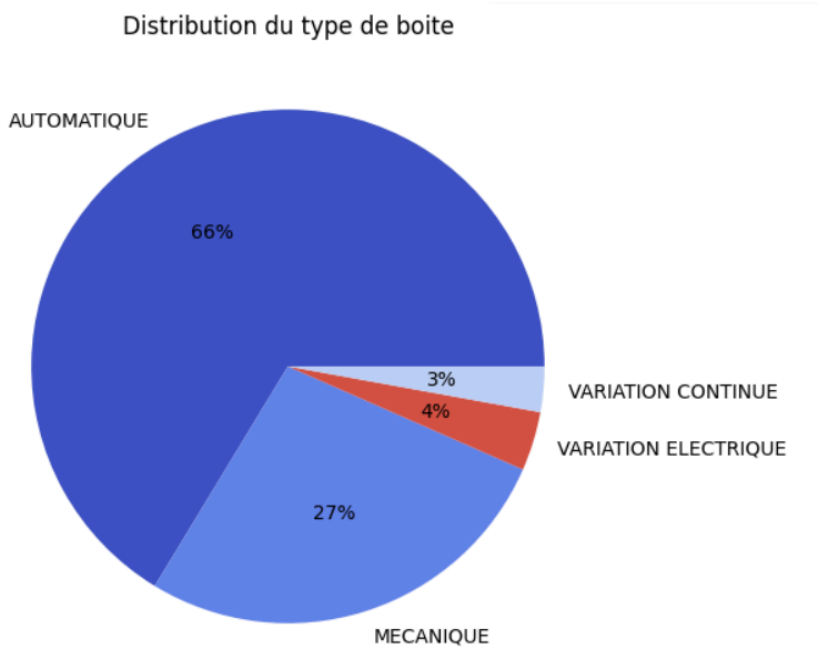
La très grande majorité des véhicules ont une **puissance fiscale comprise entre 2 et 11 CV.**



Distribution du type de boîte

La grande majorité des véhicules disposent d'une boîte automatique (66%), puis mécanique (27%), et pour finir à variation électrique et continue (respectivement 4 et 3%).

Spécifions que les boîtes à variation électrique ne concernent que les véhicules électriques (véhicules que nous suggérons de retirer du jeu de données car non émetteurs de CO2) et les boîtes à variation continue concernent les véhicules à essence et hybrides.

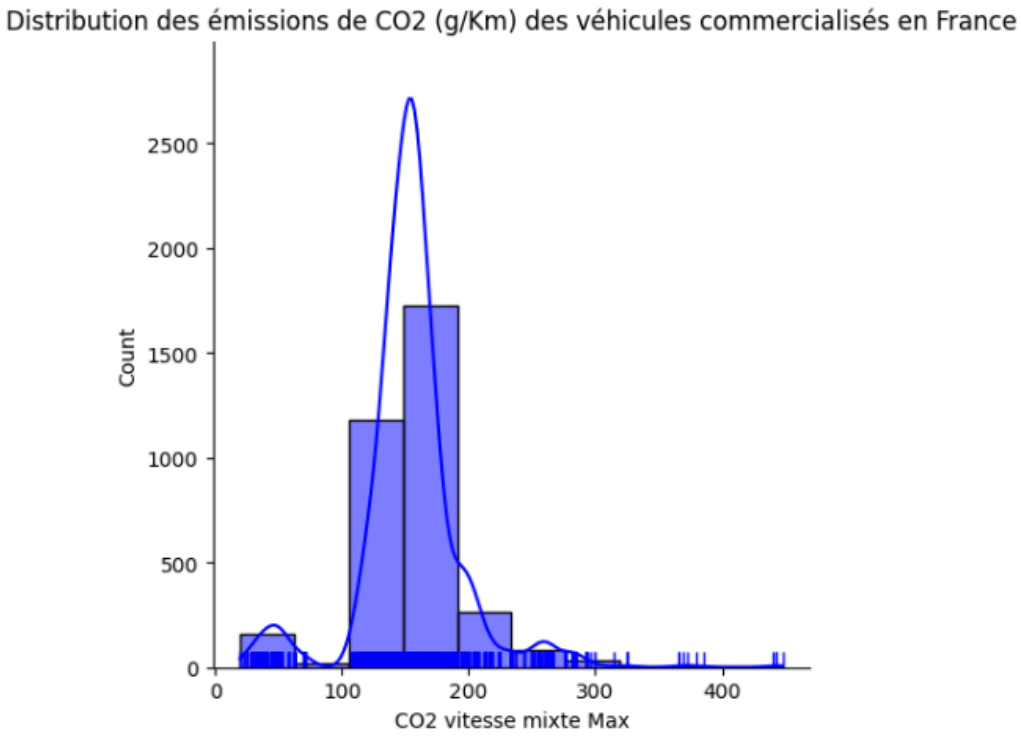


Distribution des caractéristiques des véhicules versus leurs émissions de C02

Distribution des émissions de CO2 (g/km)

La majorité des véhicules ont des émissions de CO2 situées entre environ 100 et 200 g/km.

Grâce à la courbe d'estimation de densité, on peut observer un pic aux alentours de 150 g/km, ce qui suggère que c'est la valeur la plus fréquente pour ces véhicules.

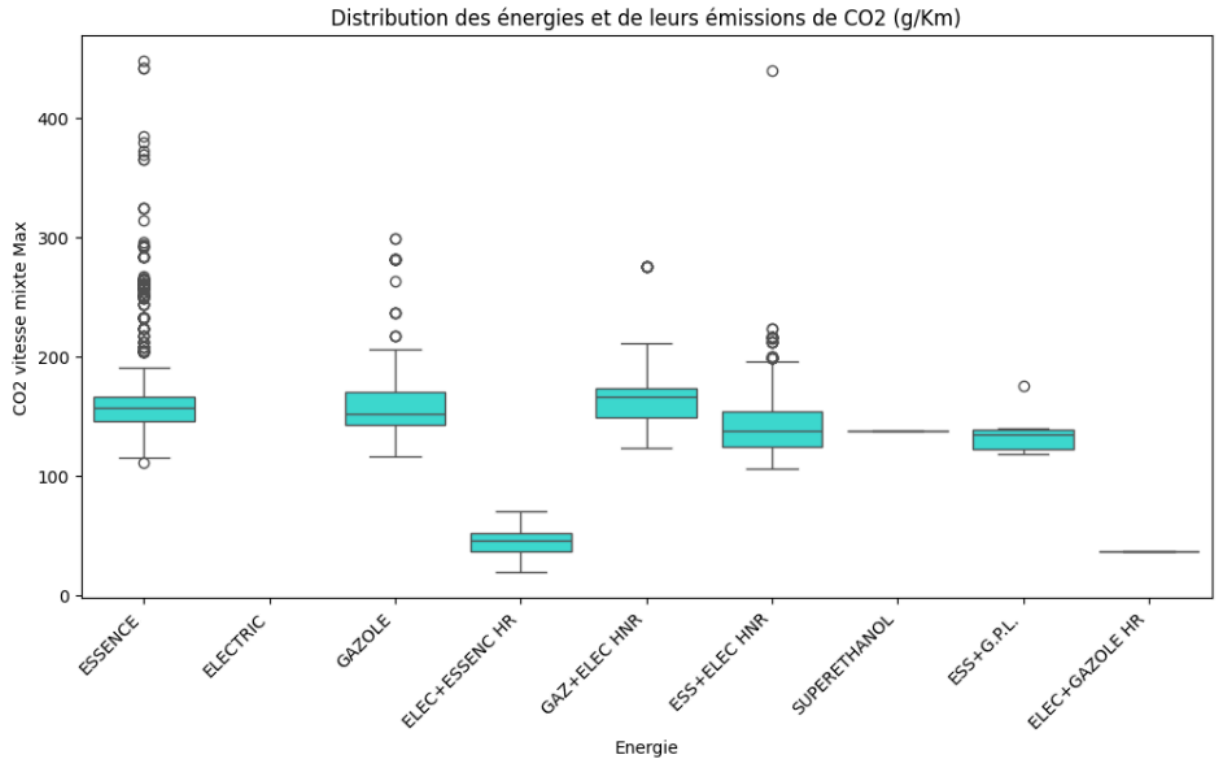


Distribution des énergies et de leurs émissions de CO2 (g/km)

Les véhicules qui émettent le plus de CO2 sont les véhicules utilisant l'essence comme type d'énergie, Ils sont également les plus représentés dans le jeu de données. Mais si on compare l'émission de CO2 des véhicules hybrides non rechargeables Essence et Gazole, on remarque que ceux à essence sont légèrement plus polluants alors que leur distribution est équivalente. La corrélation entre l'émission de CO2 et l'énergie sera approfondie par une étude statistique ci-dessous.

- Les véhicules **électriques et hybrides rechargeables** ont les émissions de C02 les plus faibles (<100).
- Les véhicules **essence et diesel** émettent le plus de CO2. Ils montrent des émissions similaires avec une médiane de 150 g/Km. Cependant, l'essence présente beaucoup de valeurs extrêmes.

- Les véhicules **hybrides non rechargeables**, le GPL et le superéthanol ont une médiane qui se rapproche de celles des véhicules essence et diesel. Ils ont cependant moins de valeurs extrêmes.

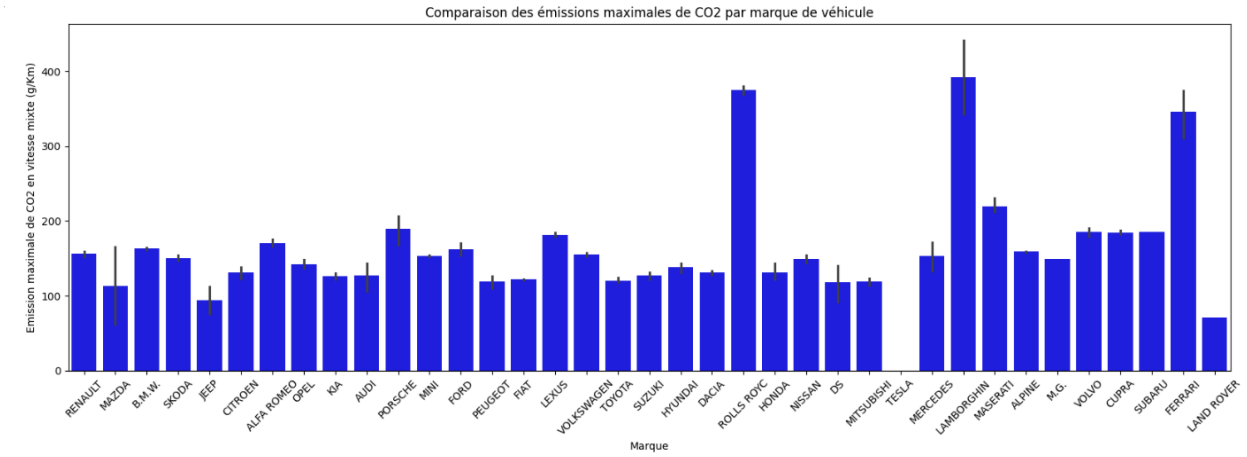


Comparaison des émissions maximales de CO2 par marque de véhicule

Les marques les plus émettrices de CO2 sont LAMBORGHINI, ROLLS ROYCE, FERRARI, MASERATI et PORSCHE (gamme LUXE).

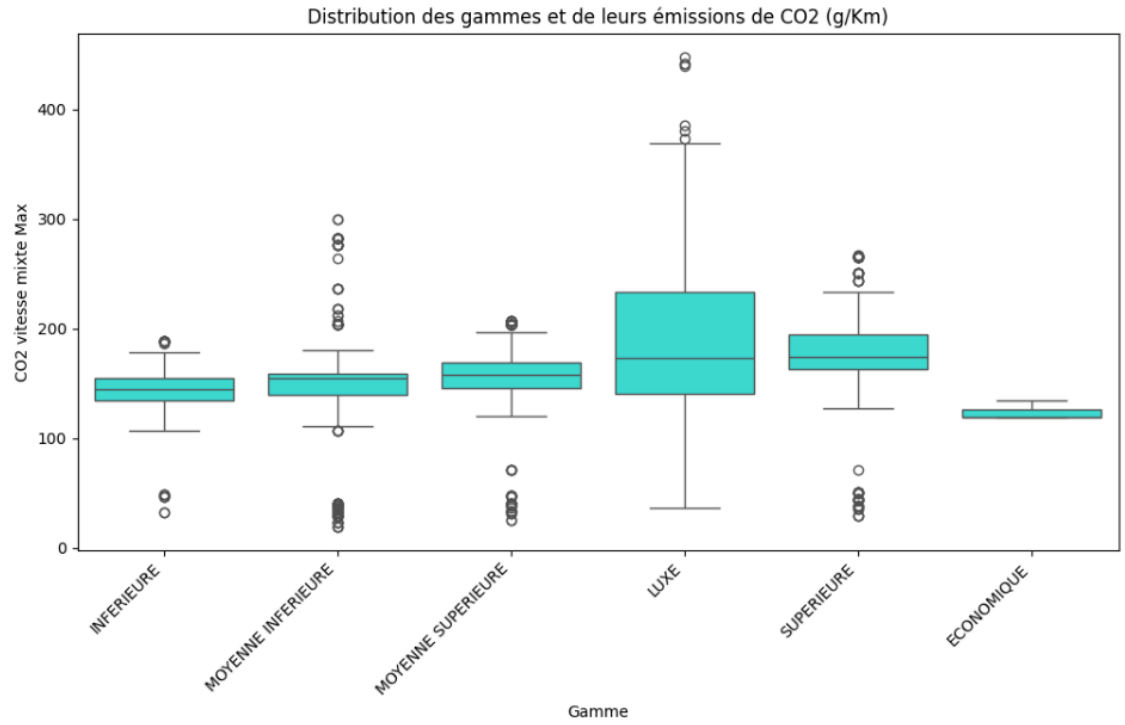
A noter que ce sont des véhicules avec une barre d'erreur plus grande que le reste des marques (ceci est aussi le cas pour les marques MAZDA, JEEP, AUDI et DS). Les barres sont un indicateur d'incertitude associée à la moyenne, qui peut être influencée par la taille de l'échantillon de ces marques ou par la grande variabilité de leurs données.

Nous ne disposons pas de données pour la marque TESLA, dont les voitures sont uniquement électriques.



Distribution des gammes de véhicules et de leurs émissions de CO2 (g/km)

On observe que les véhicules des gammes Luxe et Supérieure émettent le plus de CO2.

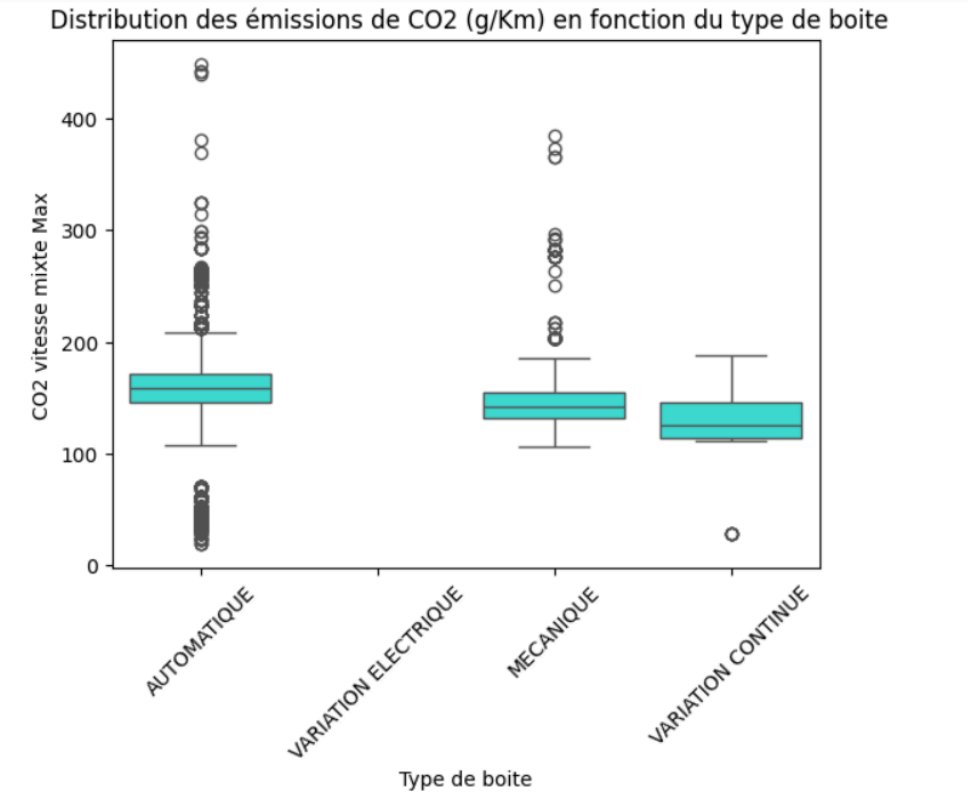


Distribution des émissions de CO2 en fonction du type de boîte

Les boîtes automatiques émettent plus de CO2 que les boîtes mécaniques et à variation continue **avec une médiane située autour des 150g de CO2/Km** (à une vitesse mixte maximum), contre 130g de CO2/Km pour les boîtes mécaniques et 120g de CO2/Km pour les boîtes à variation continue.

Nous remarquons aussi la forte présence de valeurs extrêmes, notamment pour la catégorie boite automatique et en moindre mesure pour la catégorie boîte mécanique. Cela pourrait expliquer la diversité des types de véhicules équipés de ces boîtes.

A contrario, nous pouvons remarquer la dispersion relativement faible de la catégorie variation continue, indiquant que les émissions de CO2 sont plus homogènes. L'outlier au dessous de 100g/Km pourrait indiquer la performance exceptionnelle d'un véhicule en termes d'émission de CO2.

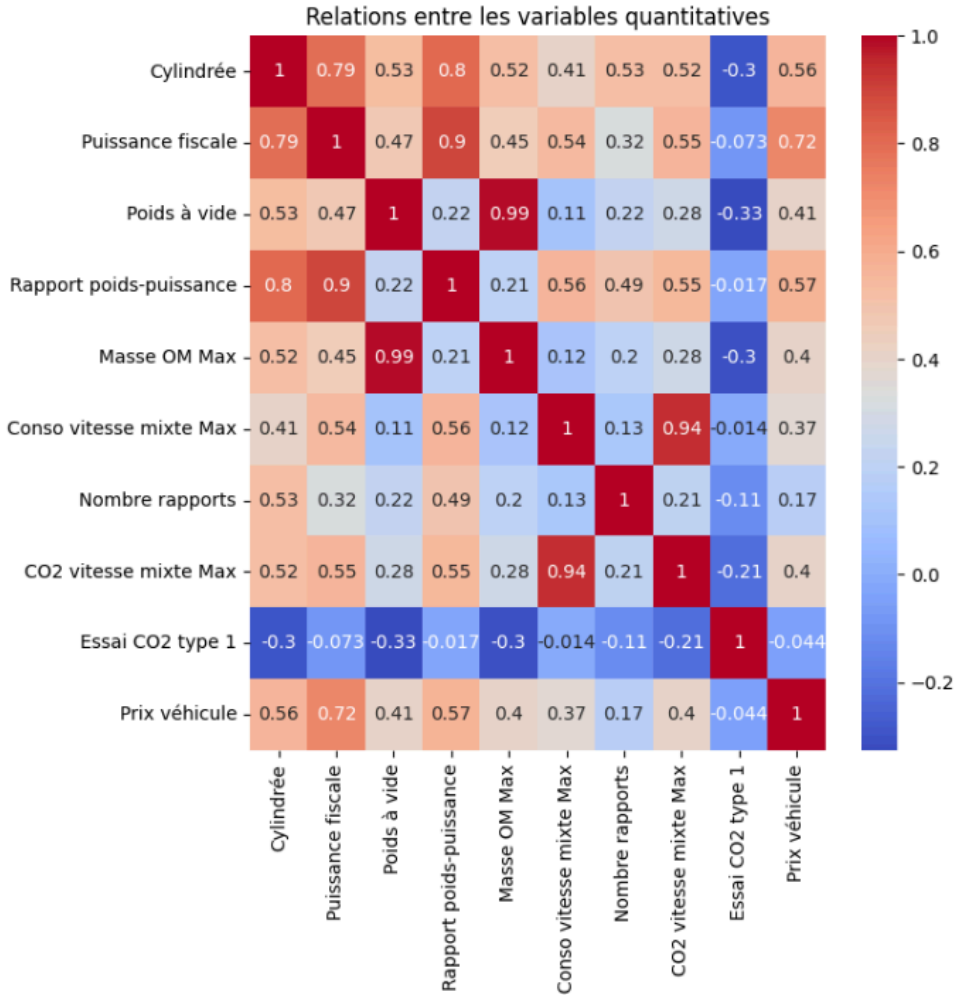


Relations entre les variables quantitatives

Heatmap des variables quantitatives

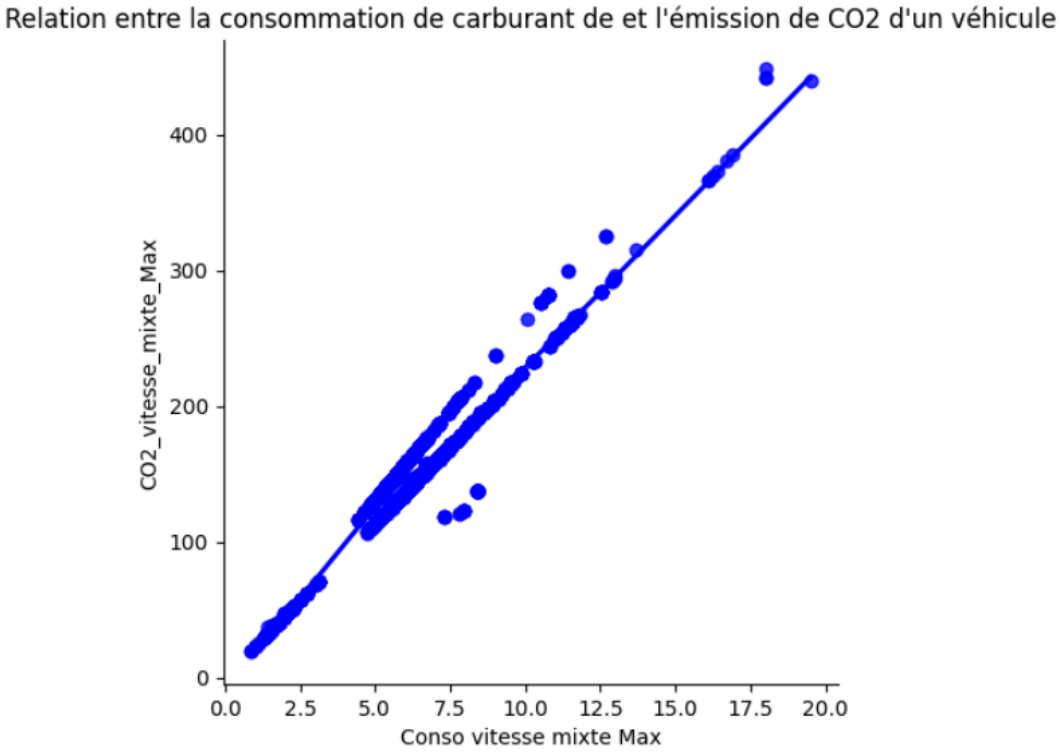
Nous avons sélectionné les variables quantitatives les plus pertinentes pour créer une Heatmap permettant d'identifier les corrélations. Ceci nous permettra de savoir quelle variable nous allons écarter du jeu de données. **On observe une forte corrélation positive entre** :

- La Consommation d'un véhicule et son Émission de CO2 (0,94) ;
- La Puissance fiscale et le rapport Poids-Puissance (0,9) ;
- Le rapport Poids-Puissance et la Cylindrée (0,8) ;
- Le Poids à vide et Masse OM Max (0,99) ;
- La Puissance fiscale et Cylindrée (0,79).



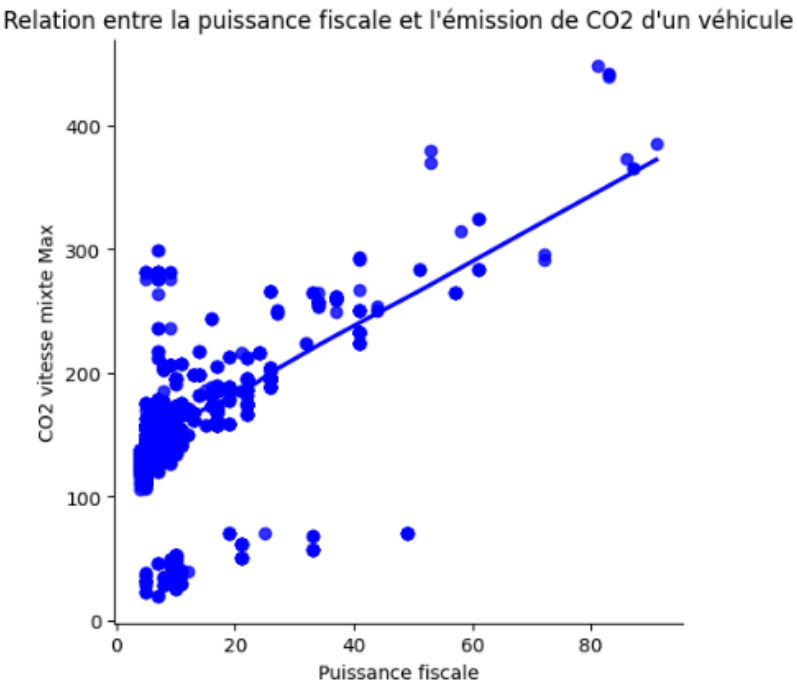
Relation entre la consommation de carburant et l'émission de CO2

Nous pouvons observer une corrélation positive entre ces 2 variables.



Relation entre la puissance fiscale et l'émission de CO2

Nous pouvons observer une corrélation positive entre ces 2 variables. La plupart des véhicules ayant une puissance fiscale inférieure à 15 ont une émission de CO2 comprise entre 100 et 200 g/km. On observe tout de même des valeurs extrêmes.



Relations entre les variables catégorielles et quantitatives

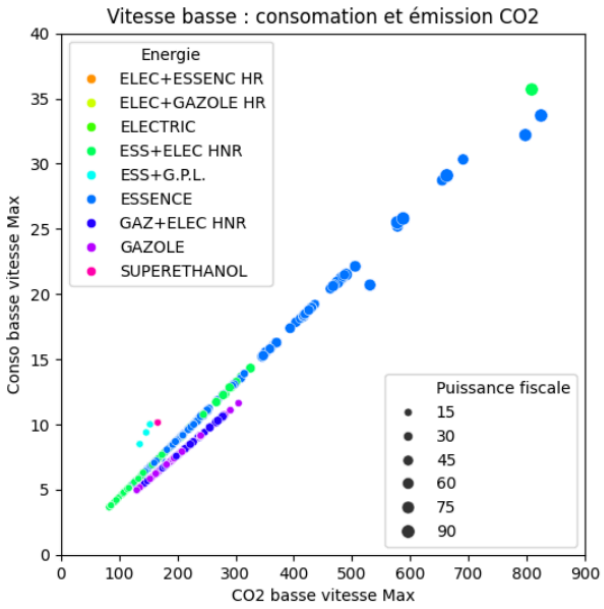
Relation entre la puissance fiscale, la consommation, l'énergie et l'émission de CO2

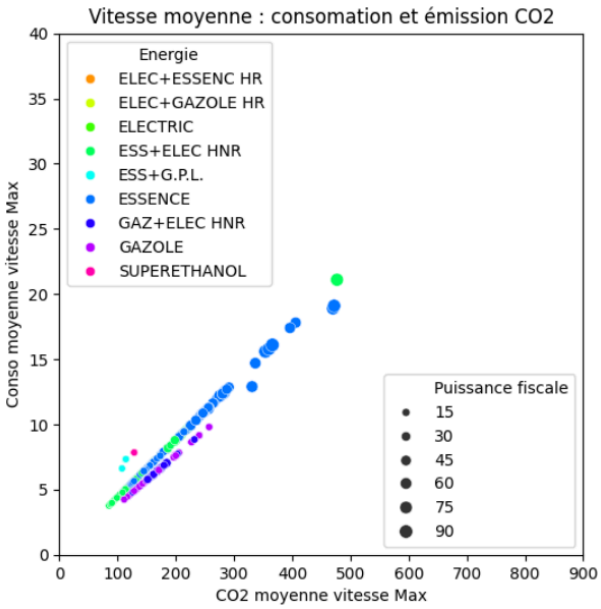
Ces graphiques décrivent la relation entre ces variables dans quatres contextes différents. Nous avons la consommation et les émissions de CO2 d'un véhicule en fonction de son énergie et de sa puissance :

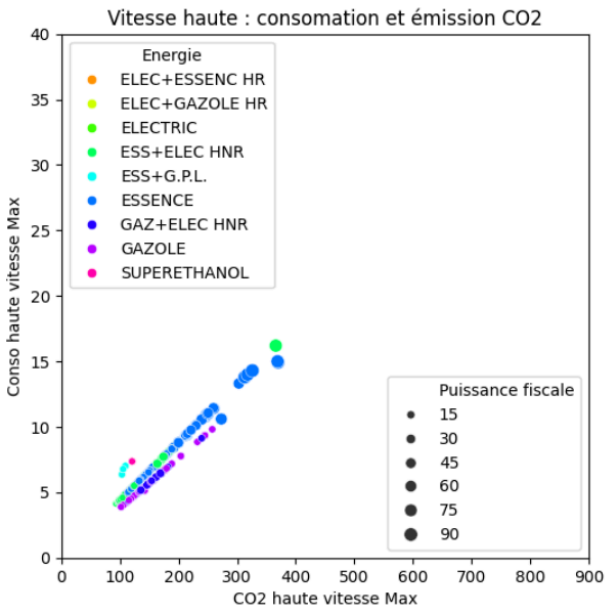
- en vitesse basse ;
- en vitesse moyenne ;
- en vitesse haute ;
- en vitesse très haute.

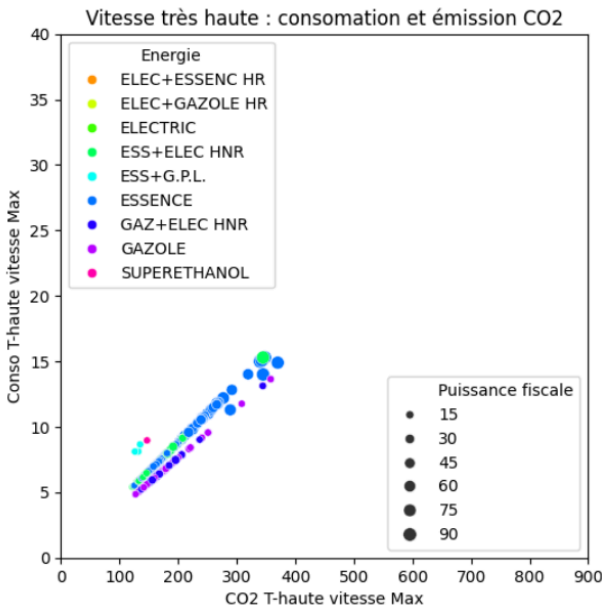
Nous pouvons observer :

- une **corrélation positive** entre ces variables ;
- Plus un véhicule est puissant et plus il émet de CO2 ;
- Les véhicules les plus puissants ont un moteur essence ;
- En vitesse basse, les véhicules à moteur essence et puissants consomment plus d'énergie et émettent beaucoup plus de CO2 (entre 400 et 800 g/km). Plus la vitesse augmente et moins les véhicules à essence émettent de CO2 ;
- A l'inverse, plus la vitesse augmente et plus certains véhicules diesel consomment du carburant et émettent du CO2.









Tests statistiques des valeurs quantitatives

<u>Corrélation entre l'émission de CO2 et la consommation de carburant</u>

Nous utiliserons le test de Pearson car il permet d'étudier le lien entre deux variables quantitatives.

Hypothèses :

- H0 : Les variables émission de CO2 et consommation de carburant ne sont pas corrélées ;
- H1 : Les deux variables sont corrélées.

Problématique rencontrée : nous avons remplacé les valeurs manquantes pour pouvoir réaliser ce test.

Résultats :

```
p-value:  0.0
coefficient:  0.941953037831601
```

Conclusion : la p-value est inférieure à 5% donc on rejette H0 et on conclut H1. On observe une forte corrélation entre les deux variables avec le coefficient (0,94).

Ces valeurs étant fortement corrélées, nous avons décidé de n'en conserver qu'une seule des deux dans le jeu de données. Nous avons sélectionné la variable 'émission de CO2' car il s'agit de notre variable cible.

Corrélation entre la puissance fiscale et la cylindrée

Nous utiliserons le test de Pearson car il permet d'étudier le lien entre deux variables quantitatives.

Hypothèses :

- H0 : Les variables sur la puissance fiscale et la cylindrée ne sont pas corrélées ;
- H1 : Les deux variables sont corrélées.

Problématique rencontrée : nous avons remplacé les valeurs manquantes pour pouvoir réaliser ce test.

Résultats :

```
p-value:  0.0
coefficient:  0.7909719947282163
```

Conclusion : la p-value est inférieure à 5% donc on rejette H0 et on conclut H1. On observe une forte corrélation entre les deux variables avec le coefficient (0,79).

Ces valeurs étant fortement corrélées, nous avons décidé de n'en conserver qu'une seule des deux dans le jeu de données. Nous avons sélectionné la variable 'puissance fiscale' car elle est plus facilement interprétable.

Tests statistiques une valeur catégorielle et une valeur quantitative

Corrélation entre l'émission de CO2 et l'énergie consommée

Nous utiliserons le test ANOVA car il permet d'étudier le lien entre une variable catégorielle et quantitative.

Hypothèses :

- H0 : Il n'y a pas d'influence significative du type d'énergie utilisé sur l'émission du CO2 ;
- H1 : Il y a une influence significative du type d'énergie utilisé sur l'émission du CO2.

Problématique rencontrée : pour pouvoir mener à bien ce test, nous avons dû renommer l'intitulé des colonnes en remplaçant les espaces par des '_'.

Résultats :

```
              df        sum_sq        mean_sq           F  PR(>F)
Energie      8.0  2.380666e+06  297583.264655  299.868681     0.0
Residual  3462.0  3.435615e+06     992.378608         NaN     NaN
Conclusion : La p-value (PR(>F)) est inférieure à 5% donc on rejette H0 et on conclut H1
```

Conclusion :

Nous pouvons donc conclure une forte influence significative du type d'énergie sur les émissions de CO2.

Corrélation entre l'émission de CO2 et du type de boîte (automatique, mécanique...)

Nous utiliserons le test ANOVA car il permet d'étudier le lien entre une variable catégorielle et quantitative.

Hypothèses :

- H0 : Il n'y a pas d'influence significative du type de boîte du véhicule sur l'émission du CO2 ;

- H1 : Il y a une influence significative du type de boîte du véhicule sur l'émission du CO2.

Problématique rencontrée : pour pouvoir mener à bien ce test, nous avons dû renommer l'intitulé des colonnes en remplaçant les espaces par des '_'.

Résultats :

```
                    df        sum_sq       mean_sq         F        PR(>F)
Type_de_boite      3.0  1.287460e+05  42915.336514  26.10714  1.075324e-16
Residual        3467.0  5.699110e+06   1643.816050       NaN           NaN
Conclusion : La p-value (PR(>F)) est inférieure à 5% donc on rejette H0 et on conclut H1
```

Conclusion :

Nous pouvons donc conclure une influence significative du type de boîte sur les émissions de CO2.

Corrélation entre l'émission de CO2 et le type de carrosserie

Nous utiliserons le test ANOVA car il permet d'étudier le lien entre une variable catégorielle et quantitative.

Hypothèses :

- H0 : Il n'y a pas d'influence significative du type de carrosserie sur l'émission du CO2 ;
- H1 : Il y a une influence significative du type de carrosserie sur l'émission du CO2.

Problématique rencontrée : pour pouvoir mener à bien ce test, nous avons dû renommer l'intitulé des colonnes en remplaçant les espaces par des '_'.

Résultats :

```
                  df        sum_sq        mean_sq           F         PR(>F)
Carrosserie      9.0  1.271970e+06  141330.008916  107.622126  3.334533e-178
Residual      3460.0  4.543692e+06    1313.205878         NaN            NaN
Conclusion : La p-value (PR(>F)) est inférieure à 5% donc on rejette H0 et on conclut H1
```

Conclusion :

Nous pouvons donc conclure une forte influence significative du type de carrosserie sur les émissions de CO2.

Corrélation entre le nombre de rapport et le type de boîte (automatique, mécanique…)

Nous utiliserons le test ANOVA car il permet d'étudier le lien entre une variable catégorielle et quantitative.

Hypothèses :

- H0 : Il n'y a pas d'influence significative du type de boîte sur le nombre de rapport ;
- H1 : Il y a une influence significative du type de boîte sur le nombre de rapport.

Problématique rencontrée : pour pouvoir mener à bien ce test, nous avons dû renommer l'intitulé des colonnes en remplaçant les espaces par des '_'.

Résultats :

```
                    df        sum_sq      mean_sq            F  PR(>F)
Type_de_boite      3.0  11614.307002  3871.435667  8388.764897     0.0
Residual        3600.0   1661.408869     0.461502          NaN     NaN
Conclusion : La p-value (PR(>F)) est inférieure à 5% donc on rejette H0 et on conclut H1
```

Conclusion :

Nous pouvons donc conclure une forte influence significative du type de boîte sur le nombre de rapport.

Ces valeurs étant fortement corrélées, nous avons décidé de n'en conserver qu'une seule des deux dans le jeu de données. Nous avons sélectionné la variable 'type de boîte' car elle est plus facilement interprétable.

Corrélation entre la gamme et la puissance fiscale

Nous utiliserons le test ANOVA car il permet d'étudier le lien entre une variable catégorielle et quantitative.

Hypothèses :

- H0 : Il n'y a pas d'influence significative de la gamme sur la puissance fiscale ;
- H1 : Il y a une influence significative de la gamme sur la puissance fiscale.

Problématique rencontrée : pour pouvoir mener à bien ce test, nous avons dû renommer l'intitulé des colonnes en remplaçant les espaces par des '_'.

Résultats :

```
               df         sum_sq       mean_sq           F  PR(>F)
Gamme         5.0  101545.432805  20309.086561  447.351606     0.0
Residual   3598.0  163343.760036     45.398488         NaN     NaN
Conclusion : La p-value (PR(>F)) est inférieure à 5% donc on rejette H0 et on conclut H1
```

Conclusion :

Nous pouvons donc conclure une forte influence significative de la gamme sur la puissance fiscale. Ces valeurs étant fortement corrélées, nous avons décidé de n'en conserver qu'une seule des deux dans le jeu de données. Nous avons sélectionné la variable 'puissance fiscale' car elle est plus facilement interprétable.

Features Engineering

Voici les étapes que nous avons réalisées dans le cadre du Features Engineering :

Choix des variables utiles

(cf Tableau "Les particularités du jeu de données")

Suppression des lignes qui concernent les véhicules électriques

(cf Tableau "Les particularités du jeu de données")

Pré-processing

Changement du type

La majorité des variables étaient en "object" alors que le dataframe contient beaucoup de variables numériques. Nous avons donc changé le type en "float" afin d'avoir des données numériques pour les statistiques et le machine learning.

Séparation du jeu de données en 1 jeu d'entraînement et de test

Nous allons séparer le jeu de données en jeu de données d'entraînement et jeu de données de test . Nous allons utiliser le ratio : 90% et 10% car nous avons un petit jeu de données.

```
Dimension du jeu d'entrainement : (594, 19)
Dimension du jeu de test : (66, 19)
```

Traitement des valeurs manquantes

Parmi les variables explicatives sélectionnées, nous n'avions aucunes valeurs manquantes :

Energie	Type d'énergie	object	0	not applicable	ELEC+ESSENC HR - ELEC+GAZOLE HR - ELECTRIC - ESS+ELEC HNR - ESS+G.P.L. - ESSENCE - GAZ+ELEC HNR GAZOLE - SUPERETHANOL
Carrosserie	Type de carrosserie	object	0	not applicable	BERLINE - BREAK - CABRIOLET - COMBISPACE - COUPE - MINIBUS - MINISPACE - MONOSPACE - MONOSPACE - COMPACT - TS TERRAINS/CHEMINS
Gamme	Type de gamme	object	0	not applicable	ECONOMIQUE - INFERIEURE - LUXE - MOYENNE INFERIEURE - MOYENNE SUPERIEURE - SUPERIEURE
Puissance fiscale	Puissance en CV fiscaux	float	0	not applicable	min : 2.00 max : 91.00 mean : 10.27 median : 8 std : 8.57
Type de boite	Type de boite de vitesse	object	0	not applicable	AUTOMATIQUE MECANIQUE VARIATION CONTINUE VARIATION ELECTRIQUE
Masse OM Max	Masse en ordre de marche maxi (en Kg) Masse maximale d'un véhicule prêt à circuler (poids du véhicule, conducteur, passagers, cargaison, fluides)	integer	0	/	min : 935 max : 2785 mean : 1603.17 median : 1560 std : 308.21

Cependant, la variable cible contenait 3.71% de valeurs manquantes :

CO2 vitesse mixte Max	CO2 mixte combiné - maximum (en g/Km)	float	3.71	Supprimer les lignes des véhicules électriques Remplacer NAN par la médiane pour les lignes des véhicules hybrides et essence	min : 19.43 max : 448 mean : 154.80 median : 154.25 std : 40.94

Ceci est dû au fait que nous avions des véhicules électriques dans le dataset.

Notons également le fait que nous n'avions pas de valeurs aberrantes à corriger. Cependant, nous avons un grand nombre de valeurs extrêmes pour les variables suivantes (que l'on constate en comparant les valeurs minimales et maximales de chaque variable, nous avons de grandes plages entre ces deux valeurs) :

Puissance fiscale	**min : 2.00** **max : 91.00** mean : 10.27 median : 8 std : 8.57
Masse OM Max	**min : 935** **max : 2785** mean : 1603.17 median : 1560 std : 308.21

CO2 moyenne mixte Max	**min : 19.43** **max : 448** mean : 154.80 median : 154.25 std : 40.94

Encodage des variables catégorielles

Ensuite, nous avons encodé les variables explicatives catégorielles avec la technique “One Hot Encoder” afin que les modèles de régression puissent fonctionner.

Energie	Type d'énergie	object	0	not applicable	ELEC+ESSENC HR - ELEC+GAZOLE HR - ELECTRIC - ESS+ELEC HNR - ESS+G.P.L. - ESSENCE - GAZ+ELEC HNR GAZOLE - SUPERETHANOL
Carrosserie	Type de carrosserie	object	0	not applicable	BERLINE - BREAK - CABRIOLET - COMBISPACE - COUPE - MINIBUS - MINISPACE - MONOSPACE - MONOSPACE - COMPACT - TS TERRAINS/CHEMINS
Type de boite	Type de boite de vitesse	object	0	not applicable	AUTOMATIQUE MECANIQUE VARIATION CONTINUE VARIATION ELECTRIQUE

Mise à l’échelle des variables numériques

Suite au constat précédent, nous avons décidé d’utiliser la technique du Robust Scaling afin de redimensionner nos variables explicatives numériques. La technique utilise la médiane et l’écart intervalle interquartile, ce qui réduit l’impact des valeurs extrêmes sur nos modèles.

Tests et choix des modèles

Notre stratégie

Notre stratégie :

- **Utiliser la bibliothèque LazyPredict** pour nous donner un **aperçu des modèles** d'apprentissage supervisé **adéquats et de leur performance** ;
- **Choisir 3 modèles dits "naïfs"** (Linear Regression, Decision Tree Regressor et Random Forest Regressor), pour leur simplicité et rapidité d'implémentation et **3 modèles plus complexes** offrant potentiellement de meilleures performances (ExtraTrees Regressor, XGBoost Regressor, KNeighbors).

Les résultats de LazyPredict :

Model	Adjusted R-Squared
ExtraTreeRegressor	0.97
ExtraTreesRegressor	0.96
XGBRegressor	0.94
PoissonRegressor	0.90
GradientBoostingRegressor	0.90
DecisionTreeRegressor	0.90
RandomForestRegressor	0.89
LassoLarsIC	0.88
Lars	0.88
LinearRegression	0.88
TransformedTargetRegressor	0.88
BayesianRidge	0.88
LassoCV	0.88
RidgeCV	0.88
LarsCV	0.87
LassoLarsCV	0.87
HuberRegressor	0.87
SGDRegressor	0.87
KNeighborsRegressor	0.84
Lasso	0.84
PassiveAggressiveRegressor	0.82
ElasticNetCV	0.81
BaggingRegressor	0.80
LGBMRegressor	0.79
HistGradientBoostingRegressor	0.79
LinearSVR	0.78
AdaBoostRegressor	0.64
OrthogonalMatchingPursuitCV	0.58
ElasticNet	0.46
TweedieRegressor	0.34
GammaRegressor	0.33
OrthogonalMatchingPursuit	0.06
MLPRegressor	-0.05
NuSVR	-0.06
SVR	-0.06
LassoLars	-0.10
DummyRegressor	-0.42

<u>Les modèles les plus performants (ordre décroissant)</u>

1. **Extra Trees Regressor**

- Score de détermination (R2) :

```
Score sur les données d'entraînement : 0.9986428769802117

Score sur les données de test : 0.9692942855420406
```

- MAE, MSE et RMSE :

	MAE train	MAE test	MSE train	MSE test	RMSE train	RMSE test
Extra Trees	0.720741	7.042788	3.93945	108.214299	1.984805	10.40261

Interprétation :

Un score de détermination de 99% sur les données d'entraînement indique que le modèle capture presque toute la variabilité des valeurs cibles, c'est-à-dire qu'il montre très bien comment les différentes caractéristiques (features) influencent les émissions de CO2 (target). Un R2 de 96% est également un très bon score, cela signifie qu'il fonctionne bien sur les nouvelles données sans trop perdre de performance.

En étendant l'analyse avec les autres métriques, nous pouvons observer que le modèle Extra Trees a une petite tendance à l'overfitting mais a globalement de bien meilleures performances que les autres modèles.

C'est un modèle robuste qui est moins sensible aux valeurs extrêmes que le Random Forest ou le Decision Tree (grâce à la création de plusieurs arbres, une forêt, et l'introduction de méthodes aléatoires).

Pour notre problématique, la métrique MAE semble être la plus interprétable et adaptée, car elle est plus robuste aux valeurs extrêmes.

- Une MAE de 7 sur le jeu de test signifie qu'en moyenne, le modèle se trompe de 7 g/km dans ses prédictions du CO2 émis par kilomètre.
- Si nous rapportons ce résultat à l'échelle des données de notre target : une MAE de 7 g/km sur le jeu de test correspond à environ 2% de l'amplitude totale des valeurs, soit une erreur que nous pouvons considérer comme modérée, notamment pour les véhicules à fortes émissions mais pas pour l'inverse car l'erreur devient significative en proportion.
- En effet, pour une voiture n'émettant que 30 g/km, une erreur de 7 g/km représente environ 23% de la valeur réelle. La prédiction est donc moins fiable pour ce type de véhicule.

2. XGB Regressor

Il fonctionne en créant une séquence de modèles d'arbres de décision peu profonds et en les ajustant pour réduire l'erreur de prédiction. Chaque arbre tente de corriger les erreurs des arbres précédents en apprenant sur les erreurs faites précédemment.

Après avoir créé une instance de XGBRegressor et entraîné le modèle, nous avons pu mettre en place des indicateurs de performance :

- Score de détermination (R2) :

```
Score du modèle sur train: 0.9978894739730259

Score du modèle sur test: 0.9553231519597563
```

- MAE, MSE et RMSE :

	MAE train	MAE test	MSE train	MSE test	RMSE train	RMSE test
XGBoost	1.817522	9.267697	7.616692	196.239668	2.759836	14.008557

Interprétation :

Nous pouvons observer un overfitting du modèle. Cependant, il est moins sujet à une variance élevée car il utilise plusieurs arbres et une pondération pour améliorer les performances du modèle.

3. Random Forest Regressor

- Score de détermination (R2) :

```
Score R2 sur les données d'entraînement : 0.9876028297190369

Score R2 sur les données de test : 0.9532433786183869
```

- MAE, MSE et RMSE :

	MAE train	MAE test	MSE train	MSE test	RMSE train	RMSE test
Random Forest	3.982542	7.940337	46.71536	157.430517	6.834864	12.547132

Interprétation :

Au regard de tous les résultats, le modèle a ici aussi probablement sur appris les données d'entraînement et a du mal à généraliser sur de nouvelles données (phénomène d'overfitting). Nous le voyons sur toutes les métriques.

Les modèles les moins performants

- **Linear Regression**

  - Score de détermination (R2) :

```
Score du modèle sur train: 0.9224890067101257

Score du modèle sur test: 0.9129592572334037
```

- MAE, MSE et RMSE :

	MSE train	MAE train	RMSE train	MSE test	MAE test	RMSE test
Regression Lineaire	279.73	11.39	16.73	382.32	13.59	19.55

Interprétation :

Le modèle de régression linéaire semble bien ajusté et généralise correctement sur les données test. Les scores R^2 élevés et les erreurs relativement faibles sur les données de test montrent que le modèle est performant. Cependant, il y a une légère augmentation des erreurs sur les données de test RMSE test (19,55) par rapport aux données d'entraînement. Son score reste moins élevé que les autres modèles mais le modèle performe sur le test à peu près comme sur le jeu d'entraînement.

- **Decision Tree Regressor**

Est un arbre de décision permettant de résoudre un problème de régression.

Après avoir créé une instance de DecisionTreeRegressor et entraîné le modèle, nous avons pu mettre en place des indicateurs de performance :

- Score de détermination (R2) :

```
Score du modèle sur train: 0.998642965862055

Score du modèle sur test: 0.9267934532931416
```

- MAE, MSE et RMSE :

	MAE train	MAE test	MSE train	MSE test	RMSE train	RMSE test
DecisionTreeRegressor	0.856577	9.894621	4.89741	321.554206	2.213009	17.931933

Interprétation :

Nous pouvons observer un important overfitting du modèle. L'arbre de décision est très sensible à la variance des données. Ce modèle n'est pas adapté à notre jeu de données.

- **KNN**

- Score de détermination (R2) :

```
Score Manhattan sur les données d'entraînement : 0.9524501029361536

Score Manhattan sur les données de test : 0.9071805289457658
```

- MAE, MSE et RMSE :

	MSE train	MAE train	RMSE train	MSE test	MAE test	RMSE test
KNN Manhattan	27826.00	156.00	166.81	848.72	12.52	29.13

Interprétation :

Le modèle KNN avec le score Manhattan semble performant. Il l'est un peu moins sur le jeu de test mais a quand même un bon score (90,7%). On note que les erreurs sur les données de test sont beaucoup plus faibles que sur les données d'entraînement. Le KNN est connu pour convenir le plus à un jeu de données avec des valeurs extrêmes comme le nôtre mais un surapprentissage est visible avec les scores. Il est également étonnant de voir que le modèle a des métriques beaucoup moins élevées que pour le jeu d'entraînement.

Optimisations des modèles

Une fois les modèles entraînés une première fois, nous passons à l'étape de la recherche des hyperparamètres en vue de les améliorer davantage.

Modification de la standardisation et du jeu de test

En remplaçant le RobustScaler par le StandardScaler, nous avons obtenu de meilleures performances sur nos modèles.

Aussi, en passant d'une **répartition du jeu de test de 10% à 20%**, nous avons pu également réduire l'overfitting.

Modifications des hyperparamètres des modèles

Afin d'optimiser au maximum les hyperparamètres de nos modèles, nous nous sommes appuyées sur les deux techniques les plus connues :

- **la technique GridSearch :** elle consiste à effectuer une recherche sur une grille prédéfinie d'hyperparamètres pour identifier la combinaison qui produit les meilleures performances du modèle.

- **la technique RandomSearch :** elle va échantillonner des combinaisons d'hyperparamètres de manière aléatoire. Elle va ensuite évaluer ces combinaisons pour déterminer celles qui offrent les meilleures performances du modèle. Cette technique a l'avantage d'être plus rapide.

Les modèles les plus performants (ordre décroissant)

1. **Extra Trees Regressor**

Nous avons choisi de modifier ces principaux hyperparamètres pour améliorer ce modèle avec les valeurs testées ci-dessous :

- n_estimators : le nombre d'arbres à construire dans le modèle (100, 200, 300,500)
- max_depth : la profondeur maximale de chaque arbre (None,3,5,6,7,15,20)
- min_samples_split : nombre minimal d'échantillons requis pour diviser les noeuds (2,3,4,5,10,15,20)
- min_samples_leaf : nombre minimal d'échantillons requis pour être une feuille (1,2,3,4,6,8,10)
- max_features : nombre maximal de features à considérer pour trouver la meilleure division à chaque noeud ('auto' , 'sqrt', 'log2')
- bootstrap : entraînement de chaque arbre sur un échantillon aléatoire plutôt que sur l'ensemble des données (True, False)

La technique RandomSearch (100 combinaisons aléatoires d'hyperparamètres testées en 5 plis de validation croisée) nous a aidé à nous rapprocher des meilleures valeurs pour chaque hyperparamètre, puis nous avons affiné le paramétrage manuellement. La technique GridSearch a été écartée pour des raisons de temps de calcul.

La meilleure combinaison d'hyperparamètres est celle-ci : {'n_estimators': 300, 'min_samples_split': 5, 'min_samples_leaf': 1, 'max_features': 'auto', 'max_depth': None, 'bootstrap': False}

Voici les résultats :

	MAE train	MAE test	MSE train	MSE test	RMSE train	RMSE test	R² train	R² test
Hyperparamètres par défaut	0.720741	7.042788	3.939450	108.214299	1.984805	10.40261	0.998955	0.967861
Meilleurs hyperparamètres RandomSearch	3.453766	7.228525	31.303593	116.156229	5.594961	10.77758	0.991693	0.965502

```
Scores de validation croisée (hyperparamètres par défaut):  [0.93045444 0.93286144 0.92818891 0.92000615 0.90502452]
Moyenne des scores (hyperparamètres par défaut):  0.923307091061502
-----
Scores de validation croisée (meilleurs hyperparamètres RandomSearch): [0.94396617 0.93957878 0.92174918 0.93488253 0.91350252]
Moyenne des scores (meilleurs hyperparamètres RandomSearch): 0.9307358323939656
```

Le modèle avec les hyperparamètres par défaut présente globalement de meilleures performances notamment et principalement sur les données d'entraînement que le modèle avec modification des hyperparamètres. Cependant, ce dernier a réussi à améliorer la validation croisée (de 0.92 à 0.93).

2. XGB Regressor

Plusieurs hyperparamètres peuvent jouer sur la performance du modèle. Nous avons choisi d'optimiser les hyperparamètres avec les valeurs testées ci-dessous:

- n_estimators : le nombre d'arbres à construire dans le modèle (100, 200, 300, 400, 500).
- max_depth : la profondeur maximale de chaque arbre (3, 5, 6, 7).
- learning_rate : le taux d'apprentissage qui détermine la contribution de chaque arbre à la prédiction finale (0.01, 0.1, 0.2).
- subsample : la proportion d'échantillons à utiliser pour construire chaque arbre (0.5, 0.7, 1)
- colsample_bytree : la proportion de caractéristiques à utiliser pour chaque arbre (0.5, 0.7, 1)
- gamma : ce paramètre contrôle la réduction de la perte minimale requise pour faire une nouvelle séparation dans un arbre (0, 0.1, 0.2).

La technique GridSearch a identifié cette combinaison d'hyperparamètres : {'colsample_bytree': 0.5, 'gamma': 0, 'learning_rate': 0.2, 'max_depth': 3, 'n_estimators': 500, 'subsample': 1}

La technique RandomSearch a identifié cette combinaison d'hyperparamètres : {'subsample': 1, 'n_estimators': 400, 'max_depth': 3, 'learning_rate': 0.1, 'gamma': 0.1, 'colsample_bytree': 0.5}

Les valeurs des hyperparamètres 'gamma', 'n_estimators' et le learning_rate diffèrent entre les deux techniques.

Nous pouvons observer ci-dessous le tableau regroupant les résultats du modèle avec les hyperparamètres par défault et avec les hyperparamètres des deux techniques GridSearch et Random:

	MAE train	MAE test	MSE train	MSE test	RMSE train	RMSE test	R^2 train	R^2 test
Hyperparamètres par défaut	1.733251	8.100200	9.923657	172.885670	3.150184	13.148600	0.997366	0.948653
Meilleurs hyperparamètres GridSearch	3.570992	8.542331	22.995960	142.547452	4.795410	11.939324	0.993897	0.957664
Meilleurs hyperparamètres RandomSearch	5.538289	8.508673	52.957229	132.620302	7.277172	11.516089	0.985946	0.960612

La validation croisée va nous permettre d'évaluer la robustesse du modèle :

Modèle de base :

```
Scores de validation croisée du modèle de base (R²) : [0.88094801 0.92976136 0.92667874 0.91557869 0.87939847]
Moyenne des scores de validation croisée du modèle de base (R²) : 0.906473055091632
```

GridSearch :

```
Scores de validation croisée du modèle de base (R²) : [0.91756063 0.92302586 0.9280213  0.92513607 0.90672781]
Moyenne des scores de validation croisée du modèle de base (R²) : 0.9200943361437531
```

RandomSearch :

```
Scores de validation croisée du modèle de base (R²) : [0.91983988 0.9206954  0.93338897 0.91830964 0.87895458]
Moyenne des scores de validation croisée du modèle de base (R²) : 0.9142376965456609
```

Ce modèle est plus performant et robuste avec les hyperparamètres GridSearch.

3. Random Forest Regressor

Nous avons choisi de modifier ces principaux hyperparamètres pour améliorer ce modèle avec les valeurs testées ci-dessous :

- n_estimators : le nombre d'arbres à construire dans le modèle (50, 60, 70, 80,90,100,150)
- max_depth : la profondeur maximale de chaque arbre (40, 50, 60, 80,100)
- min_samples_split : nombre minimal d'échantillons requis pour diviser les noeuds (2,3,4,5)
- min_samples_leaf : nombre minimal d'échantillons requis pour être une feuille (1,2,3,4)
- max_features : nombre maximal de features à considérer pour trouver la meilleure division à chaque noeud ('sqrt', 5, 10,15,18)
- bootstrap : entraînement de chaque arbre sur un échantillon aléatoire plutôt que sur l'ensemble des données (True, False)

La technique RandomSearch (100 combinaisons aléatoires d'hyperparamètres testées en 5 plis de validation croisée) nous a aidé à nous rapprocher des meilleures valeurs pour chaque hyperparamètre, puis nous avons affiné le paramétrage manuellement.

La meilleure combinaison d'hyperparamètres est celle-ci : {'n_estimators': 68, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 14, 'max_depth': 68, 'criterion': 'squared_error', 'bootstrap': False}

Le GridSearch a été écarté pour des raisons de temps de calcul.

Nous avons réussi à optimiser le modèle en diminuant l'overfitting et en améliorant les performances du jeu de test comme en atteste le tableau ci-dessous :

	MAE train	MAE test	MSE train	MSE test	RMSE train	RMSE test	R^2 train	R^2 test
Hyperparamètres par défaut	3.982542	7.940337	46.715360	157.430517	6.834864	12.547132	0.987603	0.953243
Meilleurs hyperparamètres RandomSearch	0.720234	7.248038	3.939408	110.538850	1.984794	10.513746	0.998955	0.967170

La validation croisée indique que le modèle avec les meilleurs hyperparamètres est plus robuste et stable grâce à des performances plus homogènes sur les différents splits. Il dispose d'une meilleure capacité de généralisation.

```
Scores de validation croisée (hyperparamètres par défaut):  [0.92650416 0.89980946 0.82745515 0.93643731 0.84768897]
Moyenne des scores de validation croisée (hyperparamètres par défaut):  0.8875790095246996
-----
Scores de validation croisée (meilleurs hyperparamètres avec RandomSearch): [0.92314933 0.91907857 0.84019721 0.94797132 0.87573089]
Moyenne des scores de validation croisée (meilleurs hyperparamètres avec RandomSearch): 0.9012254612962834
```

Le modèle Random Forest avec les meilleurs hyperparamètres est donc à privilégier par rapport au modèle avec les hyperparamètres par défaut.

4. Decision Tree Regressor

L'hyperparamètre à optimiser sur ce modèle est la profondeur maximale de l'arbre (max_depth). Pour chacune des techniques d'optimisation, nous avons défini une plage assez large (de 1 à 99) de cet hyperparamètre pour identifier la valeur la mieux adaptée à notre modèle.

La technique GridSearch a identifié un nombre optimal de profondeur d'arbre à 18.

La technique RandomSearch a identifié un nombre optimal de profondeur d'arbre à 13.

Nous pouvons observer ci-dessous le tableau regroupant les résultats du modèle avec les hyperparamètres par défault et avec les hyperparamètres des deux techniques GridSearch et Random:

	MAE train	MAE test	MSE train	MSE test	RMSE train	RMSE test	R^2 train	R^2 test
Hyperparamètres par défaut	0.720234	7.091136	3.939408	110.007399	1.984794	10.488441	0.998955	0.967328
Meilleurs hyperparamètres GridSearch	0.848006	7.080773	4.471703	107.061186	2.114640	10.347038	0.998813	0.968203
Meilleurs hyperparamètres RandomSearch	1.763950	7.104602	11.288692	108.384948	3.359865	10.410809	0.997004	0.967810

Nous pouvons observer de meilleurs résultats avec la technique GridSearch. L'optimisation de la mise à l'échelle et de la répartition du jeu de test nous a permis de réduire l'overfitting de manière conséquente.

La validation croisée va nous permettre d'évaluer la robustesse du modèle:

Modèle de base :

```
Scores de validation croisée du modèle de base : [0.92528947 0.8468379  0.5340663  0.9482722  0.89923907]
Moyenne des scores de validation croisée de base  : 0.8307409879055443
```

GridSearch :

```
Scores de validation croisée du modèle optimisé GridSearch : [0.92833005 0.85140142 0.90349691 0.95427384 0.90769084]
Moyenne des scores de validation croisée du modèle optimisé GridSearch  : 0.90903861160572
```

RandomSearch :

```
Scores de validation croisée du modèle optimisé RandomSearch : [0.91879469 0.84870799 0.90411773 0.95043725 0.89682528]
Moyenne des scores de validation croisée du modèle optimisé RandomSearch  : 0.9037765893329682
```

Ces résultats indiquent que le modèle avec les hyperparamères par défaut présente une grande variation sur les 5 échantillons de données, avec un score particulièrement bas (0.5341). Les techniques ont permis d'optimiser fortement la robustesse du modèle.

Le modèle DecisionTreeRegressor est plus performant avec les hyperparamètres définis par GridSearch.

Les modèles les moins performants

- **Linear Regression**

Le Grid Search ne permet pas d'améliorer le modèle, il est déjà à sa capacité optimale avec les hyperparamètres par défaut.

- **KNN**

Nous avons testé les différents scores du KNN et notre score choisi initialement (i-e Score de Manhattan) reste le score le plus élevé.

```
Score Manhattan : 0.9345387522276247

Score Minkowski : 0.9000516169438788

Score Euclidienne : 0.9000516169438788

Score Tchebyshev : 0.8172321950235344

Score Mahalanobis : 0.8649867685789077
```

En utilisant le Grid Search pour optimiser le modèle, il propose les meilleurs paramètres suivants:

```
Meilleurs paramètres :  {'metric': 'manhattan', 'n_neighbors': 3}

Meilleur score :  0.885216403454509
```

Nous constatons que sur un jeu de donnée différent le meilleur score serait de 0,88, mais sur notre jeu de donnée spécifique le modèle est plus performant avec le neighbors = 5. L'objectif est de pouvoir utiliser le modèle sur de nouveaux jeux de données donc nous ne garderons pas le KNN d'autant plus que les métrics ne s'améliorent pas non plus.

	MSE train	MAE train	RMSE train	MSE test	MAE test	RMSE test
KNN Manhattan	27834.994025	155.526648	166.838227	220.409384	8.938737	14.846191

Choix du modèle final

Voici un tableau récapitulatif des 3 modèles les plus performants :

	Modèles	MAE Train / Test	MSE Train / Test	RMSE Train / Test	R2 Train / Test	Validation Croisée
Hyperparamètres par défaut	**Extra Trees Regressor**	**0,72 / 7,04**	**3,93 / 108,2**	**1,98 / 10,40**	**0,99 / 0,96**	**0,9233**
	XGB Regressor	1,73 / 8,10	9,92 / 172,88	3,15 / 13,14	0,99 / 0,94	0,9064
	Random Forest Reg.	3,98 / 7,94	46,71 / 157,43	6,83 / 12,54	0,98 / 0,95	0,8875
Hyperparamètres modifiés	Extra Trees Regressor	3,45 / 7,22	31,30 / 116 ,15	5,59 / 10,67	0,99 / 0,96	0,9307
	XGB Regressor	3,57 / 8,54	22,99 / 142,54	4,79 / 11,93	0,99 / 0,95	0,9200
	Random Forest Reg.	0,72 / 7,24	3;93 / 110,53	1,98 / 10, 51	0,99 / 0,96	0,9012

Suite à nos différents tests, nous pouvons observer que le modèle le plus robuste et performant est l'**Extra Trees Regressor avec ses hyperparamètres par défaut**. Après plusieurs tests, le modèle s'est montré plus fiable.

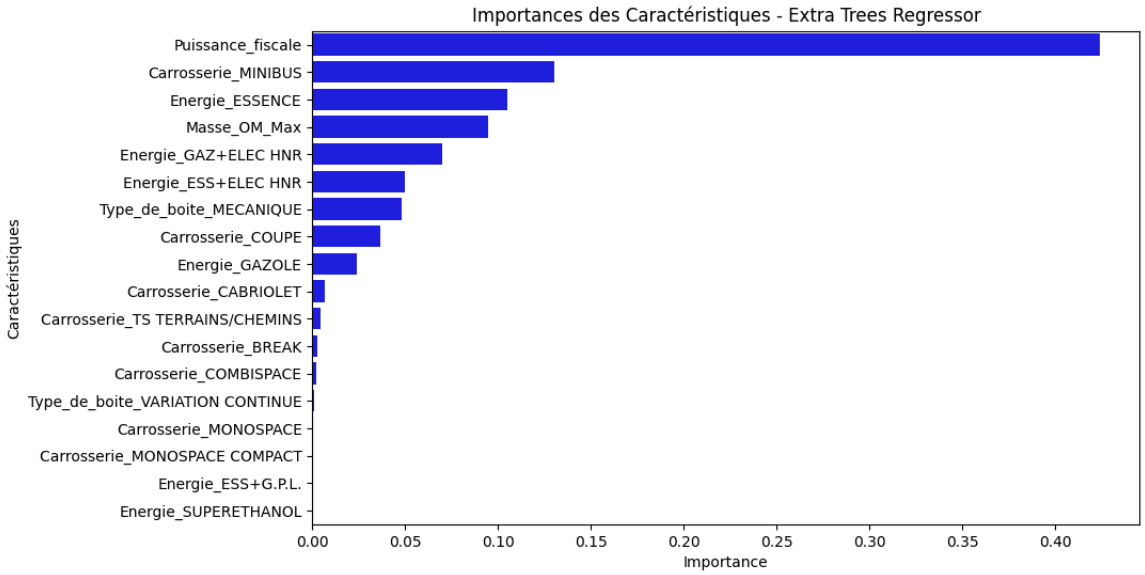
Interprétation

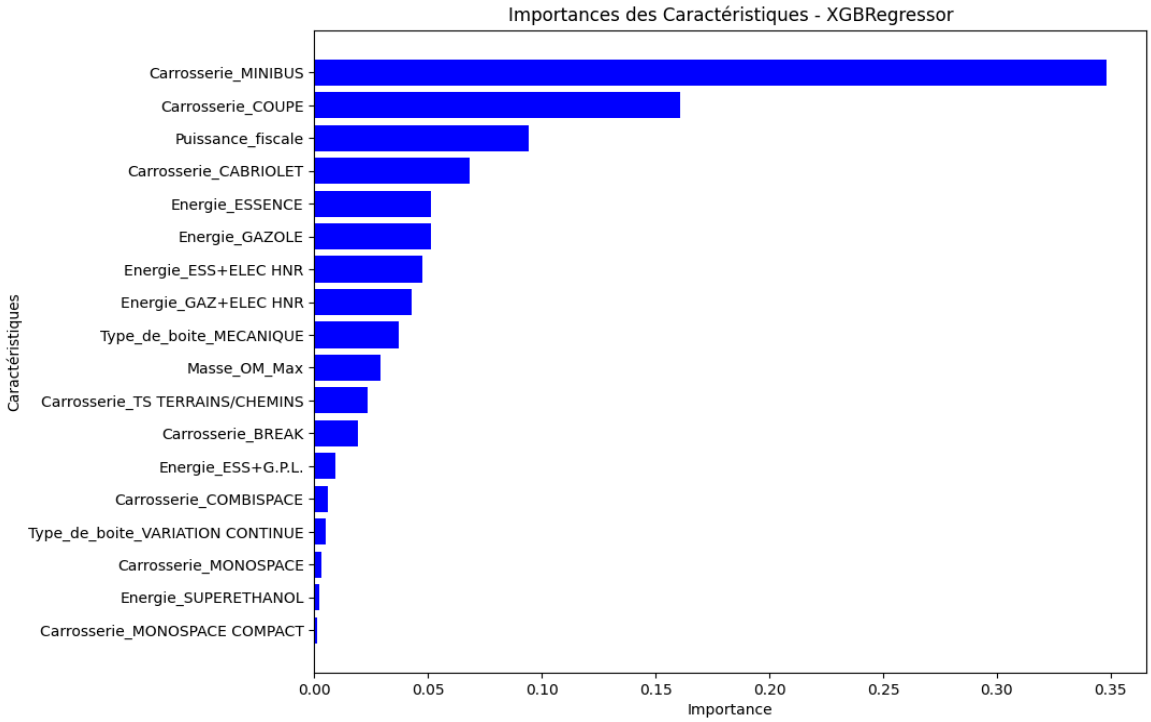
Avec le modèle **Extra Trees Regressor**, le Features Importance met en évidence la **Puissance fiscale qui est la caractéristique qui influe le plus dans les émissions de CO2** des véhicules (poids supérieur de 40%).

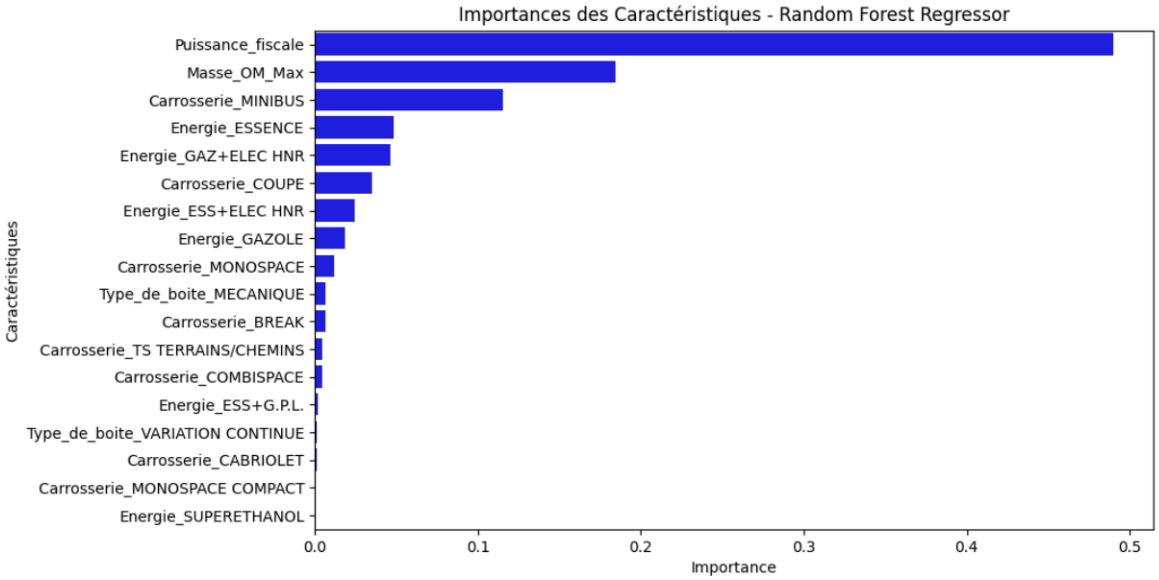
Nous avons comparé les caractéristiques et leur importance pour les 3 modèles les plus performants. En cumulant l'importance de ces caractéristiques sur les 3 modèles, les 3 caractéristiques les plus importantes sont la **Puissance fiscale,** la **Carrosserie Minibus et la Masse OM Max**. La caractéristique **Energie Essence** suit de près les précédentes caractéristiques.

Top 5 des caractéristiques				
Caracteristiques	ExtraTrees	XGB	RandomForest	Total
Puissance _Fiscale	0,4243	0,0944	0,4903	1,01
Carrosserie_ Minibus	0,1303	0,3484	0,1156	0,59
Masse_OM_Max	0,0945		0,1842	0,28
Energie_Essence	0,105	0,0512	0,0478	0,20
Carrosserie_Coupe		0,1608		0,16
Energie_Gaz_Elec_HNR	0,0699		0,0462	0,12
Carrosserie_Cabriolet		0,0682		0,07

Résultats de la Features Importance pour les 3 meilleurs modèles




Importances des Caractéristiques - XGBRegressor
Carrosserie_MINIBUS
Carrosserie_COUPE
Puissance_fiscale
Carrosserie_CABRIOLET
Energie_ESSENCE
Energie_GAZOLE
Energie_ESS+ELEC HNR
Energie_GAZ+ELEC HNR
Type_de_boite_MECANIQUE
Masse_OM_Max
Carrosserie_TS TERRAINS/CHEMINS
Carrosserie_BREAK
Energie_ESS+G.P.L.
Carrosserie_COMBISPACE
Type_de_boite_VARIATION CONTINUE
Carrosserie_MONOSPACE
Energie_SUPERETHANOL
Carrosserie_MONOSPACE COMPACT
Caractéristiques
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
Importance


Importances des Caractéristiques - Random Forest Regressor
Puissance_fiscale
Masse_OM_Max
Carrosserie_MINIBUS
Energie_ESSENCE
Energie_GAZ+ELEC HNR
Carrosserie_COUPE
Energie_ESS+ELEC HNR
Energie_GAZOLE
Carrosserie_MONOSPACE
Type_de_boite_MECANIQUE
Carrosserie_BREAK
Carrosserie_TS TERRAINS/CHEMINS
Carrosserie_COMBISPACE
Energie_ESS+G.P.L.
Type_de_boite_VARIATION CONTINUE
Carrosserie_CABRIOLET
Carrosserie_MONOSPACE COMPACT
Energie_SUPERETHANOL
Caractéristiques
0.0
0.1
0.2
0.3
0.4
0.5
Importance

Conclusion

Après la modélisation de 6 modèles et le choix des meilleurs hyperparamètres, **nous avons retenu le modèle Extra Trees Regressor (avec ses hyperparamètres par défaut)** en raison de ses **excellentes performances**.

Le modèle Extra Trees Regressor dont les hyperparamètres ont été modifiés grâce aux recommandations du Random Search et malgré son meilleur score de validation croisée, s'est montré décevant quant à ses capacités de prédictions, beaucoup moins précises, lors de notre démonstration.

En effet, afin d'évaluer concrètement la capacité de prédiction et de généralisation du modèle retenu, et de valider l'exactitude du classement du Features Importance, **nous avons réalisé une démonstration finale avec les données de notre dataset puis avec de nouvelles données**.

Cette démonstration s'est d'abord déroulée avec une sélection de véhicules présents dans notre dataset (dont nous connaissions donc les émissions de CO2). Cette étape étant concluante, nous avons poursuivi notre démonstration avec de nouveaux véhicules dont nous connaissions également les émissions de CO2. Cette dernière étape fut également une réussite, ce qui conforte **la solidité de notre modèle** et notre **pleine confiance en ses capacités de prédictions**.

- **Nous recommandons donc aux constructeurs, d'apporter une attention particulière à la Puissance fiscale, au niveau d'équipement du véhicule** (Masse OM MAX - en ordre de marche maximum) et à **l'énergie Essence**. Les **véhicules de gros volume**, telle que la **carrosserie Minibus**, ne sont également pas recommandés dans le cadre d'un objectif de réduction des émissions de CO2.

- **Ainsi, ce modèle permettra aux constructeurs désireux de maîtriser au maximum l'émission de CO2 de leur nouveau véhicule, d'ajuster avec précision chaque caractéristique influente. L'enjeu est**

important car il leur permettra d'éviter un potentiel malus écologique, qui nous le savons, impacte le prix du véhicule et donc son achat par les consommateurs.

En guise d'**amélioration du modèle**, nous pourrions **augmenter le volume du dataset avec des données futures**, par exemple de 2024 et au-delà, afin d'**améliorer sa robustesse**.

Enfin, nous souhaitons terminer ce rapport avec **une conclusion plus personnelle**. Tout d'abord, nous avons été surprises par la qualité des prédictions de notre modèle et plus largement impressionnées par la puissance du Machine Learning. Ce projet nous a permis de passer de concepts très abstraits, abordés lors des différents sprints, à une application très concrète et à des réalités professionnelles.

Nous nous sommes plus que jamais rendues compte que la conduite d'un tel projet nécessite une solide base de compétences techniques, de la rigueur mais aussi beaucoup de curiosité, d'assiduité et de ténacité pour arriver à des résultats concluants. Par ailleurs, nous tenons à souligner l'importance de la dimension humaine pour nous toutes. Le projet fut l'occasion de nous exercer à la conduite de projets data en groupe, et en distanciel, mais aussi d'échanger sur la formation, de confronter nos idées et de nous encourager mutuellement.

Pour finir, **nous tenons à remercier vivement Eliott Douieb pour la qualité de son accompagnement** tout au long de ce projet.