# PROJET DATA MINING APPRENTISSAGE

NABIL MARHAR/AYMAN YAYA/HARON REZGUI

2024-04-15

## 1° INTRODUCTION

Avec l'accélération continue des innovations technologiques, les fabricants de téléphones mobiles font face à un défi majeur : déterminer correctement le prix de leurs produits pour maximiser leur compétitivité sur le marché. Un prix précisément ajusté peut influencer significativement la perception de la marque et les décisions d'achat des consommateurs. Dans ce contexte, il devient crucial de développer des modèles prédictifs capables d'estimer les prix basés sur les caractéristiques techniques des produits.

Description des Données

Les données utilisées dans ce projet proviennent d'un ensemble de données publiquement accessible sur Kaggle, qui comprend des informations détaillées sur plusieurs modèles de téléphones mobiles. Ces données incluent, mais ne sont pas limitées à, la mémoire interne, la taille et la résolution de l'écran, la capacité de la batterie, la puissance du processeur, le poids, et les prix actuels des téléphones. Ces variables sont essentielles pour comprendre les facteurs qui influencent le plus les décisions de tarification dans l'industrie des smartphones.

Ces données incluent des caractéristiques telles que :

- **Product_id :** Identifiant unique pour chaque téléphone.
- **Price :** Le prix du téléphone, variable cible pour la prédiction.
- **Sale :** Indicateur des ventes ou remise en pourcentage.
- **Weight :** Poids du téléphone en grammes.
- **Resolution :** Taille de l'écran en pouces.
- **PPI :** Pixels par pouce, densité de pixels de l'écran.

- **CPU Core :** Nombre de cœurs du processeur.
- **CPU Frequency :** Fréquence du processeur en GHz.
- **Internal Memory :** Mémoire interne en Go.
- **RAM :** Mémoire RAM en Go.
- **RearCam :** Résolution de la caméra arrière en mégapixels.
- **Front_Cam :** Résolution de la caméra frontale en mégapixels.
- **Battery :** Capacité de la batterie en mAh.
- **Thickness :** Épaisseur du téléphone en millimètres.

## Méthodologie

### Analyse Exploratoire des Données (AED)

La première étape consistera à réaliser une analyse exploratoire des données pour évaluer la qualité des données, identifier les valeurs manquantes ou aberrantes, et comprendre les distributions statistiques des différentes caractéristiques. Cette phase comprendra également l'analyse des corrélations entre les caractéristiques techniques des téléphones et leurs prix pour identifier les variables potentiellement prédictives.

### Construction de Modèles de Prédiction

Plusieurs modèles de régression seront construits et évalués pour leur capacité à prédire les prix des téléphones mobiles :

- **Régression Linéaire :** Pour établir une baseline de performance avec une approche simple.
- **Régression Polynomiale :** Pour examiner les relations non-linéaires entre les caractéristiques et les prix.
- **Régressions Ridge et Lasso :** Pour pénaliser les modèles de régression afin de réduire le surajustement et améliorer la généralisation.
- **Arbres de Décision et Forêts Aléatoires :** Pour capturer des interactions complexes entre les caractéristiques sans la nécessité de transformations manuelles des données.

### Évaluation des Modèles

Les modèles seront évalués en utilisant la racine de l'erreur quadratique moyenne (RMSE) et le coefficient de détermination ($R^2$). Ces métriques aideront à quantifier la précision des prédictions des modèles et à comparer leur performance relative.

Analyse d'Importance des Variables

L'importance des variables sera analysée pour tous les modèles afin d'identifier les caractéristiques qui influencent le plus le prix des téléphones. Cette analyse aidera à comprendre les facteurs clés qui déterminent les prix dans l'industrie des smartphones, fournissant des insights précieux pour les décisions stratégiques de tarification.

Cette étude vise à fournir un modèle robuste pour la prédiction des prix des téléphones mobiles en utilisant des techniques statistiques et d'apprentissage machine, afin de soutenir les fabricants dans leurs stratégies de tarification. Les résultats attendus devraient offrir une compréhension approfondie des dynamiques de prix dans le secteur des télécommunications mobiles.

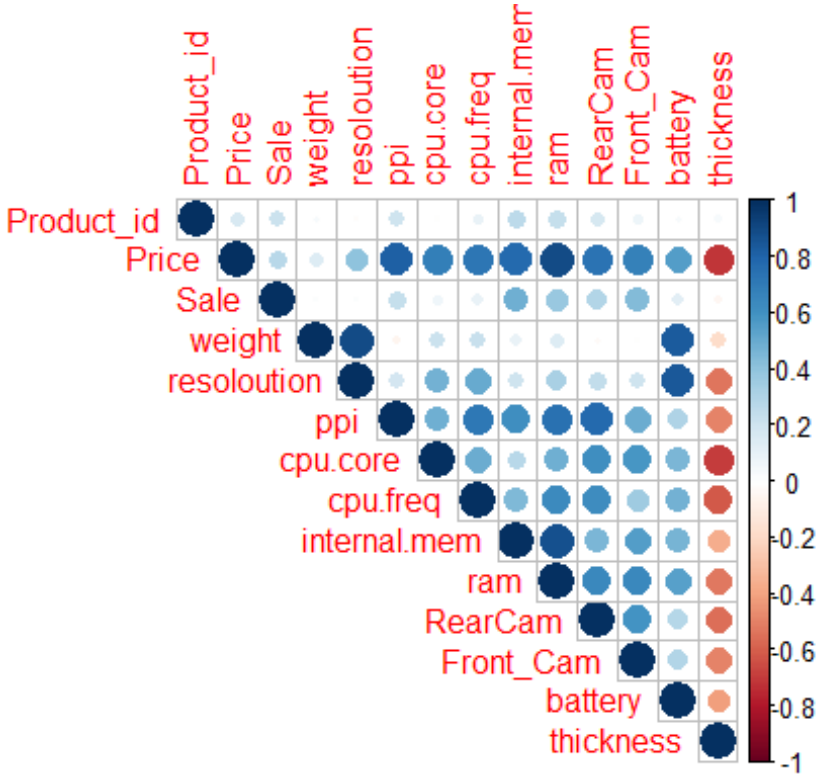
La base de données analysée contient 161 observations de 14 variables, englobant des caractéristiques telles que l'identifiant du produit, le prix, les ventes, le poids, la résolution et la taille de l'écran, la densité de pixels (ppi), les spécifications du processeur (nombre de cœurs et fréquence), la mémoire interne, la RAM, les résolutions des caméras arrière et frontale, la capacité de la batterie et l'épaisseur du téléphone, illustrant une diversité riche en informations pour modéliser les facteurs influençant les prix des téléphones mobiles.

Les premières lignes de la base de données révèlent une variété de téléphones mobiles, allant du modèle avec un identifiant de produit 203 doté d'un écran de 5.2 pouces, une mémoire interne de 16 Go et une batterie de 2610 mAh, à un autre modèle, identifiant 947, équipé d'un écran plus large de 5.5 pouces, 16 Go de mémoire interne, et une caméra arrière puissante de 16 mégapixels, illustrant ainsi la diversité et la complexité des spécifications techniques contenues dans cette base de données enrichie pour l'analyse des prix.

Le résumé statistique de la base de données des téléphones mobiles révèle une grande variété de spécifications, depuis des produits à prix modique avec des composants de base jusqu'à des appareils haut de gamme dotée de fonctionnalités avancées, reflétant une gamme étendue de performances, de capacités et d'innovations destinées à répondre aux besoins et préférences diversifiés des consommateurs sur le marché mondial.

## 2° Calibrage des modèles

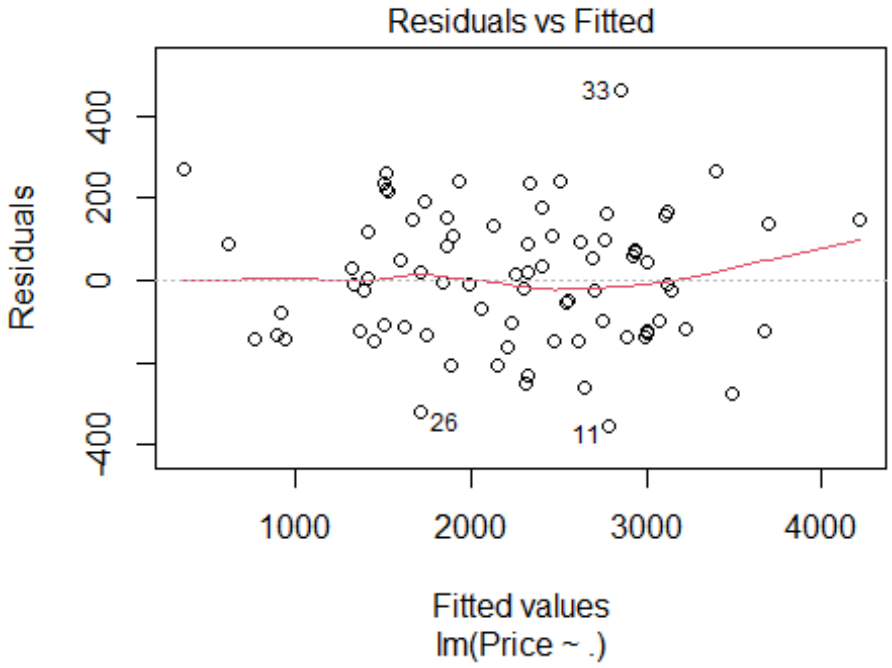
Corrélation




Ce graphique de corrélations présente un paysage de relations entre les caractéristiques des téléphones mobiles. Chaque case, où le croisement de deux attributs se rencontre, est colorée d'un bleu profond à un rouge intense, ou reste pâle, illustrant la force et la direction de leurs liaisons. Les couleurs chaudes, comme le rouge, signifient que lorsque l'une des caractéristiques augmente, l'autre diminue, et inversement, tandis que les couleurs froides, comme le bleu, suggèrent une augmentation conjointe. Les cercles les plus grands et les plus saturés parlent d'une forte corrélation, tandis que les cercles plus petits et plus transparents révèlent des liens plus faibles. Ce tableau fournit une cartographie visuelle permettant de déchiffrer le réseau interconnecté des attributs qui pourraient influencer le prix d'un téléphone.

Sur le graphique, les différentes variables comme la capacité de la batterie (« battery »), la mémoire interne (« internal.mem »), ou la densité de pixels (« ppi »), interagissent entre elles et avec le prix (« Price ») de manière unique. Par exemple, supposons que la taille de la batterie (« battery ») montre un large cercle bleu lorsqu'elle est corrélée avec le prix, cela pourrait indiquer que les téléphones avec des batteries plus grandes tendent à être plus

chers. Inversement, si le poids (« weight ») affiche un grand cercle rouge en relation avec la taille de l'écran (« resoloution »), cela suggérerait que les téléphones avec des écrans plus grands pourraient paradoxallement être plus légers, peut-être à cause de matériaux modernes et légers utilisés dans leur construction. Ces aperçus visuels aident à identifier les variables qui méritent une attention particulière dans l'élaboration de modèles de prédiction de prix et dans l'analyse des tendances du marché.




<u>Modèle AIC</u>

L'analyse de régression linéaire effectuée sur l'échantillon d'apprentissage pour prédire le prix des téléphones mobiles a produit des résultats détaillés. Le modèle initial comprenait toutes les variables disponibles, mais après une procédure de sélection par AIC (Critère d'Information d'Akaike), certaines variables ont été éliminées pour optimiser le modèle.

Les résidus, qui mesurent les écarts entre les valeurs observées et les valeurs prédites par le modèle, s'étendent de -352 à 461, ce qui montre une certaine variabilité dans la précision des prédictions du modèle. Cependant, le $R^2$ ajusté est de 0.9547, ce qui indique que le modèle explique une grande partie de la variance des prix.

Les coefficients significatifs incluent le nombre de cœurs du processeur (« cpu.core »), la densité de pixels (« ppi »), la mémoire interne (« internal.mem »), la RAM (« ram »), la capacité de la batterie (« battery »), et l'épaisseur (« thickness »). Par exemple, chaque augmentation d'un Go de RAM est associée à une augmentation moyenne de 85,23 du prix du téléphone, tout en tenant compte des autres variables. Inversement, chaque millimètre supplémentaire en épaisseur est associé à une baisse de 84,74 dans le prix, indiquant que les téléphones plus fins pourraient être plus chers.

Le modèle final après la sélection par AIC a légèrement augmenté son AIC à 1185.548, suggérant une légère diminution de l'efficacité par rapport au modèle précédent en termes d'équilibre entre la complexité du modèle et son ajustement. Ce modèle final comprend les variables « weight », « ppi », « cpu.core », « cpu.freq », « internal.mem », « ram », « battery », et « thickness ».

En résumé, cette procédure de modélisation a permis de déterminer les caractéristiques techniques les plus pertinentes pour prédire le prix des téléphones mobiles dans cet échantillon de données, ce qui pourrait éventuellement guider les fabricants sur les caractéristiques à prioriser dans la conception de nouveaux modèles.

Modèle $R^2$

L'analyse de sélection de modèle basée sur le $R^2$ ajusté a identifié le modèle de régression linéaire le plus performant pour prédire les prix des téléphones mobiles à partir de l'échantillon d'apprentissage. Le $R^2$ ajusté est un critère de sélection qui ajuste le $R^2$ pour le nombre de prédicteurs dans le modèle, favorisant un équilibre entre la complexité du modèle et la capacité de prédiction. Dans cette analyse, le modèle qui maximise le $R^2$ ajusté inclut les variables suivantes :

- Sale : représente soit les ventes soit les promotions et a un effet négatif sur le prix, indiquant que les augmentations dans cette variable sont associées à une diminution du prix.

- Resolution : la résolution de l'écran, qui ne s'est pas révélée être un prédicteur significatif du prix dans ce modèle.

- PPI : la densité de pixels de l'écran est un prédicteur positif significatif, indiquant que les téléphones avec un nombre plus élevé de pixels par pouce sont plus chers.

- CPU Core : le nombre de cœurs du processeur, qui a un effet positif significatif sur le prix, suggérant que les téléphones avec plus de cœurs tendent à être plus chers.

- CPU Frequency : la fréquence du processeur a également un effet positif significatif, indiquant que les processeurs plus rapides augmentent le prix du téléphone.

- Internal Memory : la mémoire interne est un prédicteur positif significatif, ce qui implique que plus la capacité de stockage est grande, plus le prix est élevé.

- Rear Camera : la résolution de la caméra arrière n'a pas montré de signification statistique pour influencer le prix dans ce modèle particulier.

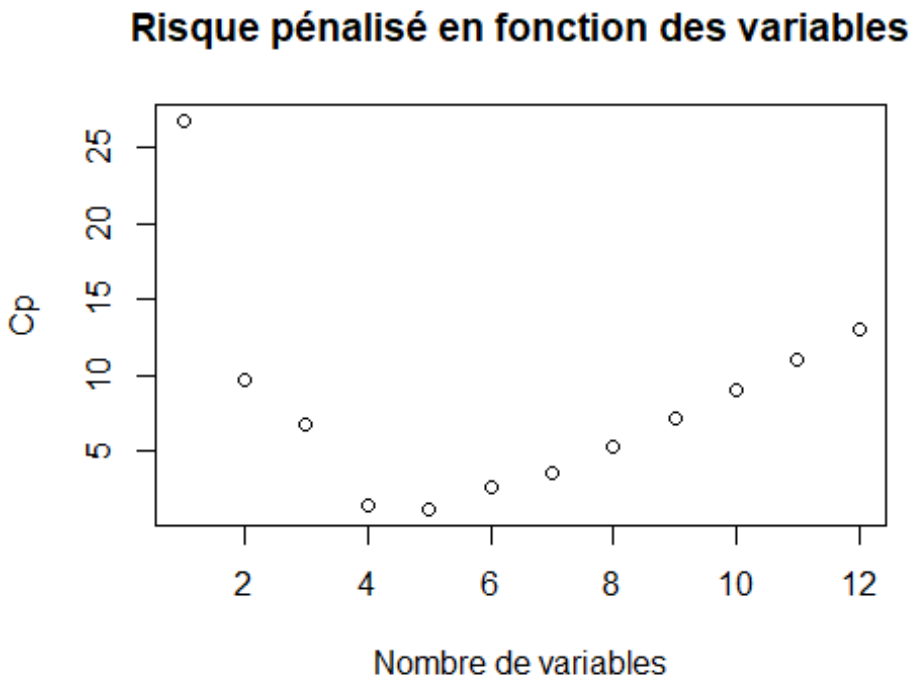
- Front Camera : la résolution de la caméra frontale est un prédicteur positif, indiquant que les téléphones avec de meilleures caméras frontales tendent à être plus chers.

- Battery : la capacité de la batterie a montré une tendance à être significative, suggérant que les téléphones avec des batteries plus grandes pourraient coûter plus cher, bien que cette relation ne soit pas statistiquement significative au niveau conventionnel de 0.05.

Le modèle a un $R^2$ ajusté de 0.9316, signifiant qu'il explique une grande proportion de la variabilité des prix des téléphones, un très bon ajustement pour un modèle de données réelles. L'erreur standard résiduelle de 204.5 indique la variabilité des résidus autour des valeurs prédites du modèle. Finalement, la valeur très faible du p-value de la statistique F confirme que le modèle est significatif globalement.

Modèle Cp Mallows

Le critère d'information de Mallows (Cp) est une mesure d'évaluation de la qualité d'un modèle statistique. Il est utilisé pour estimer la qualité de l'ajustement d'un modèle par rapport à un modèle complet, en prenant en compte à la fois l'ajustement du modèle et sa complexité. Un Cp plus petit indique généralement un meilleur ajustement du modèle aux données.




La recherche du meilleur modèle de régression linéaire pour prédire le prix des téléphones mobiles, en utilisant le critère de Mallows (Cp), a identifié un modèle optimal parmi les combinaisons possibles de variables explicatives. Le critère Cp de Mallows vise à équilibrer la précision du modèle et sa complexité, favorisant un modèle qui a moins de variables mais qui prévoit tout de même bien la variable dépendante.

La visualisation du risque pénalisé (Cp) en fonction du nombre de variables montre que le modèle avec le Cp le plus bas, donc potentiellement le meilleur modèle au sens de ce critère, in

clut les variables Sale, Resolution, PPI, CPU Core , CPU Frequency , Internal Memory , Front Camera , et Battery .

Le modèle sélectionné présente les caractéristiques suivantes :

- Coefficients significatifs : Sale, PPI , CPU Core , CPU Frequency , Internal Memory , et Front Camera ont un impact statistiquement significatif sur le prix, avec le nombre de cœurs du processeur et la mémoire interne étant parmi les prédicteurs les plus influents.

- $R^2$ ajusté : Avec une valeur de 0.931, ce modèle explique une grande part de la variabilité des prix des téléphones dans l'échantillon d'apprentissage.

- Erreur standard résiduelle : Environ 205.5, indiquant la dispersion des résidus (les écarts entre les valeurs observées et prédites par le modèle).

- Statistiques F : La valeur très basse du p-value (< 2.2e-16) associée à la statistique F suggère que le modèle est globalement significatif.

Ce processus met en évidence l'importance de certaines spécifications techniques dans la détermination du prix des téléphones et suggère que les fabricants devraient peut-être se concentrer sur ces caractéristiques lors de la conception et de la tarification de leurs produits.

Comparaison des erreurs empiriques

Les erreurs empiriques indiquées sont le résultat du calcul de l'erreur quadratique moyenne (Mean Squared Error, MSE) pour différents modèles de régression sur l'échantillon de test. Le MSE mesure la qualité d'un modèle en calculant la moyenne des carrés des écarts entre les valeurs prédites et les valeurs réelles.

Voici les erreurs calculées pour chaque modèle :

- Modèle complet (Reg) : L'erreur MSE est de 35359.91. Ce modèle, qui inclut toutes les variables disponibles, présente l'erreur la plus faible parmi tous les modèles testés sur l'échantillon de test. Cela suggère qu'il pourrait fournir la prédiction la plus précise pour les données non vues.

- Modèle AIC (modelAIC) : Avec une erreur MSE de 37238.2, ce modèle, optimisé pour minimiser le critère d'information d'Akaike, affiche une erreur légèrement plus élevée que le modèle complet. Cela peut indiquer que le processus de sélection de variables a retiré des prédicteurs qui contribuent à la précision du modèle.

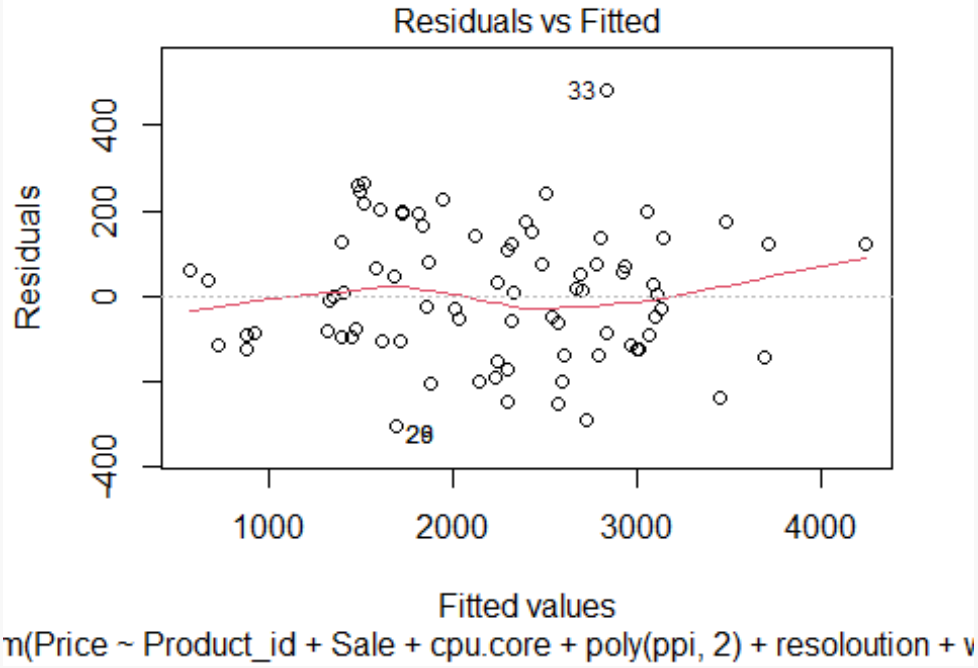
- Modèle $R^2$ ajusté (modelAdjR2) : Le modèle sélectionné pour maximiser le $R^2$ ajusté a un MSE de 42162.06, ce qui est plus élevé que les deux premiers modèles. Bien que ce modèle ait été conçu pour maximiser la variance expliquée tout en ajustant pour le nombre de prédicteurs, il semble moins performant pour prédire les prix sur de nouvelles données.

- Modèle Cp de Mallows (modelCp) : Le modèle basé sur le critère de Mallows a le MSE le plus élevé de 44203.23. Bien que ce modèle ait été choisi pour un équilibre théorique optimal entre complexité et ajustement, dans la pratique, il semble moins bien généraliser que les autres modèles testés.

En conclusion, sur la base de ces MSE, le modèle AIC semble s'approcher au plus du modèle complet pour la prédiction des prix dans l'échantillon de test.

## 3° Comparaison des modèles et analyse de l'importance des variables

Régression polynomiale




Erreur dans la prédiction

Dans cette régression polynomiale, plusieurs variables explicatives sont transformées en polynômes jusqu'au second degré pour capturer des relations potentiellement non-linéaires avec le prix des téléphones mobiles. Voici une interprétation détaillée des résultats obtenus :

Résultats de la régression :

- Résidus : La distribution des résidus montre que la médiane est très proche de zéro (-2.33), ce qui suggère que le modèle n'est pas biaisé et prédit assez précisément les prix. Les valeurs extrêmes des résidus (-304.04 à 482.28) indiquent certaines prédictions éloignées des valeurs réelles, signalant des cas où le modèle pourrait être amélioré.

- poly(ppi, 2) (premier terme polynomial de ppi): Très significatif avec un grand coefficient positif, indiquant que la densité de pixels a un effet notable et croissant sur le prix.

- poly(thickness, 2) (premier terme polynomial de l'épaisseur): Significativement négatif, montrant que des variations dans l'épaisseur peuvent réduire le prix, peut-être reflétant des préférences pour des téléphones plus fins.

- Autres variables comme cpu.core, internal.mem, Front_Cam et battery montrent également une influence positive significative sur le prix.

- Variables comme Product_id, Sale, resoloution, et RearCam ne montrent pas de signification statistique, suggérant que ces facteurs pourraient avoir des effets moins directs ou moins importants sur le prix dans ce modèle.

Performances du modèle :

- $R^2$ ajusté de 0.9572 montre que le modèle explique environ 95.72% de la variation des prix, ce qui est exceptionnellement élevé, témoignant de l'efficacité du modèle à capturer les dynamiques du prix.

- Erreur standard résiduelle de 161.9 indique la variabilité des erreurs du modèle, et bien que relativement faible, il y a toujours une marge pour réduire ces erreurs et améliorer la précision.

- F-statistique très élevée avec un p-value extrêmement faible (< 2.2e-16) confirme que le modèle dans son ensemble est statistiquement significatif.

- L'erreur quadratique moyenne (MSE) pour ce modèle sur l'échantillon de test est calculée à partir des prédictions du modèle. La valeur de 35359.91 indique l'erreur moyenne au carré entre les prix prédits et les vrais prix. Cette erreur est assez élevée, ce qui peut indiquer des surajustements ou des caractéristiques des données de test non bien captées par le modèle.

Ces résultats suggèrent que malgré l'excellente performance du modèle dans l'échantillon d'apprentissage, il pourrait y avoir des améliorations nécessaires pour mieux généraliser sur de nouvelles données, comme ajuster la complexité du modèle ou intégrer d'autres variables ou transformations qui pourraient mieux refléter les dynamiques des prix des téléphones mobiles.

Régression sans coefficients non significatifs :

- La régression est réalisée sur des données où certains coefficients jugés non significatifs ont été exclus.

- Le modèle utilise des variables explicatives telles que l'identifiant du produit, les ventes, le nombre de cœurs de CPU, la résolution, le poids, la fréquence du CPU, la mémoire interne, la RAM, la caméra frontale, l'épaisseur de la batterie, et leurs interactions polynomiales.

Diagnostics du modèle :

- Les graphiques de diagnostic aident à évaluer la qualité de l'ajustement du modèle et les hypothèses sous-jacentes.

- Le graphique Residuals vs Fitted montre les résidus par rapport aux valeurs ajustées. Une dispersion aléatoire des points autour de la ligne horizontale à 0 suggère que la variance des erreurs est constante (homoscédasticité).

Performance du modèle :

- Le R-squared ajusté de 0.9576 indique que le modèle explique environ 95.76% de la variabilité de la variable dépendante, ce qui est considéré comme très élevé.

- Le F-statistique est significatif ($p < 2.2e-16$), indiquant que le modèle est globalement significatif.

4. Erreur empirique :

- Vous calculez l'erreur empirique comme la moyenne des carrés des différences entre les prédictions du modèle et les valeurs réelles de prix (sur les données de test, je présume). Cela mesure la précision des prédictions du modèle sur de nouvelles données.

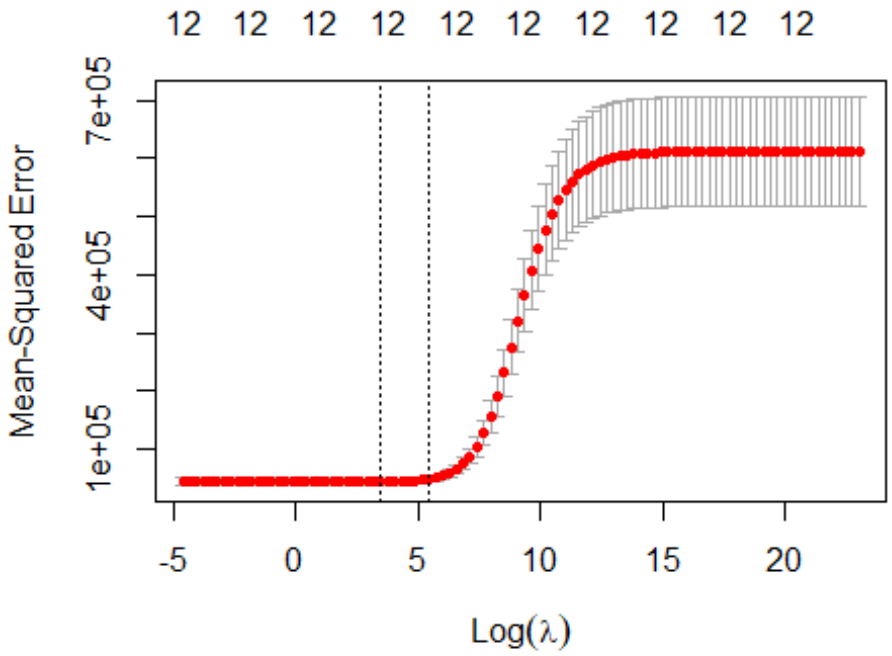
Dans l'ensemble, le modèle semble bien ajusté avec des diagnostics indiquant des résidus bien comportés.

Compilation des erreurs de prédictions :

Dans cette partie de l'analyse, nous avons évalué et comparé les performances de différents modèles statistiques. Pour ce faire, nous avons rassemblé les indicateurs de performance communs — Cp, $R^2$ ajusté et AIC — ainsi que les erreurs de prédiction mesurées par le MSE pour trois configurations de modèle : un modèle complet, un modèle polynomial et un modèle polynomial réduit.

Ensuite, nous avons calculé le RMSE, qui est la racine carrée du MSE, pour obtenir une mesure de l'erreur de prédiction qui est dans les mêmes unités que la variable dépendante. Les valeurs obtenues pour le RMSE nous donnent un aperçu direct de la précision moyenne des prédictions de chaque modèle. Plus spécifiquement, les valeurs du RMSE nous montrent que le modèle polynomial et le modèle polynomial réduit ont des erreurs de prédiction similaires et sont légèrement plus précis que le modèle complet, comme en témoignent leurs RMSE plus bas. Cela suggère que la réduction de la complexité du modèle n'a pas compromis la précision de la prédiction de manière significative et peut en fait l'avoir légèrement améliorée.

Modèle Ridge



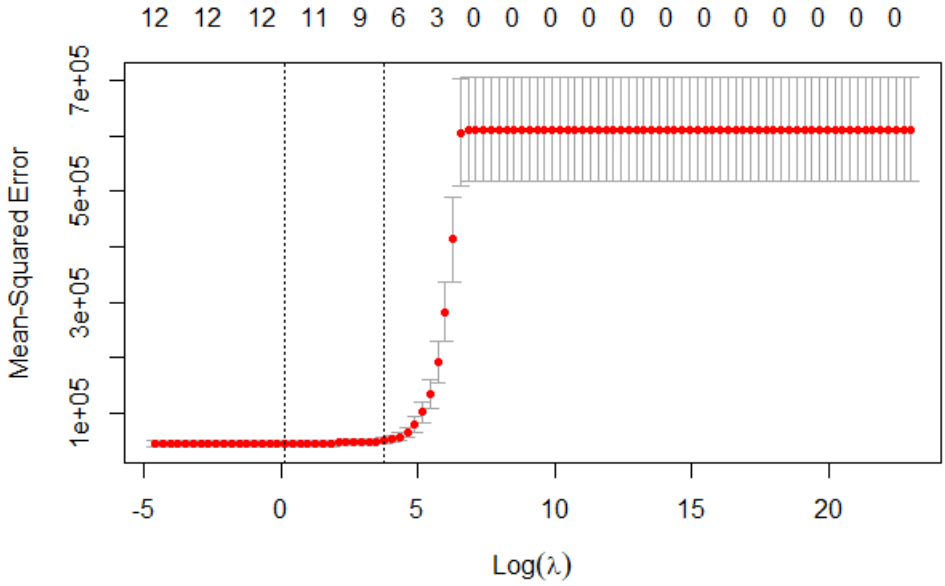

La régression Ridge, également connue sous le nom de régression avec régularisation L2, est une technique de régression linéaire qui ajoute une pénalité à la somme des carrés des coefficients de régression. Contrairement à la régression Lasso qui peut conduire à la sélection de variables, la régression Ridge réduit simplement la magnitude des coefficients.

Dans cette section de l'analyse, on a mis en œuvre une régression Ridge pour améliorer la prédiction en régularisant le modèle. Une suite de valeurs pour le paramètre de régularisation lambda a été générée, allant de 10^10 à 10^-2, pour trouver le compromis optimal entre biais et variance.

En utilisant la fonction cv.glmnet, on a réalisé une validation croisée pour sélectionner la meilleure valeur de lambda, avec alpha fixé à 0 pour spécifier une pénalité de type Ridge. Le graphique issu de cette validation croisée montre le MSE en fonction de log(lambda), où le point le plus bas correspond à la meilleure valeur de lambda (lambda.min), indiquant l'équilibre optimal pour le modèle.

Après avoir identifié la meilleure valeur de lambda, on a ajusté le modèle Ridge avec celle-ci et réalisé des prédictions sur un ensemble de données de test. Pour quantifier l'erreur de prédiction du modèle, on a calculé l'erreur empirique comme la moyenne des carrés des écarts entre les valeurs prédites et les valeurs réelles, résultant en une erreur de 36080.4. Cela donne une mesure de la performance du modèle lorsqu'il est confronté à de nouvelles données et permet d'évaluer sa capacité à généraliser.

Modèle Lasso




La régression Lasso, ou régression avec régularisation L1, est une technique de régression linéaire qui ajoute une pénalité à la somme des valeurs absolues des coefficients de régression. Cette pénalité encourage la simplification du modèle en réduisant certains coefficients à zéro, ce qui peut conduire à une sélection automatique des caractéristiques.

Pour cette partie de l'analyse, on s'est concentré sur la régression Lasso pour introduire une régularisation et potentiellement sélectionner des variables en attribuant des coefficients exactement nuls aux moins importantes. Après avoir fixé une graine aléatoire pour la reproductibilité avec set.seed(100), on a ajusté le modèle Lasso en utilisant une séquence de valeurs de lambda et un paramètre alpha de 1 pour une pénalité Lasso pure avec cv.glmnet.

Le processus a généré quelques avertissements indiquant que les paramètres xvar et label n'étaient pas reconnus lors de la tentative de création de certains graphiques, signifiant probablement une erreur dans la syntaxe ou dans la fonction utilisée.

Une fois la meilleure valeur de lambda déterminée par validation croisée, on a ajusté le modèle Lasso avec cette valeur. Des prédictions ont été effectuées sur les données de test et l'erreur empirique a été calculée, ce qui a donné une erreur de 36124.7 pour le modèle Lasso.

Enfin, on a compilé les erreurs de prédiction des modèles Ridge et Lasso dans un tableau pour comparer leurs performances. L'erreur MSE pour Ridge était de 36080.4, légèrement inférieure à celle du modèle Lasso, ce qui suggère que le modèle Ridge pourrait être légèrement plus précis dans ce cas particulier.

Arbre de décision

Dans cette partie de l'analyse, nous avons évalué et comparé les performances de différents modèles statistiques. Pour ce faire, nous avons rassemblé les indicateurs de performance communs — Cp, $R^2$ ajusté et AIC — ainsi que les erreurs de prédiction mesurées par le MSE pour trois configurations de modèle : un modèle complet, un modèle polynomial et un modèle polynomial réduit.

Ensuite, nous avons calculé le RMSE, qui est la racine carrée du MSE, pour obtenir une mesure de l'erreur de prédiction qui est dans les mêmes unités que la variable dépendante. Les valeurs obtenues pour le RMSE nous donnent un aperçu direct de la précision moyenne des prédictions de chaque modèle. Plus spécifiquement, les valeurs du RMSE nous montrent que le modèle polynomial et le modèle polynomial réduit ont des erreurs de prédiction similaires et sont légèrement plus précis que le modèle complet, comme en témoignent leurs RMSE plus bas. Cela suggère que la réduction de la complexité du modèle n'a pas compromis la précision de la prédiction de manière significative et peut en fait l'avoir légèrement améliorée.

Dans cette section de l'analyse, on a mis en œuvre une régression Ridge pour améliorer la prédiction en régularisant le modèle. Une suite de valeurs pour le paramètre de régularisation lambda a été générée, allant de 10^10 à 10^-2, pour trouver le compromis optimal entre biais et variance.

En utilisant la fonction cv.glmnet, on a réalisé une validation croisée pour sélectionner la meilleure valeur de lambda, avec alpha fixé à 0 pour spécifier une pénalité de type Ridge. Le graphique issu de cette validation croisée montre le MSE en fonction de log(lambda), où le point le plus bas correspond à la meilleure valeur de lambda (lambda.min), indiquant l'équilibre optimal pour le modèle.

Après avoir identifié la meilleure valeur de lambda, on a ajusté le modèle Ridge avec celle-ci et réalisé des prédictions sur un ensemble de données de test. Pour quantifier l'erreur de prédiction du modèle, on a calculé l'erreur empirique comme la moyenne des carrés des

écarts entre les valeurs prédites et les valeurs réelles, résultant en une erreur de 36080.4. Cela donne une mesure de la performance du modèle lorsqu'il est confronté à de nouvelles données et permet d'évaluer sa capacité à généraliser.

Pour cette partie de l'analyse, on s'est concentré sur la régression Lasso pour introduire une régularisation et potentiellement sélectionner des variables en attribuant des coefficients exactement nuls aux moins importantes. Après avoir fixé une graine aléatoire pour la reproductibilité avec set.seed(100), on a ajusté le modèle Lasso en utilisant une séquence de valeurs de lambda et un paramètre alpha de 1 pour une pénalité Lasso pure avec cv.glmnet.

Le processus a généré quelques avertissements indiquant que les paramètres xvar et label n'étaient pas reconnus lors de la tentative de création de certains graphiques, signifiant probablement une erreur dans la syntaxe ou dans la fonction utilisée.

Une fois la meilleure valeur de lambda déterminée par validation croisée, on a ajusté le modèle Lasso avec cette valeur. Des prédictions ont été effectuées sur les données de test et l'erreur empirique a été calculée, ce qui a donné une erreur de 36124.7 pour le modèle Lasso.

Enfin, on a compilé les erreurs de prédiction des modèles Ridge et Lasso dans un tableau pour comparer leurs performances. L'erreur MSE pour Ridge était de 36080.4, légèrement inférieure à celle du modèle Lasso, ce qui suggère que le modèle Ridge pourrait être légèrement plus précis dans ce cas particulier.

L’analyse de l’arbre de décision que nous avons réalisée utilise la méthode de partition récursive avec le contrôle des paramètres minsplit et cp pour ajuster la complexité de l’arbre. Le résumé de l'arbre montre le nombre optimal de divisions ainsi que la complexité de chaque nœud et l'erreur quadratique moyenne associée.

Nous avons déterminé l'importance des différentes variables dans la prédiction du prix. Par exemple, la RAM et la mémoire interne sont parmi les variables les plus influentes, ce qui signifie qu'elles jouent un rôle majeur dans la détermination du prix.

L’arbre de décision est construit en faisant des divisions binaires des variables prédictives. Par exemple, le premier nœud se divise sur la variable RAM, où une valeur inférieure à 1.75 va à gauche et une valeur supérieure va à droite. Les divisions subséquentes se font de manière similaire, en cherchant à maximiser l'amélioration apportée par chaque division, ce qui est mesuré par le paramètre improve.

Des divisions de substitution sont également identifiées, qui sont utilisées lorsque les données pour la variable principale de division sont manquantes. Ces divisions permettent à l'arbre de rester robuste même en présence de données incomplètes.

En examinant la structure de cet arbre, nous pouvons comprendre les relations entre les caractéristiques des téléphones et leur prix. Cela fournit des insights précieux pour la prise de décision et la stratégie de tarification dans le contexte commercial.

L'analyse détaillée de l'arbre de décision révèle comment chaque variable contribue à la prédiction des prix. Les observations sont réparties en nœuds basés sur les caractéristiques telles que le PPI, le poids et la résolution, avec chaque division cherchant à minimiser l'erreur quadratique moyenne (MSE).

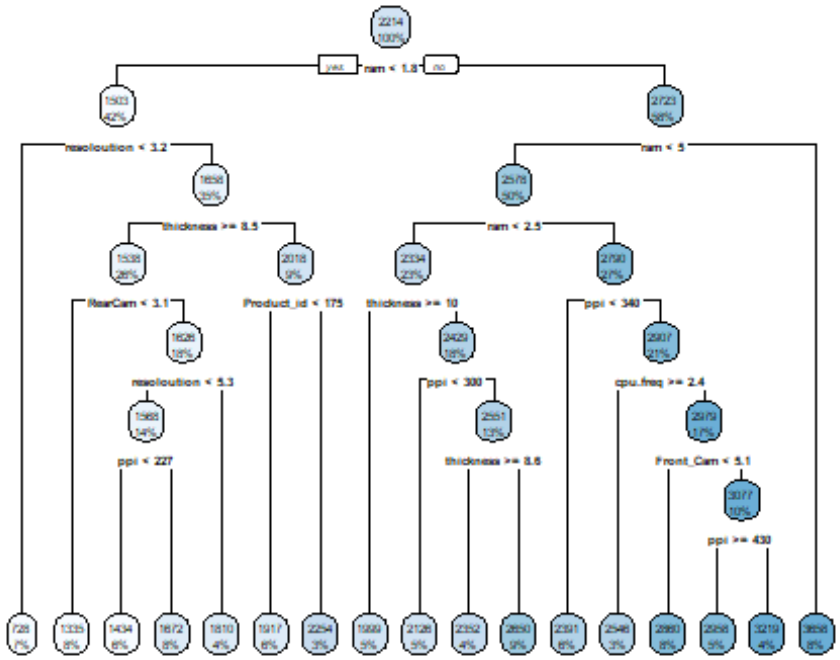
Dans les noeuds terminaux, on observe la valeur moyenne des observations, qui est utilisée pour prédire le prix lorsque les conditions de chaque nœud sont remplies. Les splits primaires montrent la condition de division et l'amélioration apportée par celle-ci, tandis que les splits de substitution offrent une alternative lorsque les données de la division principale sont manquantes.

L'erreur MSE associée à chaque nœud terminal donne une idée de la précision de la prédiction à ce stade. En examinant ces noeuds, on peut déduire les caractéristiques les plus influentes sur le prix et comment elles interagissent entre elles pour déterminer le prix final. Cette méthode de partitionnement récursif est utile pour identifier des modèles complexes et non linéaires dans les données.

La dernière partie de l'analyse de l'arbre de décision montre la sélection du modèle optimal à travers un processus d'élagage basé sur le coût de complexité (cp). Le graphique de complexité illustre l'évolution de l'erreur relative en fonction de différentes valeurs de cp, permettant de sélectionner la valeur de cp qui équilibre la précision du modèle et sa complexité.

L'arbre élagué est visualisé, montrant les divisions finales utilisées pour prédire le prix. Chaque nœud représente une règle de décision basée sur les attributs des téléphones, et les nœuds terminaux donnent la valeur prédite moyenne du prix. Ce modèle réduit permet d'éviter le surajustement tout en maintenant une bonne capacité prédictive.

La performance du modèle élagué est également évaluée en calculant le RMSE, qui mesure la différence moyenne entre les valeurs prédites et les valeurs réelles. Cette métrique permet de juger de la précision des prédictions de l'arbre de décision sur les données de test.




Random Forrest

L'analyse avec la forêt aléatoire a commencé par déterminer le nombre optimal d'arbres. Après avoir fixé la graine aléatoire pour assurer la reproductibilité des résultats, la forêt aléatoire a été construite avec 1000 arbres. L'erreur empirique a été calculée pour évaluer la performance du modèle.

Le graphique montre que l'erreur de prédiction se stabilise après un certain nombre d'arbres, indiquant qu'augmenter le nombre d'arbres au-delà de ce point n'apporte pas d'amélioration significative. Ainsi, le nombre optimal d'arbres a été fixé à 250.

Ensuite, l'importance des variables a été évaluée, révélant que la RAM est la variable la plus influente dans la prédiction des prix, suivie par la mémoire interne et la fréquence du CPU. Les graphiques d'importance des variables montrent l'effet de chaque variable sur l'erreur de prédiction et la pureté des nœuds, fournissant un aperçu visuel de la contribution de chaque variable à la performance du modèle.

Cette analyse fournit des informations précieuses sur les caractéristiques les plus déterminantes du prix et sur la robustesse du modèle de forêt aléatoire dans le contexte des données analysées.

## CONCLUSION

Nous avons mené une exploration complète des techniques de modélisation prédictive pour estimer le prix des téléphones portables. Notre travail a inclus la mise en œuvre et la comparaison de régressions multiples, des méthodes de régression Ridge et Lasso, de l'arbre de décision, et de la forêt aléatoire. Nous avons soigneusement ajusté chaque modèle, veillé à interpréter la pertinence des variables explicatives et à prévenir le surajustement. Nos modèles ont mis en lumière les facteurs clés qui influencent les prix et ont permis de déterminer la méthode de modélisation la plus précise. Cette analyse a été une opportunité d'appliquer de manière pratique des concepts de data mining et de consolider notre compréhension des approches prédictives en analyse de données.

# ANNEXE (commandes)

```
data <- read.csv("Cellphone.csv") #charger les données
library(knitr)
library(tidyverse)
library(corrplot)
library(caret)
library(leaps)
library(scales)
library(rpart)
library(glmnet)
library("ISLR")
library(tidyverse)
library(car)
library(Metrics)
library(randomForest)
library(rpart.plot)
# Vérifier et observer le bon format de la base de données
dim(data)
str(data)
head(data)
summary(data)
correlation <- cor(data)
corrplot(correlation, type = 'upper')
# découpage de l'échantillon en échantillon d'apprentissage et de test
set.seed(100)
indxTrain = createDataPartition(data$Price,p=0.70,list=FALSE)
LAtrain= data[indxTrain,] # Echantillon d'apprentissage
LAtest = data[-indxTrain,] # Echantillon de test
Reg<-lm(Price~.,data=LAtrain)
summary(Reg)
```

```
res=residuals(Reg)

plot(Reg,which=1:2)

####Modèle au sens du critère AIC

RegAIC= step(Reg, trace=TRUE)

extractAIC(RegAIC)

modelAIC = lm(Price ~ weight + ppi + cpu.core + cpu.freq + internal.mem + ram + battery
+ thickness, data=LAtrain)

summary(modelAIC)

extractAIC(modelAIC)

####Modèle au sens du R2 ajusté

Price.choixR2=leaps(LAtrain[,2:13],LAtrain$Price,method="adjr2",nbest=1)

#Meilleur modèle

t = (Price.choixR2$adjr2==max(Price.choixR2$adjr2))

#Liste des variables explicatives du meilleur modèle

colnames(LAtrain)[Price.choixR2$which[t]]

#Modèle sélectionné selon le R2 ajusté

modelAdjR2 = lm(Price ~ Sale + resoloution + ppi + cpu.core + cpu.freq + internal.mem
+ RearCam + Front_Cam + battery, data=LAtrain)

summary(modelAdjR2)

####Modèle au sens du CP Mallows

#Recherche des meilleurs modèles au sens du Cp

Price.choix =leaps(LAtrain[,2:13],LAtrain$Price,method="Cp",nbest=1)

Price.choix$Cp

plot(Price.choix$size-1,Price.choix$Cp,ylab="Cp",xlab="Nombre de variables",

main="Risque pénalisé en fonction des variables")

#Meilleur modèle

t=(Price.choix$Cp==min(Price.choix$Cp))

colnames(LAtrain)[Price.choix$which[t]]

#Modèle sélectionné selon le Cp de Mallows

modelCp = lm(Price ~ Sale + resoloution + ppi + cpu.core + cpu.freq + internal.mem +
Front_Cam + battery, data=LAtrain)

summary(modelCp)

####Erreurs empiriques
```

```
#modèle complet

pred = predict(Reg, newdata= LAtest)

err= mean((pred-LAtest$Price)^2)

err

#modèle AIC

predAIC = predict(modelAIC, newdata= LAtest)

errAIC= mean((predAIC-LAtest$Price)^2)

errAIC

#modèle R2 ajusté

predAdjR2 = predict(modelAdjR2, newdata= LAtest)

errAdjr2= mean((predAdjR2-LAtest$Price)^2)

errAdjr2

#modèle Cp Mallows

predCP = predict(modelCp, newdata= LAtest)

errCP= mean((predCP-LAtest$Price)^2)

errCP

####Régression Polynomiale####

polyAdjR2 <- lm(Price ~ Product_id + Sale + cpu.core + poly(ppi, 2) + resoloution +
weight + cpu.freq + internal.mem + poly(ram, 2) + RearCam + Front_Cam + battery +
poly(thickness, 2), data = LAtrain)

summary(polyAdjR2 )

plot(polyAdjR2)

plot(polyAdjR2$residuals)

# erreur empirique

predpolyR2=predict(polyAdjR2, newdata= LAtest)

errpoly= mean((predpolyR2-LAtest$Price)^2)

#### Régression sans les coefficients non significatifs

RegReduc <- lm(Price ~ Product_id + Sale + cpu.core + poly(ppi, 2) + resoloution

+ weight + cpu.freq + internal.mem + poly(ram, 2) + Front_Cam + battery + poly(thickness,
2), data = LAtrain)

summary(RegReduc )

plot(RegReduc)

plot(RegReduc$residuals)
```

```
# Erreur empirique
predReduc=predict(RegReduc, newdata= LAtest)
errReduc= mean((predReduc-LAtest$Price)^2)
#### Compilation des erreurs de prédictions
table.erreur <- data.frame(row.names = c("CP","R2 ajusté","AIC", "Polynomial",
"Polynomial réduit"), "MSE"=c(errCP,errAdjr2,errAIC, errpoly, errReduc))
table.erreur$RMSE= sqrt(table.erreur$MSE)
table.erreur
####Modèle Ridge
# Modèle Ridge
lbd = 10^seq(10, -2, length = 100)
xtrain = model.matrix(Price ~ ., LAtrain)[, -14]
ytrain = LAtrain$Price
xtest = model.matrix(Price ~ ., LAtest)[, -14]
ytest = LAtest$Price
set.seed(100)
# Validation croisée
cv.ridge = cv.glmnet(xtrain, ytrain, alpha = 0, lambda = lbd)
cv.ridge
# Affichage de la validation croisée
plot(cv.ridge)
# Meilleure valeur de lambda
bestlam.ridge.min = cv.ridge$lambda.min
# Ajustement du modèle Ridge avec la meilleure valeur de lambda
fit.ridge = glmnet(xtrain, ytrain, alpha = 0, lambda = bestlam.ridge.min)
# Pr´ediction sur les donn´ees de test
pred.ridge = predict(fit.ridge, newx = xtest)
# Calcul de l’erreur empirique
err.ridge = mean((pred.ridge - ytest)^2)
err.ridge
####Modèle Lasso
```

```
set.seed(100)

lasso_fit <- cv.glmnet(xtrain,ytrain,alpha=1, lambda=lbd)

plot(lasso_fit, xvar = "lambda", label = TRUE)

bestlam.lasso.min=lasso_fit$lambda.min

lasso_fit_opt= glmnet(xtrain,ytrain,alpha=1,lambda =bestlam.lasso.min )

pred.lasso = predict(lasso_fit_opt,newx=xtest)

err.lasso = mean((pred.lasso-ytest)^2)

#Compilation erreurs de prédiction Ridge Lasso

table.erreur <- data.frame(row.names = c( "Ridge", "Lasso"),
"MSE"=c(err.ridge,err.lasso))

table.erreur

####Arbre de décision

ctrl = rpart.control(minsplit = 10, cp = 0.002)

reg1 = rpart(Price ~ .,control=ctrl, data= LAtrain)

summary(reg1)

print(reg1)

printcp(reg1)

plotcp(reg1)

cp.opt= 0.0026570

tree.opt = prune(reg1, cp = cp.opt,)

rpart.plot(tree.opt)

table_pred <- data.frame(reel = LAtest$Price, pred = predict(tree.opt, LAtest))

Rmse <- sqrt(mean((table_pred$reel-table_pred$pred)^2))

print(tree.opt)

rpart.plot(tree.opt)

# Random Forest

set.seed(12)

rf.sp=randomForest(Price~.,data = LAtrain, xtest = LAtest[,-13],

ytest=LAtest[,"Price"],ntree=1000,importance=TRUE)

plot(rf.sp)

ntree.opt = 250 # Le nombre d’arbre optimal est ntree.opt

set.seed(12)
```

```
rf.sp2=randomForest(Price~.,data = LAtrain, xtest = LAtest[,-13],
ytest=LAtest[,"Price"],ntree=ntree.opt,ntry=11,do.trace=TRUE,importance =TRUE)
pred.rf2 = rf.sp2$test$predicted
# Racine de l’erreur empirique
rmse.rf <-sqrt(mean((pred.rf2- LAtest$Price) ^2))
#Importance des variables
sort(round(importance(rf.sp2),2)[,1])
varImpPlot(rf.sp2,main="Average Importance plots")
```