

Estimation du prix de véhicules d'occasion

L'objectif est de prédire le prix de véhicules d'occasion à partir d'un fichier d'annonces de vente de 166,695 véhicules. Chaque annonce contient :

- la marque du véhicule;
- le modèle;
- son année d'édition;
- la date de mise en ligne de l'annonce;
- le kilométrage;
- la boîte de vitesse;
- la motorisation;
- une description du modèle avec sa version et ses options;
- et le prix de vente, à prédire.

Les différents traitements de données ainsi que les études de modèles sont effectués dans un Jupyter notebook, disponible dans ce dossier Github : github.com/dilva/used_car_estimator.
Le notebook est organisé du traitement de données à la recherche d'un meilleur score.

Ce document rapporte la démarche en trois parties. Premièrement, les approches liées au métier sont détaillées, puis les choix techniques, et enfin les résultats obtenus.

Sommaire

I. Approche métier	3
A. Traitements des données disponibles	3
• <i>Marque et modèle du véhicule</i>	3
• <i>Année du véhicule (Année) et date de mise en ligne (Online)</i>	3
• <i>Motorisation (Fuel)</i>	3
• <i>Kilométrage (Mileage)</i>	4
• <i>Boîte de vitesse (Gearbox)</i>	4
• <i>Description</i>	4
B. Équilibre des données	4
C. Enrichissement	6
• <i>Segment</i>	6
• <i>Prix du neuf</i>	6
II. Choix techniques	7
A. Encodage des variables catégorielles	7
B. Données textuelles	7
• <i>One-hot encoding de la présence d'une option</i>	7
• <i>Count vectorizer et Tf-idf</i>	7
C. Modèle et mesures de performance	7
• <i>Modèle de régression</i>	7
• <i>Mesures de performance</i>	8
III. Résultats	8
A. Variance des résultats	8
B. Importance des features	9
C. Résultats des traitements de données	10

I. Approche métier

Le projet concerne le secteur automobile. Le choix des traitements de données à appliquer découle de sa connaissance. Chaque information disponible dans le jeu de données doit être comprise avant d'être utilisée comme variable d'un modèle. Il peut aussi être utile d'apporter des données extérieures, ce qui est facilité avec connaissance du secteur.

A. Traitements des données disponibles

- Marque et modèle du véhicule

À première vue, le jeu de données contient 92 marques de véhicules (*Marque*) différentes et 846 modèles (*Modèle*) uniques.

Certaines marques sont aujourd'hui regroupées («*McLaren*», «*Mercedes*», «*Mercedes - AMG*») ou bien n'existe plus, d'autres sont des mariages entre marques et séries (*BMW*, *BMW-Alpine*). Ces cas de marques pourrait être regroupés mais ne le sont pas ici. Il aurait fallu une meilleure connaissance des marques automobiles pour choisir rapidement quelles marques rassembler, et ces marques ne représentent pas un grand volume de données. Pour le modèle, **ces marques sont encodées de 0 à 91** (cf. encodage des données catégorielles).

Le même principe est appliqué à la colonne *Modèle*, les valeurs **sont encodées de 0 à 845**. Certains modèles sont appelés par des chiffres, et pourraient être redondant d'une marque à l'autre. On utilise ici un troisième encodage, qui regroupe la marque et le modèle. Le modèle est issue de la colonne *Description*, qui contient une appellation du modèle parfois plus précise (*124 (2E GENERATION) SPIDER* plutôt que *124*). **On obtient 1810 couples marque-modèle encodés aussi de 0 à 1809** (*Marque&modèle*).

- Année du véhicule (*Année*) et date de mise en ligne (*Online*)

La majorité des véhicules des annonces datent des années 2010. Quelques années sont fausses (1900 et 5018) mais concernent peu de véhicules. L'année 5018 est transformée en 2018 après avoir vérifié la moyenne des années du modèle, et les 12 véhicules datés de 1900 ont été corrigés à la main.

La deuxième indication de temps est la date de mise en ligne d'une annonce, au format *jj/mm/aaa h:mm*. La première intuition est de transformer cette date en timestamp et de la comparer à la date d'édition du véhicule, afin d'obtenir l'âge du véhicule lors de la vente (*Age*). Le calcul des premières corrélations entre les variables montre que les corrélations *Prix/Age* et *Prix/Année* sont identiques, et que les corrélations *Age/Année* et *Année/Age* sont complémentaires. En effet **toutes les dates de mises en lignes sont de 2018. Les écarts d'année d'édition entre les véhicules du jeu de données sont donc les mêmes que les écarts d'âge lors de la publication de l'annonce.** Pour la suite, **on se concentre sur la variable *Année*.**

- Motorisation (*Fuel*)

La motorisation peut prendre 8 valeurs : Diesel, électrique, essence, Hybride diesel ou essence - électrique, et 3 bi-carburant essence différente. La majorité des véhicules sont des diesels ou à essence. On aurait donc pu regrouper les 6 autres catégories en une variable « autre ». Seulement le prix des motorisations électriques est souvent plus élevé que les autres. **On regroupe donc les 3 bi-carburant en une « bi-carburant essence ». Les catégories sont numérisées par one-hot encoding.**

- Kilométrage (*Mileage*)

Il suffit ici de transformer le texte sous format « *x km* », en une variable flottante.

- Boîte de vitesse (*Gearbox*)

Il suffit ici d'encoder les deux valeurs que prend cette donnée (automatique : 0, mécanique : 1)

- Description

Les descriptions sont composées de 6 champs : modèle, version, puissance fiscale, portes, couleur et options.

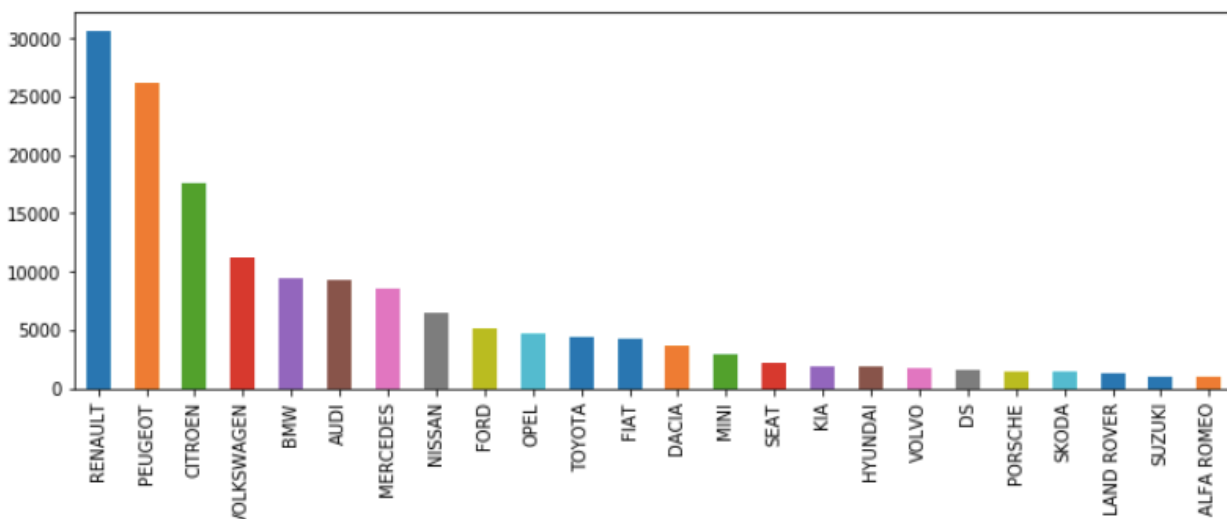
Les champs modèle (*Description_modèle*), versions (*Version*), couleur et options (*Options*) sont extraits dans de nouvelles colonnes tels quels.

- Comme décrit précédemment, le modèle est utilisé pour créer un nouveau champ plus précis d'identification du véhicule (*Marque&modèle*).
- Les versions contiennent plus d'informations sur les véhicules, des détails sur le moteur (*1.6 HDI FAP 92CH*) ou sa classe (*Allure*). Deux données permettant de décrire efficacement la puissance d'un véhicule sont la cylindrée et le nombre de chevaux, et on retrouve ces données dans la majorité des descriptions. On extrait donc le texte sous la forme **xx.x ou x.x pour la cylindrée**, puis tous les **xxx ou xx accompagnés ou non de CH, HP, H, AH, R ou Q, pour le nombre de chevaux** (*x étant un chiffre*). **AH, R et Q sont des indications proche du nombre de chevaux pour les véhicules électriques** (non équivalentes mais la conversion reste proche).

Le texte des options devait être un champ libre lors de la création des annonces. Il n'y a pas de cohérence d'un véhicule à l'autre. Les options sont traités avec des méthodes de compréhension de texte (cf. NLP/NLU du texte). Après le nettoyage des options, il reste 52,198 options uniques. Une majorité d'entre elles n'apparaissant que quelques fois dans les annonces. **Ici, on ne sélectionne que les 30 options qui apparaissent dans plus de 15,000 annonces (~10%).**

B. Équilibre des données

Aucune variable n'est vraiment équilibrée, de la distribution des marques :

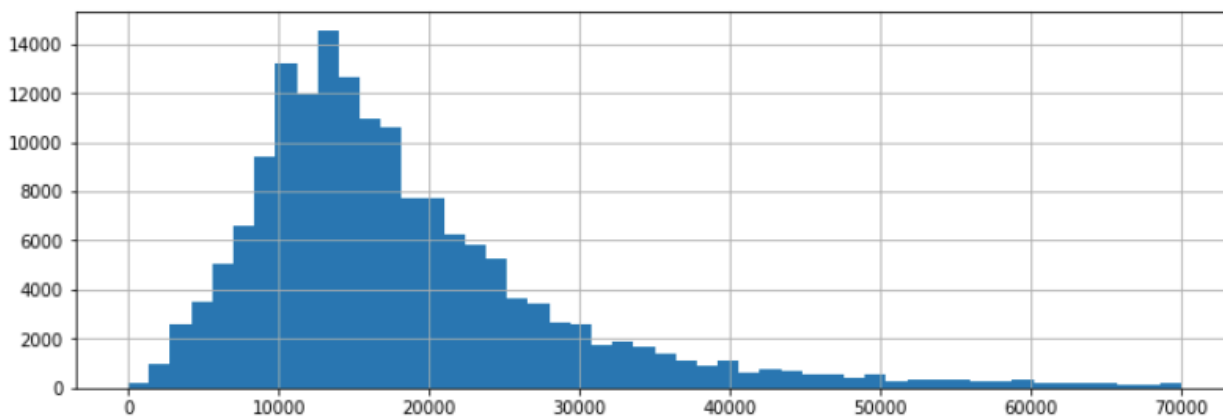


à la répartition des modèles par marque.

Exemple du nombre de véhicules par modèle pour RENAULT :

Make	Model	
RENAULT	CLIO	7967
	MEGANE	5212
	CAPTUR	4410
	SCENIC	3045
	KADJAR	2269
	TWINGO	2124
	GRAND SCENIC	1588
	ESPACE	943
	TALISMAN	897
	KANGOO	551
	LAGUNA	545
	KOLEOS	352
	ZOE	325
	MODUS	164
	GRAND MODUS	89
	GRAND ESPACE	57
	GRAND KANGOO	52
	LATITUDE	29
	VEL SATIS	14
	TWIZY	12
	FLUENCE	11
	WIND	9
	R5	6
	AVANTIME	4
	R4	3

Le prix de vente à prédire est aussi très déséquilibré, avec une asymétrie positive autour de 15,000€, et des valeurs extrêmes de 300,000 à 1M€.




Ces distributions représentent la réalité du parc automobile et sont donc conservées dans les données à présenter au modèle. Quelques outliers créent sûrement du bruit, mais leur retrait n'ont pas grandement influencé les performances du modèle. Toutes les données sont conservées pour la suite.

C. Enrichissement



On peut ajouter des données afin d'affiner les performances du modèle, comme les **prix des véhicules neufs** comme proposé pendant le cours, ou encore **le segment du véhicule** (citadine, berline, sportive, etc.)

• Segment

L'idée vient de la variété et du grand nombre de marques et modèles. Les prix des véhicules correspondent souvent à un rapport entre la marque et la classe du véhicule, comme deux compacte de marques opposées, ou une berline et une compacte de la même marque.




DS 3

Marque  Citroën
 DS Automobiles

Années de production Citroën : 2009 - 2016
Phase 1 : 2010 - 2014
Phase 2 : 2014 - 2016
DS Automobiles : 2016 - 2019

Production 500 000¹ exemplaire(s)

Classe Petite citadine

Usine(s) d'assemblage  Poissy

Les classes des modèles sont affichées explicitement sur les pages Wikipédia des modèles de véhicule. Pour récupérer les segments de chaque véhicule, on interroge la page Wikipédia de chaque valeur de *Marque&modèle* unique. La vingtaine de valeurs récupérées peuvent être regroupées en 10 segments suivants :

- micro citadine
- mini citadine;
- citadine;
- compactes;
- grande routières;
- SUV
- sportive
- luxe
- prestige
- supercar

Chaque segment est ensuite encodé de 0 à 9.

• Prix du neuf

De la même manière, on récupère le prix neuf d'un couple *Marque&modèle* en interrogeant la page de détail du modèle sur autoplus.fr. Par contrainte de temps, le prix ne correspond pas forcément rigoureusement au modèle de la voiture. Il correspond à la **moyenne des prix affichés sur le site**, s'il est supérieur au prix d'occasion. Sinon, le **prix du neuf est estimé par rapport au prix des véhicules neufs de la même marque et du même segment**. De cette manière on obtient près de 85% des prix neufs.

MODÈLE	VERSION	ÉNERGIE	BOÎTE DE VITESSE	PUISS. FISCALE	DATE ENTRÉE	MALUS	PRIX À PARTIR DE...
Peugeot 308	1.2i 12V 110 Access (5p.)	Essence	Manuelle	5 CV	06/2019	0 €	21 650 €
Peugeot 308 SW	1.2i 12V 110 Access (Break)	Essence	Manuelle	5 CV	06/2019	0 €	22 600 €
Peugeot 308 Société	1.5 BlueHDi 100 BVM6 Premium (5p.)	Diesel	Manuelle	5 CV	04/2018	0 €	22 860 €
Peugeot 308	1.5 BlueHDi 100 BVM6 Access (5p.)	Diesel	Manuelle	5 CV	04/2018	0 €	23 500 €
Peugeot 308	1.2i 12V 110 Active (5p.)	Essence	Manuelle	5 CV	06/2019	0 €	24 000 €

Pour ces deux méthodes de scraping, la meilleure page possible a été trouvée par un moteur de recherche, en proposant les mots « *[marque] [modèle] wikipédia* » et « *[marque] [modèle] autoplus prix neuf* »

II. Choix techniques

A. Encodage des variables catégorielles

- One-hot encoding ou encodage ordinal ?

Seules les données de motorisation (essence, diesel, électrique, bi-carburant, et les deux hybrides) et celles de la boîte de vitesse (automatique ou mécanique) ont été encodées avec la méthode du one-hot encoding. Pour la boîte de vitesse, cela permet de ne garder qu'une colonne, l'autre étant complémentaire. Le nombre de catégories de motorisation (6) reste assez suffisant pour ne pas être discriminé par les forêts aléatoires, le modèle choisi par la suite.

Les marques, modèles, couples marques-modèles, ainsi que les segments ont été encodées avec des entiers. Les marques ont été encodées ordinairement (les marques apparaissant le plus avec une valeur proche de zéro), et les modèles par ordre alphabétique. Le premier encodage affiche une corrélation importante avec le prix, due à la volumétrie de ces marques, mais les prédictions du modèle ne se trouvent pas changées par l'une ou l'autre technique d'encodage.

Le nombre de marque et de modèles est très important, et les encoder de façon binaire créerait une matrice de données conséquente. De plus, une forêt aléatoire cherche à séparer les données par leurs valeurs. Avec autant de catégories, séparer les données à partir d'une marque n'isolerait qu'une marque parmi les autres, soit un petit nombre de données. **Les variables des marques pourraient donc être considérées comme peu importantes par le modèle.**

B. Données textuelles

Les options d'un véhicule sont décrites par des valeurs remplies à la main, avant nettoyage, il y en a plus de 60,000 différentes. Pour réduire leur nombre, les options sont mises en minuscules, les accents, symboles et mots courants sont retirés, les mots sont tronqués, et rangés par ordre alphabétique. Ainsi « *Banquette arrière 3 places* » devient « *3 arrier banquet plac* ». De cette manière, on remarque qu'une minorité d'options est présente dans un grand nombre de données. Un minimum de 10% de présence est conservé, réduisant à 30 options récurrentes. La mise en valeur de ces 30 options a été testée avec ces 3 méthodes.

- One-hot encoding de la présence d'une option

Pour chaque donnée, les colonnes OPTION_1 à OPTION_30 prennent la valeur 1 si l'option est présente, 0 sinon.

- Count vectorizer et Tf-idf

Ici, on s'intéresse directement aux mots présents dans ces options, au nombre réduit de 56 pour 30 options.

Pour chaque donnée, les colonnes OPTION_1 à OPTION_56 prennent la valeur X quand l'option est présente X fois, pour le Count Vectorizer, ou le poids du mot dans le dictionnaire calculé avec Tf-idf.

De ces trois méthodes, le Count Vectorizer a apporté les meilleures corrélations. Mais les encodages de cette manière peuvent rendre presque insignifiantes ces catégories. **Ces traitements ont eu peu d'influence positive sur le modèle, et on ajouté du bruit.**

C. Modèle et mesures de performance

- Modèle de régression

Pour ce problème, on choisit une régression par un RandomForest, pour les raisons suivantes :

- Il est efficace sur les grand volumes de données;
- avec un grand nombre de données, on augmente le nombre d'arbre ce qui rend le modèle résistant au sur-apprentissage;
- il n'est pas trop sensible à un petit nombre d'outliers ou de données manquantes, ou à la variété des valeurs prises par les données (plusieurs tests avec RobustScaler ou MinMaxScaler, n'ont pas particulièrement amélioré les prédictions);
- il nécessite très peu d'optimisation de paramètres (par opposition à un essai d'un AdaBoost, moins performant avec ces paramètres par défaut.)

Toutes les prédictions ont donc été effectuées avec le RandomForestRegressor de Scikit-learn, avec 100 arbres, la valeur par défaut. En cherchant de meilleurs paramètres pour le nombre d'arbres et leur profondeur avec une Grid Search, il n'y avait que très peu d'amélioration (variance de 10 pour une MAE de 1650).

• Mesures de performance

La véracité des prédictions du RandomForestRegressor ont été mesurées dans un premier temps avec la mesure MAPE, puis opposées à la mesure MAE (cf. Variance des résultats)

III. Résultats

A. Variance des résultats

Résoudre ce problème avec un RandomForest devait nous faire arriver rapidement à une MAPE de 10%, mais avec la difficulté de la faire diminuer ensuite.

En créant un modèle de référence à partir d'un petit nombre de variables, afin d'avoir un score à améliorer, on obtient toujours une MAPE de près de 60% par cross validation. Le score ne s'améliore pas même après l'essai de différentes méthodes : suppression des valeurs extrêmes, normalisation des données, changement des encodages, suppression de certaines variables, changement de modèles.

Résultats de la première cross-validation :

```
CV1 (133017, 13)(33255, 13)
MAPE : 9.44%, MAE : 1703.41

CV2 (133017, 13)(33255, 13)
MAPE : 9.40%, MAE : 1701.36

CV3 (133018, 13)(33254, 13)
MAPE : 201.82%, MAE : 1720.78

CV4 (133018, 13)(33254, 13)
MAPE : 25.29%, MAE : 1663.39

CV5 (133018, 13)(33254, 13)
MAPE : 48.67%, MAE : 1732.99

-----
MAPE : 58.93%, MAE : 1704.39
```

En observant les résultats de chaque fold de la cross-validation, certains segments de données produisent une MAPE de ~10%, quand d'autres atteignent 200%. Pourtant, la MAE varie très peu. En échangeant la répartition des données d'entraînement, on peut obtenir 4 folds à 10% de MAPE, avant le dernier à 600%.

La mesure MAPE est très sensible à la variance des erreurs de prédictions, et cette erreur se propage inévitablement dans le jeu de données.

La cross-validation est indispensable ici, pour avoir une bonne compréhension de la mesure MAPE, à moins de la confronter à une autre mesure.

En se concentrant uniquement sur les résultats du jeu de test, on peut ne pas observer le comportement du modèle sur les différents ordres de grandeurs des prix de véhicules.

Il suffit par exemple de fixer le tirage des données sur un échantillon avantageux pour obtenir une bonne MAPE.

- Ici, avec un random state à 40 :

```

j: 1 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=40)
2 print(X_train.shape, X_test.shape)
3 rf = RandomForestRegressor(n_jobs=-1, random_state=0)
4 rf.fit(X_train, y_train)
5 predictions = rf.predict(X_test)
6 print("MAPE : {:.2f}%, MAE : {:.2f}".format(mape(y_test, predictions), mae(y_test, predictions)))

```

executed in 21.2s, finished 20:14:17 2020-05-23

(111402, 12) (54870, 12)
MAPE : 116.32%, MAE : 1770.64

- random state à 0 :

```

l: 1 RANDOM_STATE = 0
2 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=RANDOM_STATE)
3 print(X_train.shape, X_test.shape)
4 rf = RandomForestRegressor(n_jobs=-1, random_state=0)
5 rf.fit(X_train, y_train)
6 predictions = rf.predict(X_test)
7 print("MAPE : {:.2f}%, MAE : {:.2f}".format(mape(y_test, predictions), mae(y_test, predictions)))

```

executed in 24.1s, finished 20:40:21 2020-05-23

(111402, 13) (54870, 13)
MAPE : 9.54%, MAE : 1721.08

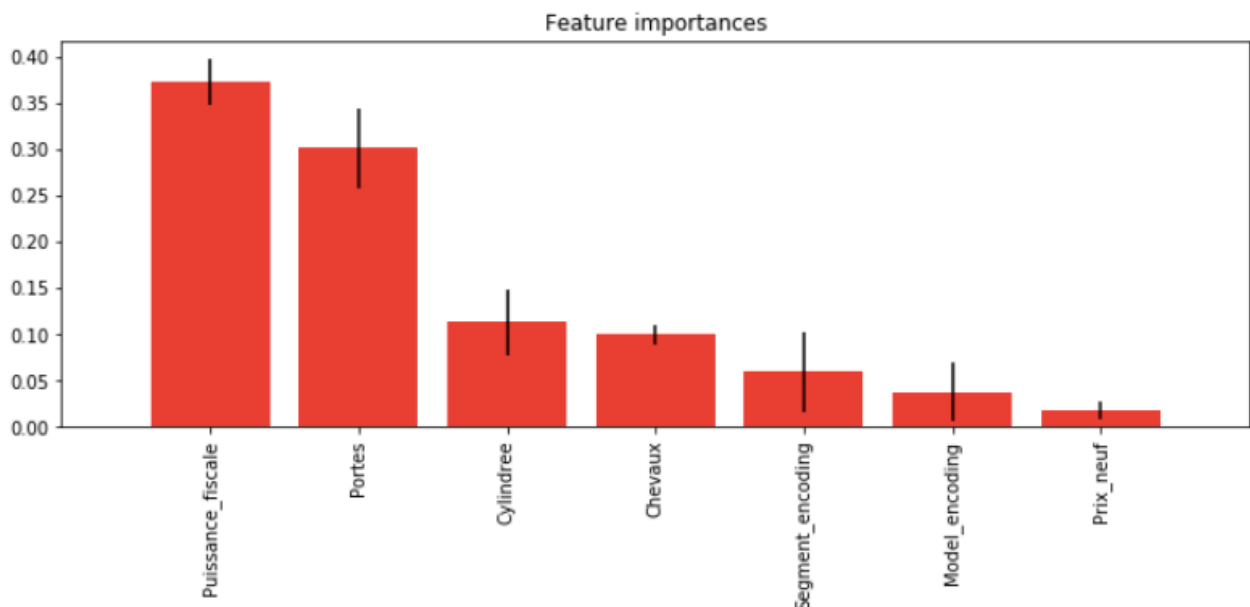
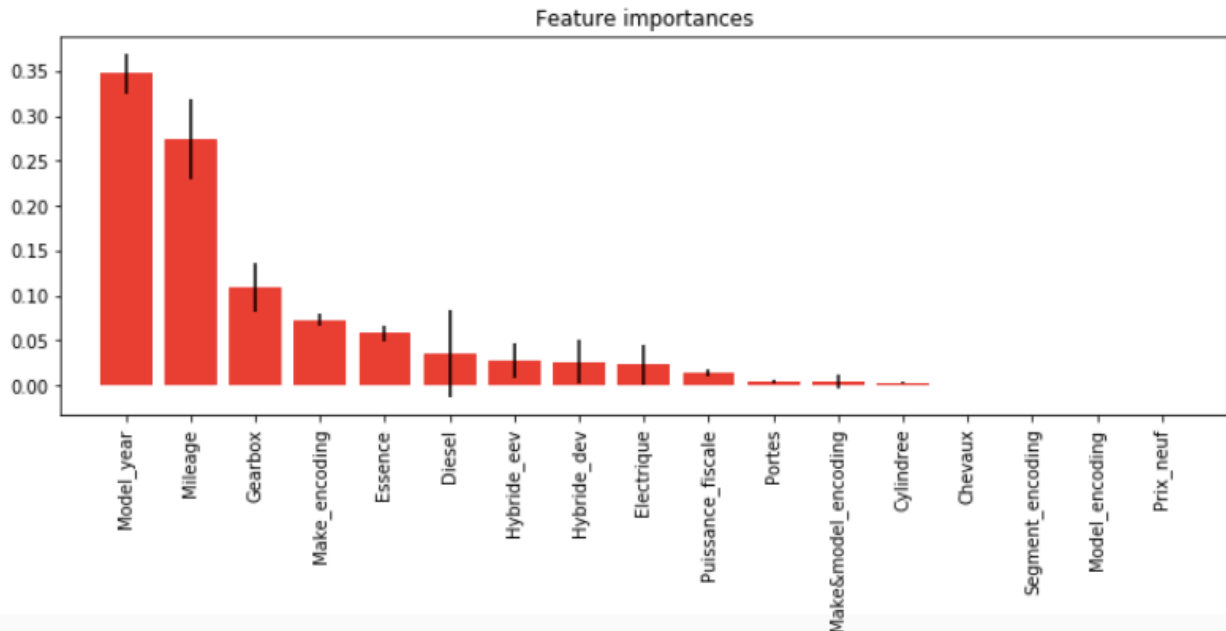
La MAPE seule n'est donc pas une mesure suffisante de la performance du modèle sur un échantillon de test. Mais, pour l'évaluation des traitements de données on se concentre sur ce dernier résultat, en prenant en compte MAPE et MAE. Le score de référence est donc :
MAPE : 9.54% et MAE : 1721.08.

B. Importance des features

Au fil de l'étude des données, on calcul la corrélation entre les nouvelles variables et la variable de prix de vente à prédire. Mais ces valeurs ne donnent pas beaucoup d'informations sur comment apprend le modèle.

	Price	Model_year	Mileage	Gearbox	Make_encoding
Price	1.000000	0.201825	-0.301137	-0.408782	0.240674
Model_year	0.201825	1.000000	-0.704815	-0.067363	-0.127472
Mileage	-0.301137	-0.704815	1.000000	0.050867	0.015952
Gearbox	-0.408782	-0.067363	0.050867	1.000000	-0.120080
Make_encoding	0.240674	-0.127472	0.015952	-0.120080	1.000000
Model_encoding	0.033411	-0.048143	0.019660	-0.053635	0.032481
Essence	-0.004578	0.040444	-0.256478	0.095469	0.080623
Diesel	-0.012297	-0.050195	0.263545	-0.029602	-0.111611

En affichant les scores Gini calculés par la forêt aléatoire, on observe une diminution de l'importance des variables, suivant l'ordre de traitement du modèle. **Cela explique le très bon score du modèle de référence, ainsi que la difficulté d'affiner le résultat.** Le modèle apprend très vite dès les premières études de features. Les prochaines variables sont ensuite importantes pour l'optimisation du modèle, mais celui ci ne les considère plus comme importante.



C. Résultats des traitements de données

Notre score de référence est donc **MAPE : 9.54% et MAE : 1721.08**, en utilisant les features marque, modèle, marque&modèle, date, motorisation, boîte de vitesse, portes et puissance fiscale.

Le premier affinage est l'ajout des données de la **cylindrée** et du **nombre de chevaux**. Ce sont deux variables qui indique la puissance du moteur d'un véhicule, et sont donc fortement corrélées au prix de vente. Sur le même échantillon que précédemment, on obtient une bonne amélioration : **MAPE : 9.19%, MAE : 1683.82**. Les prochains modèles utilisent ces deux variables.

Les traitements du champ *Option* ont apportés plus de bruit que d'informations utiles au modèle.

One-hot encoding : **MAPE : 9.80%, MAE : 1801.03**

Count vectorizer : **MAPE : 9.87%, MAE : 1821.04**

TF-IDF : **MAPE : 10.46%, MAE : 1920.52**

L'ajout du segment des véhicules n'a pas beaucoup influencé le second modèle : **MAPE : 9.22%, MAE : 1686.79**

L'ajout des prix du neufs, bien que peu rigoureux apporte la seconde meilleure amélioration : **MAPE : 8.97%, MAE : 1669.45.**

En utilisant le meilleur nombre d'arbres indiqué par la GridSearch (150 arbres), on note une infime amélioration : **MAPE : 8.96%, MAE : 1666.78.**

Finalement, la mesure **MAPE globale a progressé de 58% à 34%**, avec le meilleur échantillon d'entraînement à **8.85%**. La **MAE est passée de 1704 à 1656.**

```
CV1 (133017, 17)(33255, 17)
MAPE : 8.86%, MAE : 1660.49
```

```
CV2 (133017, 17)(33255, 17)
MAPE : 8.85%, MAE : 1645.68
```

```
CV3 (133018, 17)(33254, 17)
MAPE : 84.65%, MAE : 1703.74
```

```
CV4 (133018, 17)(33254, 17)
MAPE : 22.65%, MAE : 1612.30
```

```
CV5 (133018, 17)(33254, 17)
MAPE : 46.65%, MAE : 1661.48
```

```
-----
MAPE : 34.34%, MAE : 1656.74
```

Le modèle généralise bien mieux ces prédictions. Comparer ces deux mesures permet d'avoir une idée correcte de la performance du modèle. **Une MAE de 1656 sur des prix de véhicules pouvant aller de 5,000 à plus de 300,000€ est un bon score, notamment avec une MAPE à 34%.**