Dauphine | PSL
UNIVERSITÉ PARIS

Pratique de la Data Science

Régression régularisée

Le but de ce projet est d'implémenter des méthodes de régression (régularisées ou non) sur le jeu de données « `credit_immobilier_ISF.csv` ». Il s'agit d'un dataset d'informations concernant des clients voulant effectuer un emprunt. L'objectif est de prédire le montant emprunté à un client en fonction de son profil. Ceci inclue donc des informations sur sa situation personnelle et socio-professionnelle, ainsi que sur son historique en tant qu'emprunteur.

Afin de mettre au point ce projet, nous avons suivi les différentes étapes de poursuite d'un projet Data Science, à savoir une analyse préliminaire des données (**E**xploratory **D**ata **A**nalysis), suivi d'une phase de modélisation où nous avons testé différents types de modèles afin de les comparer et de choisir le modèle optimal.

Nous avons présenté nos résultats sous forme de notebook reprenant ces différentes étapes accompagnées de paragraphes et de commentaires décrivant plus en détail les différentes lignes de code. Ce compte rendu vise surtout à synthétiser nos travaux et à répondre aux différentes problématiques du projet. Pour plus de détails, nous vous prions donc de consulter le notebook.

1 – Analyse des données

Le jeu contient au total 1021 observations et 17 variables. Les données présentes sont des données numériques et catégorielles. En examinant le type des variables, les données sont déjà correctement encodées (pas de variable quantitative de type autre de numérique). Le dataset contient les variables suivantes :

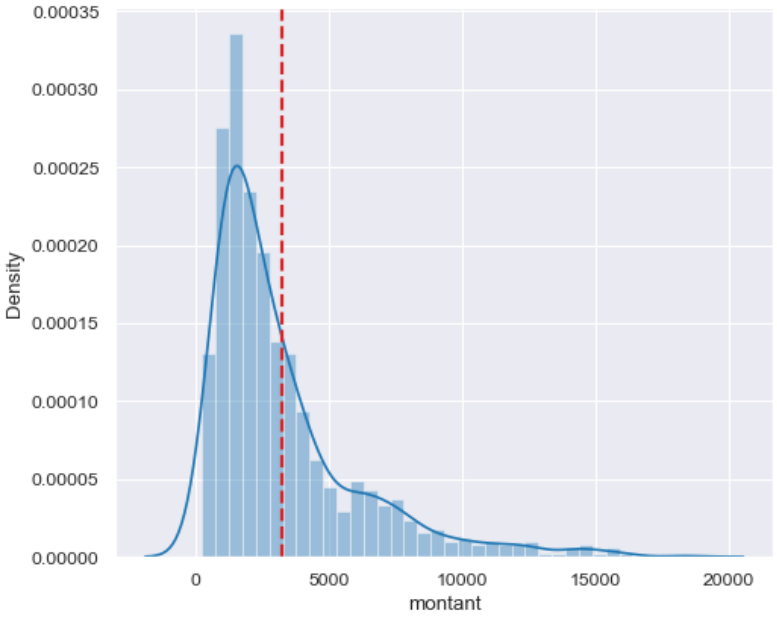
- La variable `compte_courant` (resp. `compte_epargne`) décrit la valeur dont le client dispose sur son compte courant (resp. compte épargne), mesurée en intervalles.
- La variable `historique_credit` permet de connaître la fiabilité du client par rapport à ses crédits précédents, et ainsi sa capacité à honorer ses engagements.
- La variable `raison` mentionne le motif du client dans sa démarche de demande de crédit.
- Des variables booléennes comme `dependants`, `telephone` et `defaillant`.
- D'autres variables dont le sens peut être déduit directement à partir de leurs noms.

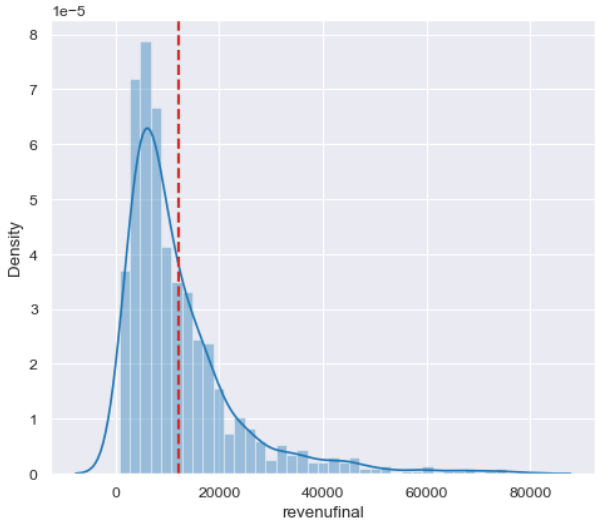
Quant à la variable target `montant`, celle-ci est mesurée en euros.

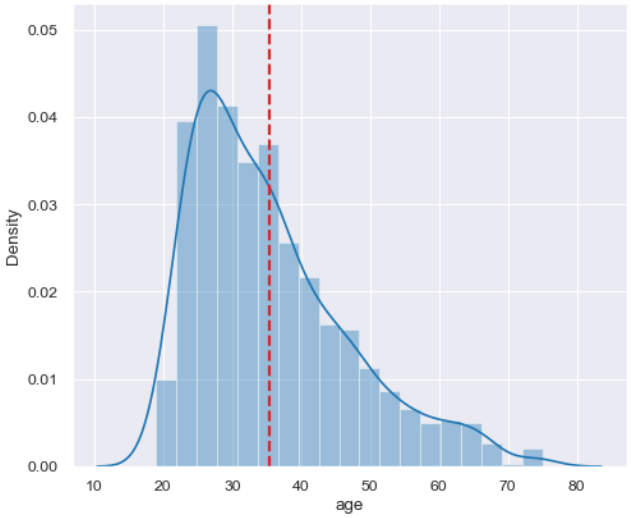
On commence par faire des **statistiques descriptives simples** afin d'avoir un premier aperçu de la distribution des variables numériques et éventuellement détecter des outliers potentiels. En regardant les résultats (minimum et maximum), il n'y a à priori aucune valeur aberrante dans le dataset. Concernant les variables catégorielles, on remarque que la variable telephone ne prend qu'une seule modalité. Elle n'est donc d'aucune utilité et on décide de la sortir du dataset.

Après une première analyse descriptive, nous passons à l'étape de **nettoyage des données**. Nous détectons un nombre très faible de données manquantes, notamment au niveau d'une observation (n° 29) qui se distingue par des erreurs flagrantes en termes de cohérence. Nous décidons alors de la supprimer du dataset. Au vu du nombre très faible de valeurs manquantes, ces dernières sont simplement remplacées par la médiane dans le cas d'une variable numérique, et par le mode dans le cas d'une variable catégorielle.

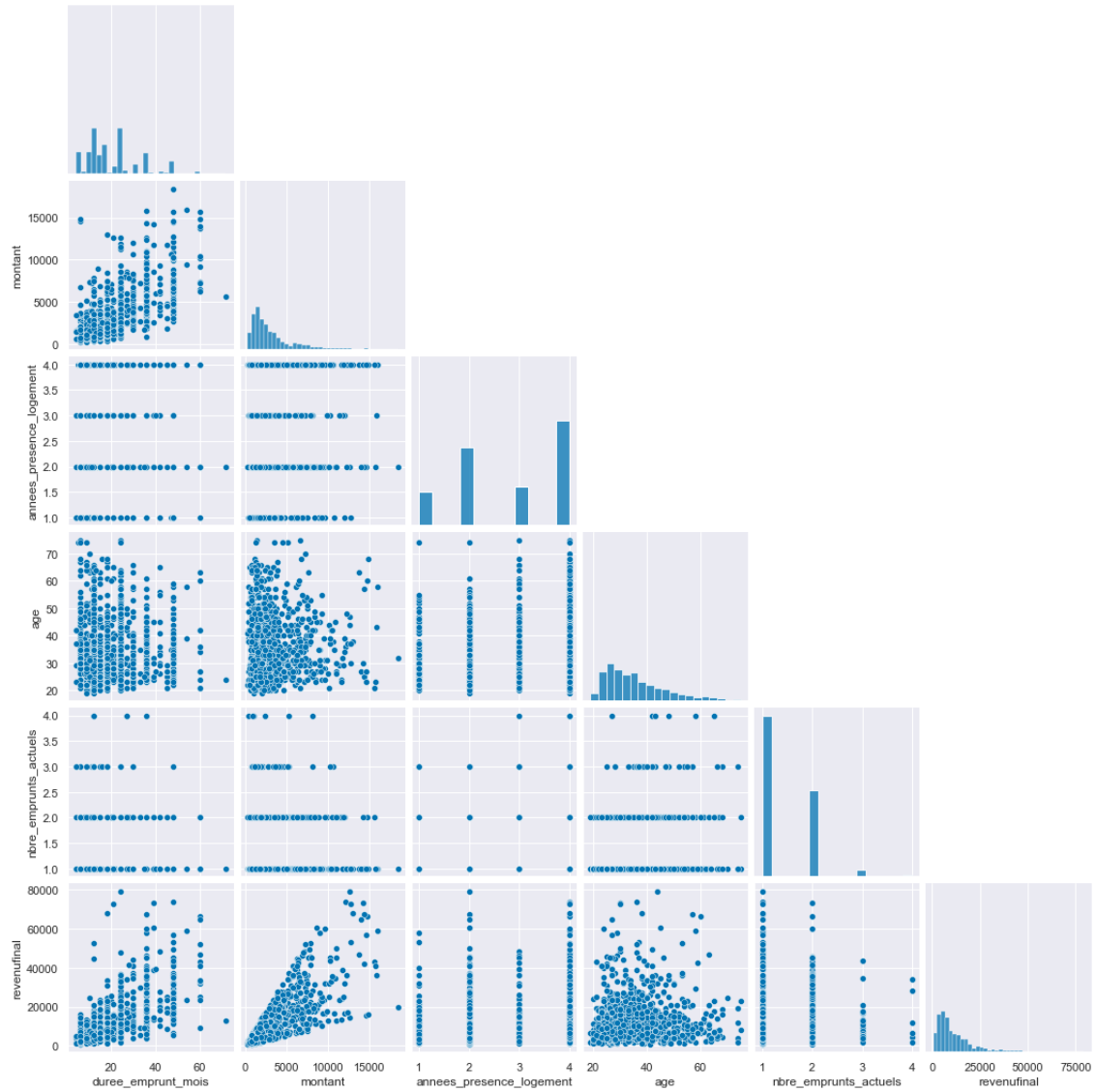
Lors de **l'analyse univariée**, nous avons décidé de partir par la suite sur une transformation des variables numériques `montant`, `age` et `revenufinal` en raison de leur asymétrie apparente au niveau des histogrammes correspondants. Il s'agira ainsi de modifier les variables afin d'atténuer l'effet de *skewness* sur leurs distributions respectives.

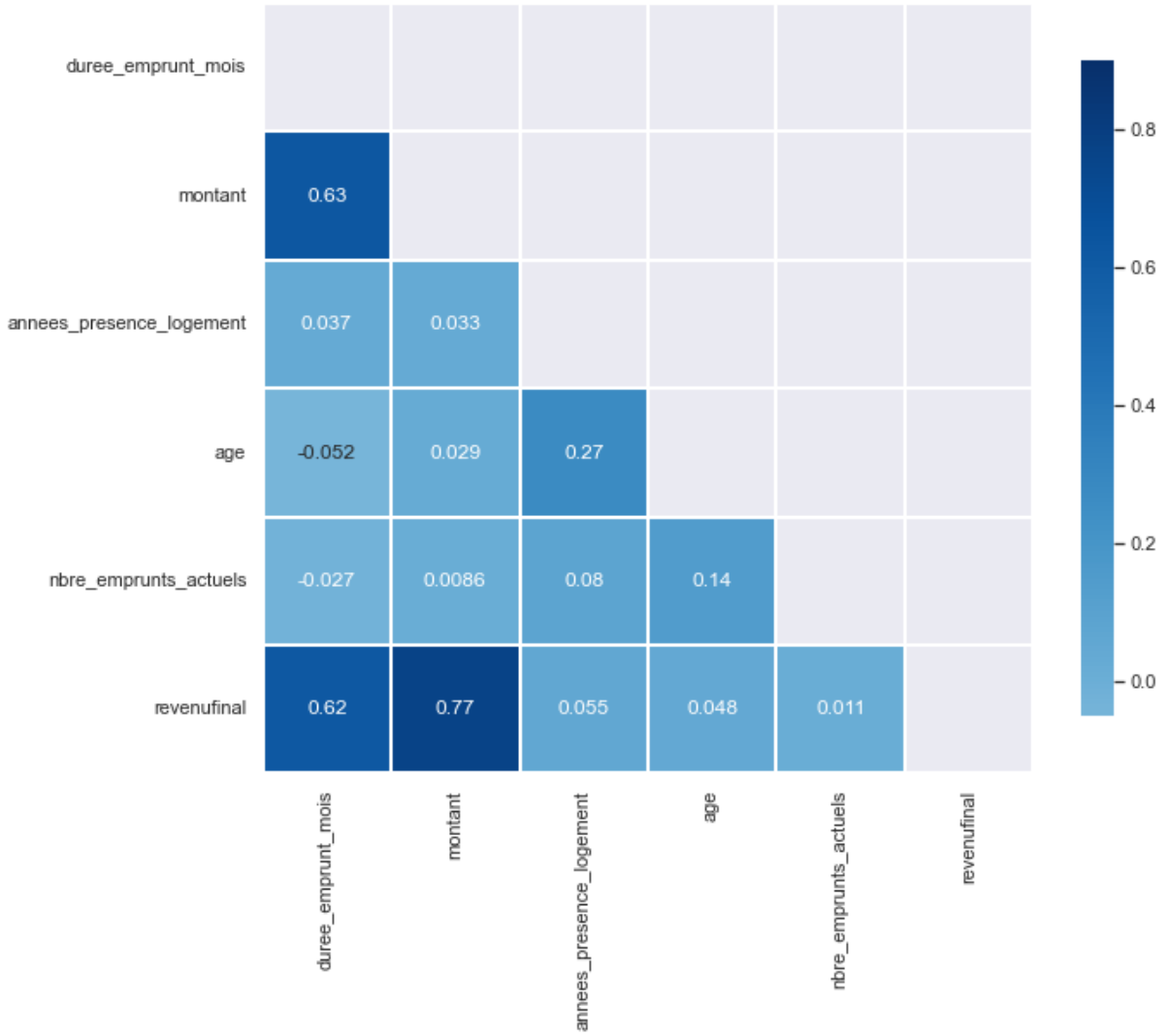






Lors de **l'analyse multivariée**, nous remarquons une forte corrélation entre la variable target `montant` et les variables `revenufinal` et `duree_emprunt_mois`. Ceci nous laisse penser que ces variables explicatives joueront un rôle significatif dans la qualité de prédiction des modèles.





Lors de la phase de **feature engineering**, nous appliquons la transformation `boxcox` et nous remarquons que les variables transformées sont bel et bien symétriques par rapport à avant. Les transformations en question passent notamment par la méthode `boxcox` qui appliquent aux données une fonction de transformation spécifique.

Il ne nous reste plus qu'à passer à l'étape **d'encodage des variables**. Dans notre cas, il s'agit d'effectuer un *ordinal encoding* sur la variable `historique_credit`, ce qui est logique compte tenu de la nature hiérarchique de cette variable. Quant aux autres variables catégorielles, nous décidons tout simplement d'effectuer un *one-hot encoding*, qui associe à chaque modalité de chaque variable une nouvelle variable booléenne représentant l'appartenance ou pas à cette modalité.

2 – Modélisation

Avant de modéliser, nous séparons notre jeu de données en données `train` et `set`, avec une proportion de données test de 30%. Nous commençons par effectuer une **simple régression linéaire**. Nous retrouvons les résultats suivants :

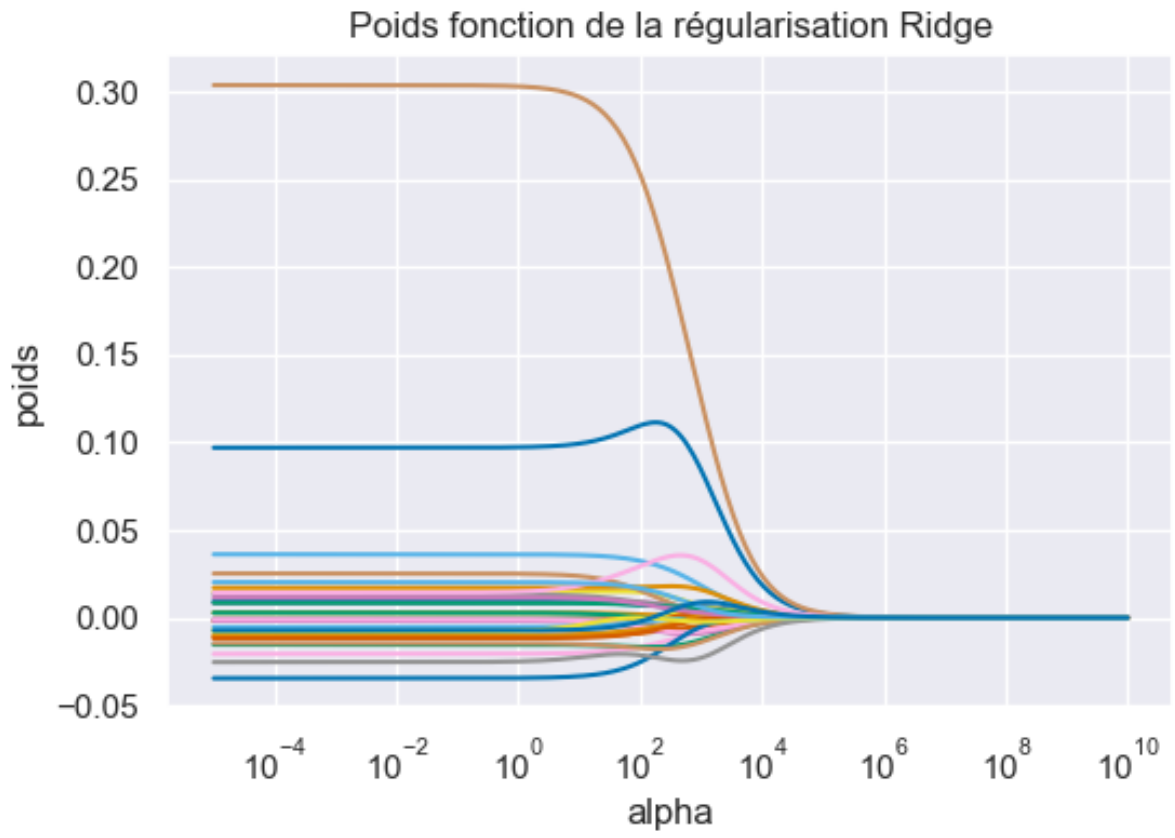
MSE	**3957619.400243014**
RMSE	**1989.3766360955922**
R2	**0.6331248827208673**
R2 AJUSTE	**0.5916171139776079**

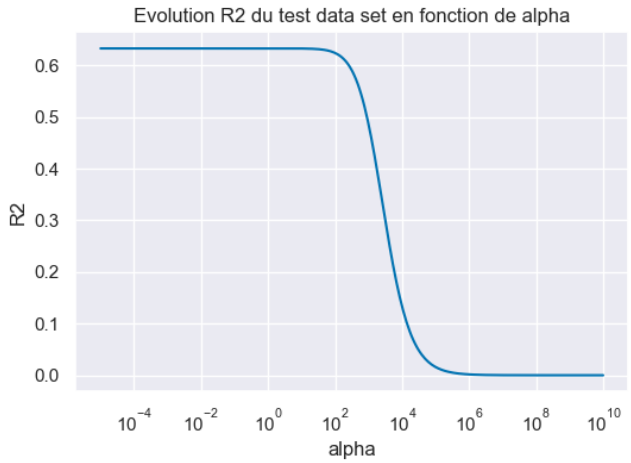
On obtient donc un R^2 d'environ 63,31%.

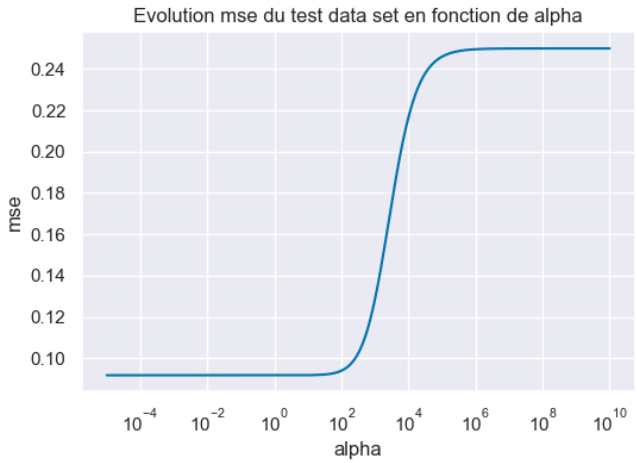
On procède ensuite à la phase de régularisation des données. L'objectif de cette partie est d'effectuer une régression Ridge, Lasso et Elastic Net puis de sélectionner le meilleur modèle. Pour la **régression Ridge**, on cherche d'abord le coefficient **α** optimal dans la régularisation. Nous trouvons une valeur optimale de **0.05788177037701274.** Voici les résultats de la régression Ridge :

MSE	**4051282.4267858523**
RMSE	**2012.779776027634**
R2	**0.6308737331050696**
R2 AJUSTE	**0.5891112722519936**

On obtient un R^2 d'environ 63,09%, ce qui est sensiblement similaire à la régression classique.



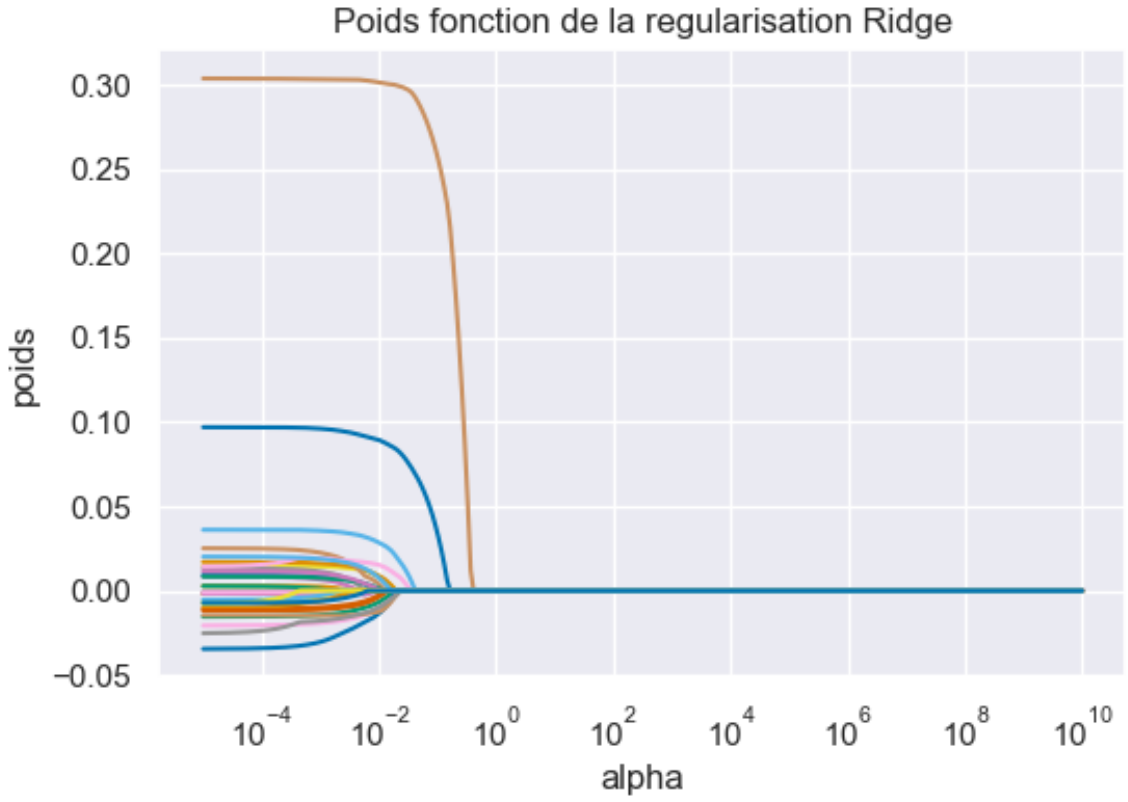


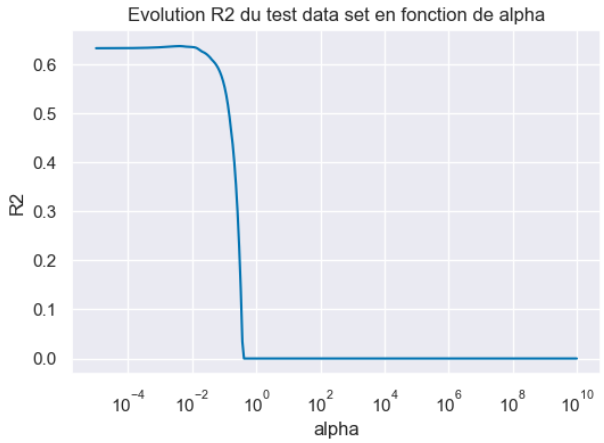


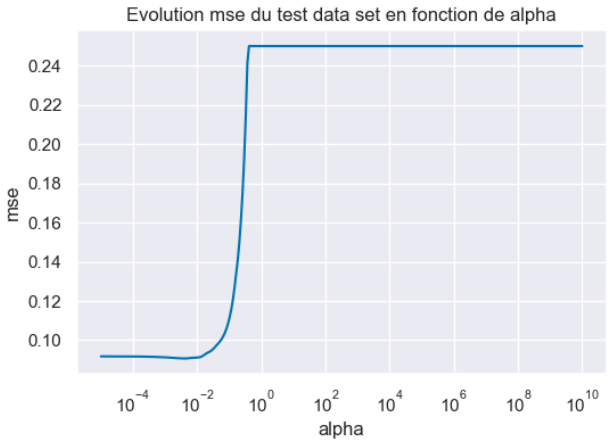
Passons à **la régression Lasso**. Comme pour la régression Ridge, on cherche à calculer le coefficient de régularisation optimal. Ici, la valeur optimale est **0.0016119733573888157.**
Voici les résultats de la régression Lasso :

MSE	**4442598.819878846**
RMSE	**2107.7473330261496**
R2	**0.6082713890958794**
R2 AJUSTE	**0.5639517287381139**

On obtient un R^2 d'environ 60,83%. La régression Lasso permet entre autres d'effectuer une sélection de variables. Les variables retenues sont `revenufinal` et `duree_emprunt_mois`, ce qui est en accord avec les observations précédentes des fortes corrélations entre ces variables et la variable target `montant`.







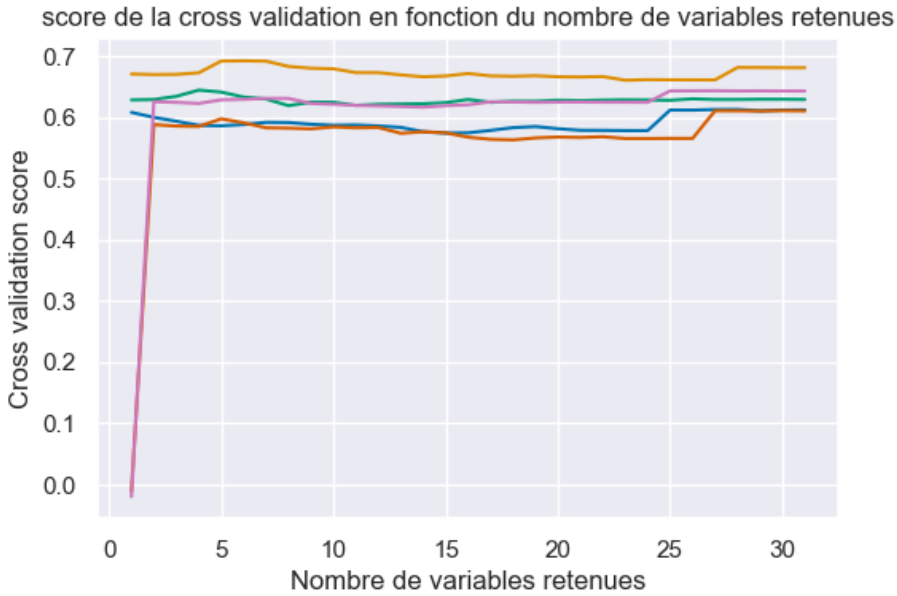
Place maintenant à **la régression Elastic Net**. Avec les paramètres par défaut, on a les résultats suivants :

MSE	**6004249.087173229**
RMSE	**2450.3569305660817**
R2	**0.41139285928239877**
R2 AJUSTE	**0.3447986207340571**

Puis en choisissant comme coefficients de régularisation les coefficients utilisées pour les méthodes Ridge et Lasso, nous obtenons un R^2 d'environ 62,23% sur les données train mais avec un score très mauvais sur les données test, ce qui pourrait être le signe d'un overfitting important.

En conclusion, il semblerait que le meilleur modèle est la régression linéaire classique. Il serait opportun d'utiliser d'autres métriques afin de comparer les différents modèles, ou de partir sur d'autres types de modèles. En effet, certaines hypothèses à valider avant de pouvoir effectuer une régression linéaire peuvent ne pas être vérifiées (voir plus bas).

Enfin, on cherche à connaître **l'importance des features** en effectuant une sélection automatique par récursivité grâce au module `RFECV`.



On trouve un nombre optimal de variables sélectionnées de **28**, ainsi qu'un classement de l'importance des différentes features (voir notebook).

A partir des features sélectionnées plus haut, on cherche à vérifier les **quatre hypothèses principales** de la régression linéaire :

a) Les erreurs sont-elles centrées ?

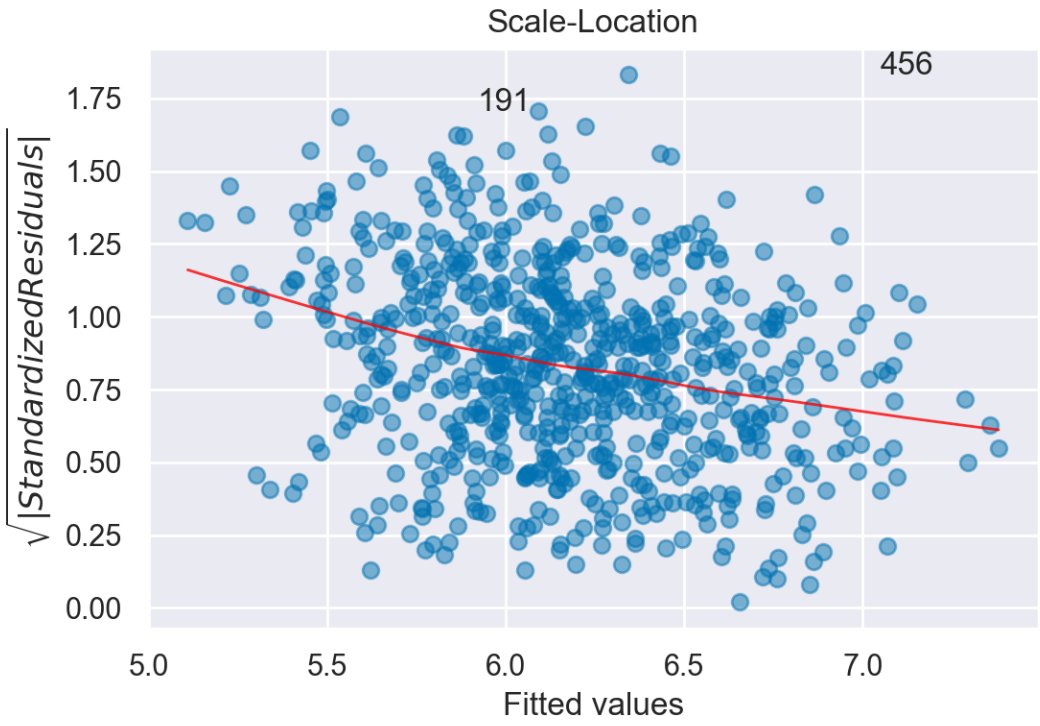
b) Les erreurs sont-elles de variance constante (homoscédasticité) ?

c) Les erreurs sont-elles supposées indépendantes ?

d) Les erreurs sont-elles supposées gaussiennes ?

On vérifie ces quatre hypothèses à partir des graphiques/captures suivantes :

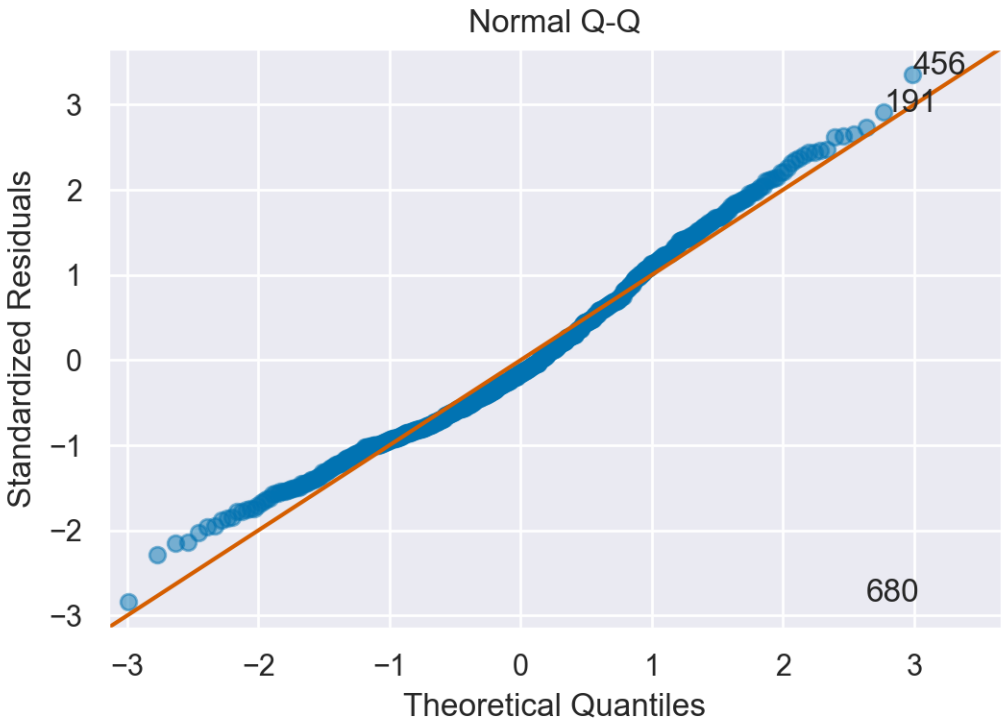
Les points sont répartis de part et d'autre de zéro sans tendance particulière, les erreurs sont donc centrées et l'hypothèse a) est vérifiée.



La courbe rouge est décroissante, **l'hypothèse b) n'est donc pas vérifiée** !

Omnibus:	25.189	Durbin-Watson:	2.026
Prob(Omnibus):	0.000	Jarque-Bera (JB):	27.216
Skew:	0.472	Prob(JB):	1.23e-06
Kurtosis:	2.840	Cond. No.	396.

À l'aide du test de Durbin-Watson dont la valeur est égale à **2.026**, on déduit qu'il n'y a pas de corrélation entre les résidus. L'hypothèse c) est donc vérifiée.



Les points sont alignés tout au long de la première bissectrice, l'hypothèse d) est vérifiée.

Conclusion

En conclusion, nous choisissons de garder le modèle de régression linéaire classique avec un R^2 de 63%. Après avoir tenté les différentes méthodes de régularisation, nous pouvons en déduire que certaines features sont plus importantes que d'autres en termes d'impact sur la qualité de prédiction de la variable target.

Il est aussi important de noter que l'hypothèse d'homoscédasticité du modèle n'est pas vérifiée, ce qui pourrait nous guider vers le choix d'autres modèles qui ne sont pas forcément linéaires.