TRAVAUX PRATIQUES ENCADRÉS

RAPPORT FINAL

IFI-Promotion 23

APPLICATION MACHINE LEARNING EN ÉCONOMÉTRIE

INSTITUT FRANCOPHONE INTERNATIONAL

UNIVERSITÉ NATIONALE DU VIETNAM

Auteur :
M.Mamadou B H Cissoko

Superviseur :
Dr. Ho Tuong Vinh

Remerciements

Tout d'abord je commence à remercier Dieu le tout Puissant, le très Miséricordieux, qui m'a donné la force, le courage et la persévérance tout au long de la réalisation de ce travail et que la paix et le salut soient sur notre prophète Mohammed (Paix et Salue d'ALLAH sur Lui et toute sa famille).. C'est avec une certaine émotion et une grande reconnaissance que je remercie l'Université Nationale du Vietnam à travers l'Institut Francophone International. La rencontre avec des professeurs brillants et des camarades remarquables m'a permis de progresser dans un domaine qui me passionne mais également de me sentir soutenu et sans oublié tous les employés de l'Institut Francophone International (IFI) qui ont fait preuve d'un professionnalisme remarquable tout au long de notre parcours. Ces périodes riches en découvertes resteront longtemps gravées dans ma mémoire, tout comme les méthodologies et les principes d'enseignement inculqués par les enseignants : la curiosité, le goût du travail et de l'effort, le sens de la persévérance, la volonté de se remettre en question etc... Autant de trésors qui me seront d'une utilité capitale tout au long de ma vie professionnelle comme quotidienne. Pour toutes ces différentes raisons, je tiens vous à exprimer ma profonde gratitude, ainsi qu'à toute l'équipe pédagogique. Ce souvenir de passage dans cette école restera pour toujours car elle m'a tant donnée. Grâce vous, j'aborde une nouvelle étape de ma vie avec confiance et dynamisme et à travers vos enseignements je suis plus serein a aborder une nouvelle vie avec plein de confiances.

Table des matières

Table des figures

Liste des sigles et acronymes

ML	*MACHINE LEARNING*
SCT	*SOMME DES CARRES TOTAUX*
SCR	*SOMME DES CARRES RÉSIDUELS*
SCE	*SOMME DES CARRES EXPLIQUES*
R^2	*COEFFICIENT DE DÉTERMINATION*

Chapitre 1

Introduction

1.1 Contexte

L'utilisation de techniques quantitatives en économie remonte probablement au 16ème siècle, comme le montre **Morgan (1990)**. Mais il faudra attendre le début du XXième siècle pour que le terme "économétrie" soit utilisé pour la première fois, donnant naissance à l'Econometric Society en 1933.Quant aux techniques de machine learning (apprentissage automatique ou apprentissage statistique) elles sont plus récentes et c'est à **Arthur Samuel**, considéré comme le père du premier programme d'auto-apprentissage, que l'on doit le terme "machine learning" qui le définit comme étant "a field of study that gives computer the ability to learn without being explicitly programmed". Parmi les premières techniques de machine learning , on peut penser à la théorie des assemblées de neurones proposée dans Hebb (1949) (qui donnera naissance au perceptron dans les années 1950, puis aux réseaux de neurones) dont Widrow Hoff (1960) montreront quinze ans plus tard les liens avec les méthodes de moindres carrées, aux SVM (support vector machine) et plus récemment aux méthodes de boosting qui toutes se basent sur des approches statistiques pour donner aux ordinateurs la capacité d' apprendre à partir de données, c'est à dire d'améliorer leurs performances à résoudre des taches sans être explicitement programmés pour chacune des tâches. Si les deux communautés ont grandi en parallèle, les données massives nous imposent de créer des passerelles entre elles, en rapprochant les "deux cultures" évoquées par Breiman (2001a), opposant la statistique mathématique (que l'on peut rapprocher de l'économétrie traditionnelle, comme le note Aldrich en 2010) à la statistique computationnelle, et à l'apprentissage machine de manière générale. Car la réalité économique quant à elle se présente comme une multitude de phénomènes économiques (croissance, production, répartitions, inflation etc...), dont il faudra les étudier pour déterminer les différents mécanismes universels permettant d'expliquer au mieux ces phénomènes à savoir leurs fonctionnements, pour effectuer cette étude les économistes ont fait recours à deux approches à savoir :

— **L'observation et l'induction (l'approche empirique) :**

C'est une approche qui consiste à la collection des informations sur la réalité économique que l'on cherche à étudier en utilisant deux méthodes à savoir l'observation et et l'interprétation des données afin de pouvoir extraire des généralisations à partir des données en utilisant la théorie des probabilités statistiques dans le but d'expliquer et de mesurer la réalité économique (enquêtes, sondages...), anthropologique qui vise à saisir les phénomènes économiques dans leur contexte historique et social.

— **L'abstraction et la déduction (l'approche théorique) :**

Cette étape repose sur l'observation et l'analyse des phénomènes observés afin de pouvoir tirer des conclusions logiques et cette explication des phénomènes repose sur l'induction [André et al., 2018].
En raison des nouvelles technologies informatiques, beaucoup d'applications ont été crées comme par exemple la reconnaissance d'images, la prise de voix et traduction dans une autre langue par Google translator mais aussi la reconnaissance vocale par Siri vue toutes ces avancées prouve que l'apprentissage par la machine d'aujourd'hui n'est pas comparable à celui du passé.Il est né de la reconnaissance des formes et de la théorie selon laquelle les ordinateurs peuvent apprendre sans être programmés pour effectuer des tâches spécifiques d'où son but est de comprendre la structure des données afin de les intégrer dans des modèles qui peuvent être compris et utilisés par tout le monde de plus l'idée principale de ces percées est aussi statistique que computationnelle. L'intelligence artificielle IA est devenue possible lorsque les chercheurs ont cessé d'aborder les taches de façon procédurale et ont commencé à les aborder de manière empirique pour savoir si les ordinateurs pourraient apprendre des données. L'aspect itératif de l'apprentissage automatique est important car, à mesure que les modèles sont exposés à de nouvelles données, ils peuvent s'adapter indépendamment. Ils tirent des enseignements de calculs antérieurs pour produire des décisions et des résultats fiables et reproductibles. Au point de vue de ces traits communs entre ses deux disciplines on pourra dire que l'économétrie et les techniques d'apprentissage statistique supervisé de Machine learning semblent avoir une finalité en commun car elles cherchent à construire un modèle de prédiction c'est à dire une fonction $m : X \rightarrow Y$ qui sera interprétée comme une prévision sur la base d'un phénomène naturel à partir d'éventuels évènements aléatoires (ce qui consiste à prédire une variable dépendante « Endogène » à partir des variables indépendantes « Exogènes» [Michie et al., 1994]. [1] [2] [3]

1. https://fr.wikipedia.org/wiki/Apprentissage_automatique
2. https://openclassrooms.com/en/courses/4011851-initiez-vous-au-machine-learning/
3. http://eric.univ-lyon2.fr/~ricco/cours/cours_econometrie.html

Chapitre 2

État de l'art

Dans cette section nous parlons des traits communs entre les méthodes empiriques traditionnelles d'économétrie (qui est une recherche basée sur l'expérimentation ou l'observation 'évidence') car elle est utilisée enfin de tester une hypothèse et les techniques d'apprentissage de machine dans le sens ou à partir d'ensemble de jeux de données représenté en couple (X_i, Y_i) avec X_i définit comme étant la variable indépendante et Y_i la variable dépendante qui est la variable à prédire. Sur la base de laquelle on va construire un modèle de prédiction en spécifiant une fonction de corrélation entre les variables indépendantes X_i.....X_n et la variable dépendante Y_i qu'on cherchera à prédire m : **f(x)**. Mais si cette similarité existe entre ces deux cultures, il y a aussi des différences entre elles dans leurs processus à suivre enfin d'arriver à la prédiction de la variable dépendante Y_i (variable endogène).

2.1 Approche empirique traditionnelle économique

Un modèle économique est une représentation simplifiée d'un phénomène observé sous forme d'équations dont les variables sont des grandeurs économiques à la suite des quels on cherchera à expliquer, la corrélation entre ses différentes variables et la causalité dans le but d'estimer la variable dépendante, afin de le comprendre, de le reproduire et de le prévoir. Cette représentation quantitative d'un phénomène économique est fondée sur des hypothèses concernant le comportement des agents impliqués d'où l'inférence statistique ce qui nous permettra d'obtenir des généralisations à partir en utilisant la théorie des hypothèses. Car c'est à partir d'une réflexion théorique, elle sera amenée à imaginer des mécanismes d'interaction entre les différentes variables économiques du phénomène qu'on cherche à prédire. Comme toute approche, la finalité étant de vérifier si une théorie est valide ou pas, et de cette finalité découle deux sortes de régression à savoir : **régression simple** dans laquelle on cherchera à expliquer un phénomène par un autre phénomène et la **régression multiple** dans laquelle un phénomène est expliqué par deux ou plusieurs autres phénomènes. Dans une modélisation empirique traditionnelle on fait face à quatre types de pro-

blèmes qui sortent du lot

2.1.1 Problèmes Typiques

1. **Choix des variables explicatives :** objectif de cette contribution est de proposer une méthodologie permettant d'optimiser la phase de sélection des variables explicatives dans les modèles de régressions utilisés à des fins exploratoires, c'est-à-dire en l'absence de modèle théorique. La démarche consiste dans un premier temps, en la recherche systématique de toutes les variables statistiquement liées à la variable dépendante : les variables candidates. Dans un second temps, les inter-corrélations entre candidates sont analysées, afin de ne conserver que les variables présentant un faible taux d'inter-corrélation c'est à dire les variables explicatives afin d'éviter le problème de multicolinéarité.
   a) Descriptif : Le premier objectif constitue la partie la plus importante lors d'une modélisation qui est le choix des variables à considérées dans le modèle, c'est-à-dire quelles sont les variables qui expliqueraient au mieux la variable endogène (variable dépendante). À cette stratégie, à laquelle peuvent contribuer des analyses en composantes principales **PCA**, correspondant à des algorithmes de recherche (pas à pas) moins performants mais économiques en temps de calcul si **p** est grand. mais au cas ou, **n** est petit, dès lors la recherche devient suffisamment longue avec beaucoup de variables explicatives, il sera toujours possible de trouver un modèle expliquant y; c'est l'effet data mining dans les modèles économétriques appelé maintenant data snooping.
   b) Explicatif : Le deuxième objectif est sous-tendu par une connaissance à priori du domaine concerné et dont des résultats théoriques peuvent vouloir être confirmés, infirmés ou précisés par l'estimation des paramètres. Dans ce cas, les résultats inférentiels permettent de construire le bon test conduisant à la prise de décision recherchée. Utilisées hors de ce contexte, les statistiques de test n'ont qu'une valeur indicative au même titre que d'autres critères plus empiriques.
   c) Prédictif : Dans le troisième cas, l'accent est mis sur la qualité des prévisions. C'est la situation rencontrée en apprentissage. Ceci conduit à rechercher des modèles parcimonieux c'est-à-dire avec un nombre volontairement restreint de variables explicatives pour réduire la variance. Le modèle ainsi obtenu peut favoriser des estimateurs biaisés au profit d'une variance plus faible même si le théorème de **Gauss-Markov** indique que, parmi les estimateurs sans biais, celui des moindres carrés est de variance minimum. Un bon modèle n'est donc plus celui qui explique le mieux les données au sens d'un R2 maximum mais celui conduisant aux prévisions les plus fiables.
2. **Choix d'une forme fonctionnelle :** C'est une démarche consistant à savoir quelle forme est la mieux adaptée au problème compte tenant des différentes variables indépendantes (explicatives) disponibles afin de mieux modéliser le phénomène. Car souvent dans une modélisation empirique, c'est une fonction linéaire entre variables qui est la plus adaptée et la plus utilisée avec l'utilisation de la méthode

des moindres carrés ordinaires (MCO) car ça convient au mieux aux phénomènes à tendance linéaire parce que nous chercherons à prédire une variable dépendante à partir des variables indépendantes donc un problème de corrélation.

3. **Choix d'une distribution de probabilité :** Avoir une distribution de probabilité spécifique s'avère être un choix prépondérant dans la construction d'un modèle. Car tous les phénomènes économiques ne sont pas toujours spécifiés, donc pour des échantillons avec peu d'observations la puissance des tests statistiques pourrait être faible.
4. **Inclusion des interactions :** En effet ce problème découle du fait de la non-linéarité entre variables évoquées au deuxième obstacle. Le problème étant la difficulté de pouvoir modéliser un grand nombre d'interaction entre les différentes variables avec ces méthodes. Les 4 obstacles décrits précédemment sont des problèmes liés de façon étroite à la modélisation empirique, et se retrouvent être renforcés par l'avènement des données massives (Big data).

[1]

2.1.2 La formulation des problèmes économiques

2.1.2.1 Modèle de régression simple

C'est le modèle dans lequel on cherche à modéliser la relation entre deux variables quantitatives continues c'est-à-dire expliquer une variable dépendante Y(variable endogène) par rapport à une variable indépendante X (variable exogène), car le modèle revient à supposer, qu'en moyenne, E(Y), est une fonction affine de X. L'écriture du modèle suppose implicitement une notion préalable de causalité dans le sens où Y dépend de X car le modèle n'est pas symétrique.[Guyader, 2011]
E(y) = f(X) = $\beta_0 + \beta_1 x$ ou y = $\beta_0 + \beta_1 x + \epsilon$ (1)

- y est la variable à expliquer (à valeurs dans R) ;
- x est la variable explicative (à valeurs dans R) ;
- ϵ est le terme d' erreur aléatoire du modèle ;
- β_0 et β_1 sont deux paramètres à estimer.

Pour n observations , on peut écrire le modèle de régression linéaire simple sous la forme :
$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$ Dans ce cas, on suppose que :

- ϵ_i est une variable aléatoire, non observée.
- x_i est observée et non aléatoire.
- y_i est observée et aléatoire.

1. http://www.bsi-economics.org/

2.1.2.2 Modèle de régression multiple :

La régression linéaire multiple est la forme la plus courante d'analyse de régression linéaire. En tant qu'analyse prédictive, la régression linéaire multiple est utilisée pour expliquer la relation entre une variable dépendante continue et deux variables indépendantes ou plus. Les variables indépendantes peuvent être continues ou catégoriques (codées factices selon le cas).Elle constitue la généralisation naturelle de la régression simple. Une variable quantitative Y dite à expliquer (endogène, dépendante) est mise en relation avec p variables quantitatives $X_1, \ldots, X_p$ dites explicatives (exogènes ou encore indépendantes). Les données sont supposées provenir de l'observation d'un échantillon statistique de taille n (n supérieur à p + 1) de $R^(p+1)$:
$(x_1^i, \ldots, x_j^i, \ldots, x_p^i, y_i)$ i = 1, . . ., n.
L'écriture du modèle linéaire dans cette situation conduit à supposer que l'espérance de Y appartient au sous-espace de R^n engendré par :
1, X^1 , . . ., X^p où 1 désigne le vecteur de R n constitué de 1.
C'est à dire que les (p + 1) variables aléatoires vérifient :
$y_i = \beta_0 + \beta_1 x_i^1 + \beta_2 x_i^2 + \ldots\ldots + \beta_p x_i^p + \epsilon_i$ avec i = 1, 2 . . . n (2) avec les hypothèses suivantes :

- Les ϵ_i sont des termes d' erreurs non observés indépendants et identiquement distribués ; $E(\epsilon_i) = 0$, $Var(\epsilon) = \sigma^2$ I.
- 2. Les termes x j sont supposés déterministes (facteurs contrôlés) ou bien l'erreur ϵ est indépendante de la distribution conjointe de X_1 , . . . , X_p. On écrit dans ce dernier cas que :
  $E(Y|X_1, \ldots, X_p) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p$ et $Var(Y|X_1, \ldots, X_p) = \sigma^2$.(3)
- Les paramètres inconnus $\beta_0, \ldots, \beta_p$ sont supposés constants
- En option , pour l' étude spécifique des lois des estimateurs , une quatrième hypothèse est considère la normalité de la variable d'erreur avec :ϵ (N (0, σ^2 I)).[Guyader, 2011]

2.1.3 La résolution des phénomènes économiques

2.1.3.1 Estimation des coefficients avec la méthode des moindres carrés :

L'estimateur des moindres carrés ordinaires reste l'un des estimateurs les plus fréquemment utilisés dans une modélisation empirique. Il a de nombreux usages. On peut l'utiliser par exemple pour procéder à une description des données : quelles sont les variables rendant compte le mieux de la variabilité d'une variable d'intérêt. On peut aussi l'utiliser dans de nombreuses autres situations pour estimer un paramètre auquel on donne un sens causal : c'est à dire que se passerait-il si on faisait varier une variable donnée d'un montant donné.Il est basé sur l'hypothèse essentielle que les résidus et les variables explicatives sont orthogonaux. La méthode des moindres carrés, indépendamment élaborée par **Legendre et Gauss** au début du $19^{ème}$ siècle, permet de comparer des données expérimentales, généralement entachées d'erreurs de mesure,

à un modèle mathématique censé décrire ces données. Ce modèle peut prendre diverses formes. Il peut s'agir de lois de conservation que les quantités mesurées doivent respecter. La méthode des moindres carrés permet alors de minimiser l'impact des erreurs expérimentales en « ajoutant de l'information » dans le processus de mesure. On généralise ou adapte le plus souvent les propriétés de l'estimateur à la condition que l'hypothèse centrale d'absence de corrélation entre perturbations et variables explicatives soit maintenue. Donc Notre objectif est de chercher d'estimer les paramètres suivants :

- β_0 et β_1 par la méthode des moindres carrés.
- Estimation de β_0 et β_1 par les moindres carrés.

Les points (x_i, y_i) étant donnés, le but est maintenant de trouver une fonction affine f telle que la quantité

$$\sum_{i=1}^{n} L(y_i - f(x_i))$$

Soit minimale. Pour pouvoir déterminer f, encore faut-il préciser la fonction de coût L. Deux fonctions sont classiquement utilisées

- le coût absolu L(u) = |u| ;
- le coût quadratique L(u) = u^2.

Les deux ont leurs vertus, mais on privilégiera dans la suite la fonction de coût quadratique. On parle alors de méthode d'estimation par moindres carrés (terminologie due à Legendre dans un article de 1805 sur la détermination des orbites des comètes).

$$S(\beta_1, \beta_2) = \sum_{i=1}^{n} (y_i - \beta_1 - \beta_2 x_i)^2.$$

Autrement dit, la droite des moindres carrés minimise la somme des carrés des distances verticales des points (x_i, y_i) du nuage à la droite ajustée

$$y = \hat{\beta}_1 + \hat{\beta}_2 x.$$

La fonction de deux variables S est une fonction quadratique et donc on cherchera sa minimisation.

$$\hat{\beta}_1 = \bar{y} - \hat{\beta}_2 \bar{x},$$

avec :

$$\hat{\beta}_2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})y_i}{\sum_{i=1}^{n}(x_i - \bar{x})^2}.$$

La première méthode consiste à remarquer que la fonction S(β_1, β_2) est strictement convexe, donc qu'elle admet un minimum en un unique point pour les coefficients estimés (β_1, β_2) , lequel est déterminé en annulant les dérivées partielles de S. On obtient les "équations normales"

$$\begin{cases} \dfrac{\partial S}{\partial \beta_1} &= -2\displaystyle\sum_{i=1}^{n}(y_i - \hat{\beta}_1 - \hat{\beta}_2 x_i) = 0 \\ \dfrac{\partial S}{\partial \beta_2} &= -2\displaystyle\sum_{i=1}^{n} x_i(y_i - \hat{\beta}_1 - \hat{\beta}_2 x_i) = 0 \end{cases}$$

La première équation donne :

$$\hat{\beta}_1 n + \hat{\beta}_2 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$

d'où l'on déduit immédiatement :

$$\hat{\beta}_1 = \bar{y} - \hat{\beta}_2 \bar{x}, \quad (1.1)$$

où x¯ et y¯ sont comme d'habitude les moyennes empiriques des x_i et des y_i. La seconde équation donne :

$$\hat{\beta}_1 \sum_{i=1}^{n} x_i + \hat{\beta}_2 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i$$

et en remplaçant le coefficient β_1 estimé par son expression (1.1), nous avons :

$$\hat{\beta}_2 = \frac{\sum x_i y_i - \sum x_i \bar{y}}{\sum x_i^2 - \sum x_i \bar{x}} = \frac{\sum x_i (y_i - \bar{y})}{\sum x_i (x_i - \bar{x})} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})(x_i - \bar{x})}. \quad (1.2)$$

La seconde méthode consiste à appliquer la technique de Gauss de réduction des formes quadratiques, c'est-à-dire à décomposer S(β_1, β_2) en somme de carrés, carrés

qu'il ne restera plus qu'à annuler pour obtenir les coefficients des estimateurs β_1 et β_2. Dans notre cas, après calculs, ceci s'écrit :

$$S(\beta_1,\beta_2) = n\left(\beta_1 - (\bar{y} - \beta_2\bar{x})\right)^2 + \left(\sum_{i=1}^{n}(x_i-\bar{x})^2\right)\left(\beta_2 - \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}\right)^2 + \left(\sum_{i=1}^{n}(y_i-\bar{y})^2\right)\left(1 - \frac{\left(\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})\right)^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2\sum_{i=1}^{n}(y_i-\bar{y})^2}\right),$$

Il existe différents critères de jugement de la qualité d'ajustement du modèle. Le coefficient de détermination R², doit être calculé pour nous aider à dire si notre modèle est intéressant ou pas. L'objectif est de construire des estimateurs qui minimisent la somme des carrés des résidus. Pour cela nous aurons à passer par le calcul des différents termes qui nous permettra de pouvoir en déduire la valeur du coefficient de détermination R^2 :

1. **Estimateur des moindres carrés ordinaire MCO :** L'estimateur des moindres carrés ordinaires est défini comme le vecteur b de dimension K+1,b=(b_0,...,b_K) 0,des coefficients de la combinaison linéaire de e, x_1,...,x_K réalisant le minimum de la distance de y à l'espace vectoriel de R^N engendré par , x_1,...,x_K,pour la norme euclidienne [Crépon, 2005] :

$$\widehat{b}_{mco} = \bar{} \arg\min \left\| \underline{y} - \underline{x}b \right\|^2$$

Le coefficient de détermination R^2 : le coefficient de détermination, noté R^2 ou r^2, est une mesure de la qualité de la prédiction d'une régression linéaire. [2] Il est défini par :

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \in [0,1]$$

Où **n** est le nombre de mesures , y_i la valeur de la mesure n^o i y_i estimé la valeur prédite correspondante et y_- représentant la moyenne des mesures.
Dans le cas d'une régression linéaire univariée (une seule variable prédictive), on montre que la variance (totale) SST est la somme de la variance expliquée par la régression SSE et de la moyenne des carrés des résidus SSR, de sorte que :

$$R^2 = \frac{\mathrm{SSE}}{\mathrm{SST}} = \frac{\mathrm{SST} - \mathrm{SSR}}{\mathrm{SST}} = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

2. https://fr.wikipedia.org/wiki/Coefficient_de_détermination

c'est-à-dire que le coefficient de détermination est alors le rapport de la variance expliquée par la régression SSE sur la variance totale SST.

2. **Le coefficient de détermination R^2 ou coefficient ajusté :** Le coefficient de détermination ajusté tient compte du nombre de variables. En effet, le principal défaut du R² est de croître avec le nombre de variables explicatives. Or, on sait qu'un excès de variables produit des modèles peu robustes. C'est pourquoi on s'intéresse davantage à cet indicateur qu'au R². Mais ce n'est pas un véritable carré et il peut même être négatif. Voici deux expressions du R² ajusté, sachant que certains auteurs lui donnent une définition légèrement différente : [3]

$$\bar{R}^2 = 1-\left(1-R^2\right)\left[\frac{n-1}{n-\left(k+1\right)}\right]$$

   - Si $R^2 = 0$; le modèle n'explique rien, les variables X et Y ne sont pas corrélées linéairement.
   - Si $R^2 = 1$; les points sont alignés sur la droite, la relation linéaire explique toute la variation.
   - Une valeur de R^2 proche de 1 est nécessaire pour avoir un ajustement raisonnable mais cela ne veut pas dire que c'est suffisant.

3. **Somme des carrés totaux ou variance totale (SCT) :** Permet de minimiser l'erreur dans le modèle. Il traduit la variabilité totale de l'endogène ; La somme des carrés est une mesure de variation ou d'écart par rapport à la moyenne. Elle représente la somme des carrés des différences par rapport à la moyenne. Le calcul de la somme totale des carrés prend en compte les différences dues aux facteurs et de celles dues au hasard ou à l'erreur.
   Somme totale de carrés = somme des carrés de la régression (SCR) + somme des carrés de l'erreur résiduelle (SCER)

$$\Sigma(y-\overline{y})^2 = \Sigma(\hat{y}-\overline{y})^2 + \Sigma(y-\hat{y})^2$$

4. **Somme des carrés expliqués(SCE) :** somme des carrés expliqués, traduit la variabilité expliquée par le modèle.

3. https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/coefficient-of-determination-r-squared/.

$$SCE = \sum_{i=1}^{n} (\hat{y}_i - \overline{y})^2$$

Qui représente la variance expliquée par la régression (mesure la variation des valeurs ajustées autour de la moyenne de la variable endogène y) .

5. **Somme des carrés des résiduels ou variance résiduelle (SCR) :** somme des carrés résiduels correspond à la variabilité non-expliquée par le modèle. La somme des carrés des résidus (ou SCR) est une valeur statistique qui ne sert en soi à rien, mais qui peut être utilisée pour calculer d'autres mesures plus intéressantes. Comme par exemple dans une étude d'une série statistique, il est toujours intéressant de connaître le degré de corrélation des valeurs de la série et la SCR y contribue.

$$SCR = \sum_{i=1}^{n} \hat{e}_i^2$$

Représente la variance résiduelle ou non expliquée (partie de la variation totale qui n'est pas expliquée par le modèle de régression). [Besse, 2003]

- Si SCR= 0, X explique parfaitement Y
- Si SCE= 0, X n'explique en rien Y

4

4. https://support.minitab.com/fr-fr/minitab/18/help-and-how-to/modeling-statistics/anova/supporting-topics/anova-statistics/understanding-sums-of-squares/

Chapitre 3

Approche empirique algorithmique

3.1 Les méthodes de machine learning(Algorithmes)

Les algorithmes sont des ensembles d'instructions explicitement programmées utilisées par les ordinateurs pour calculer ou résoudre des problèmes. Les algorithmes d'apprentissage automatique permettent aux ordinateurs de s'entraîner sur les entrées de données et utilisent l'analyse statistique pour produire des valeurs qui se situent dans une plage spécifique. Pour cette raison, l'apprentissage automatique facilite l'utilisation des ordinateurs dans la construction de modèles à partir de données d'échantillonnage afin d'automatiser les processus de prise de décision en fonction des données saisies. Il y a deux types d'algorithme en machine learning à savoir : Les algorithmes supervisés fonctionnent en deux phases : une phase d'apprentissage, et une phase de traitement des données d'entrée et les algorithmes non supervisés fonctionnent sans phase d'apprentissage et fournissent directement une réponse à partir des données d'entrée. Utilisés en complément des algorithmes supervisés. [1]

3.1.1 Fonctionnement de l'apprentissage automatique et fonctions de perte :

L'objectif de l'apprentissage automatique est de comprendre la structure des données afin de les intégrer dans des modèles qui peuvent être compris et utilisés par tout le monde. Les algorithmes d'apprentissage automatique supervisés recherchent des fonctions qui prédisent un phénomène suivant un ensemble d'évènements aléatoires. Par exemple, déterminer la probabilité d'une maladie sur une personne à partir de symptômes ou mesurer les taux de succès des compagnes marketing, prédire les revenus d'un certain produit. Une photo peut être transformée en vecteur, disons un tableau de 100 par 100, de sorte que le vecteur x résultant ait 10 000 entrées. La valeur y est 1 pour les images avec un visage et 0 pour les images sans visage. La fonction de

1. https://www.supinfo.com/articles/single/6041-machine-learning-introduction

perte L (y, y) capture les gains d'une classification correcte ou inappropriée de «face» ou de «pas de visage».

3.1.2 But de l'utilisation des algorithmes :

Extraire et exploiter automatiquement l'information présente dans un jeu de donne es pour un objectif de généralisation d'un mode le, c'est-à -dire sa performance selon un critère choisi à priori sur des données nouvelles, et donc des tests hors échantillon. Les algorithmes non supervisés vont chercher à produire des représentations compactes des données, en regroupant les individus similaires. Les algorithmes ont simplement besoin de moyens de mesurer les proximités entre les individus. Pour les algorithmes supervise s, c'est un peu diffèrent. **X** représente les valeurs d'entrée. Il faudra leur adjoindre, pour chaque individu, un vecteur Y représentant les valeurs de sortie. Le but étant de décrire une relation liant **X** à **Y** permettant d'expliquer a mieux une relation linéaire.

3.1.2.1 Fonction de perte :

La fonction de perte (le terme perte a été utilisé pour la première fois par Wald, 1939) représente une certaine mesure de la différence entre les valeurs observées des données et les valeurs calculées à l'aide de la fonction d'ajustement C'est la fonction qui est minimisée dans la procédure d'ajustement d'un modèle. Pour un modèle, l'apprentissage signifie déterminer les bonnes valeurs pour toutes les pondérations et le biais à partir d'exemples étiquetés. Dans l'apprentissage supervisé, un algorithme de Machine Learning crée un modèle en examinant de nombreux exemples, puis en tentant de trouver un modèle qui minimise la perte. Ce processus est appelé minimisation du risque empirique. La perte correspond à la pénalité pour une mauvaise prédiction. Autrement dit, la perte est un nombre qui indique la médiocrité de la prévision du modèle pour un exemple donné. Si la prédiction du modèle est parfaite, la perte est nulle. Sinon, la perte est supérieure à zéro. Le but de l'entraînement d'un modèle est de trouver un ensemble de pondérations et de biais pour lesquels la perte, en moyenne sur tous les exemples, est faible. Pour un problème de régression, elle correspond à une erreur quadratique, et en classification, nous utilisons un indicateur de mauvaise qualification.

Fonction :©Perte

$$m^{\star}(\boldsymbol{x}) = \underset{m \in \mathcal{M}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \ell(y_i, m(\boldsymbol{x})) \right\}$$

2

2. http://www.statsoft.fr/concepts-statistiques/glossaire/f/fonction-perte.html

3.1.3 Boosting et Apprentissage séquentiel (Lent) :

L'idée du Boosting, tel qu'introduit par Shapire Freund (2012), est d'apprendre, lentement, à partir des erreurs du modèle, de manière itérative. À la première étape, on estime un modèle m_1 pour y, à partir de X, qui donnera une erreur ϵ_1. À la seconde étape, on estime un modèle m_2 pour ϵ_1, à partir de X, qui donnera une ϵ_2, etc. On va alors retenir comme modèle, au bout de k itération [3]

$$m^{(k)}(\cdot) = \underbrace{m_1(\cdot)}_{\sim \boldsymbol{y}} + \underbrace{m_2(\cdot)}_{\sim \boldsymbol{\epsilon}_1} + \underbrace{m_3(\cdot)}_{\sim \boldsymbol{\epsilon}_2} + \cdots + \underbrace{m_k(\cdot)}_{\sim \boldsymbol{\epsilon}_{k-1}} = m^{(k-1)}(\cdot) + m_k(\cdot).$$

Ici, l'erreur ϵ est vue comme la différence entre y et le modèle m(x), mais elle peut aussi être vue comme le gradient associé à la fonction de perte quadratique.

3.1.4 Sur-apprentissage et Pénalisation :

Le surapprentissage signifie que l'on construit un modèle trop complexe, qui aura de faibles qualités prédictives sur un nouvel échantillon (on parlera alors de généralisation). Il est en général provoqué par un mauvais dimensionnement de la structure utilisée pour classifier. De par sa trop grande capacité à stocker des informations, une structure dans une situation de sur apprentissage aura de la peine à généraliser les caractéristiques des données. Elle se comporte alors comme une table contenant tous les échantillons utilisés lors de l'apprentissage et perd ses pouvoirs de prédiction sur de nouveaux échantillons. On pourra remédier à ce problème en utilisant deux techniques qui sont : Validation croisée et la régularisation. La première technique consiste à séparer les données en deux sous-ensembles : l'ensemble d'apprentissage et l'ensemble de validation. L'ensemble de validation n'est pas utilisé pour l'apprentissage mais permet de vérifier la pertinence du réseau avec des échantillons qu'il ne connaît pas. La deuxième consiste à pénaliser les valeurs extrêmes des paramètres, car ces valeurs correspondent souvent à un surapprentissage. C'est une forme de régularisation ce qui consiste à ajouter de l'information a un problème pour éviter le surapprentissage. [Charpentier et al., 2017]

3.1.5 Les techniques de validation croisée :

La validation croisée est une procédure statistique qui produit une estimation de la compétence de prévision qui est moins biaisée que les estimations de compétence de prévision arrière habituelles. La méthode de validation croisée supprime systématiquement un ou plusieurs cas dans un jeu de données, dérive un modèle de prévision

3. https://freakonometrics.hypotheses.org/files/2017/07/econometrics-ML-final-1.pdf

des cas restants et le teste sur le ou les cas supprimés. La procédure est non paramétrique et peut être appliquée à toute technique de construction de modèle automatisée. Il peut également fournir des informations de diagnostic importantes sur les cas influents de l'ensemble de données et la stabilité du modèle. Dans la validation croisée, les données sont divisées en k sous-ensembles. Et à partir de là, la méthode de rétention est répétée k fois, de sorte qu'à chaque fois, l'un des k sous-ensembles est utilisé comme ensemble de test / validation et les autres sous-ensembles k-1 sont assemblés pour former un ensemble d'apprentissage. L'estimation de l'erreur est moyennée sur tous les k essais pour obtenir l'efficacité totale de notre modèle. Comme on peut le voir, chaque point de données se trouve dans un ensemble de validation exactement une fois, et obtient dans un ensemble d'entraînement k-1 fois. Cela réduit considérablement le biais dans notre modèle. Car nous utilisons la plupart des données pour l'ajustement, et réduit également la variance de manière significative du fait que la plupart des données sont également utilisées dans l'ensemble de validation. [4]

3.2 Travaux réalisés dans le domaine de Machine Learning en Économétrie

Sendhil Mullainathan and Jann Spiess, (2017), dans leur étude ils utilisent des algorithmes de data mining à savoir : les arbres de régression, le Lasso ainsi que les forets aléatoires pour résoudre un problème de prédiction des prix des maisons aux États-Unis, par conséquent ils ont manipulés les données dont la source est fournie par l'enquête statistique sur les logements aux États-Unis « American Housing Survey », en considérant 10000 unités choisies au hasard dans le Metropolitan Sample. En plus des valeurs de chaque unité, ils incluent également 150 variables qui contiennent des informations sur l'unité et son emplacement, tels que le nombre de chambres, la superficie de base et la région de recensement aux Etats-Unis. Pour comparer différentes techniques de prédiction, nous évaluons dans quelle mesure chaque approche prévoit la valeur unitaire sur un ensemble se pare de 41 808 unités du même échantillon.

4. https://fr.wikipedia.org/wiki/Validation_croisée

Performance of Different Algorithms in Predicting House Values

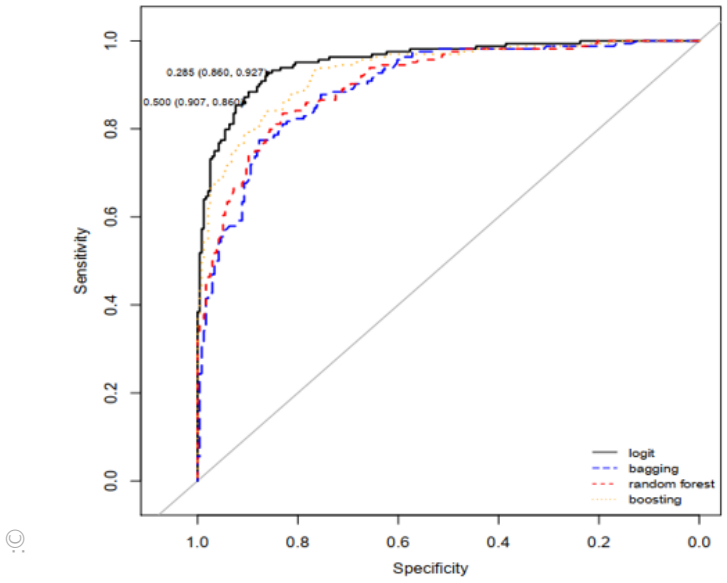
	Prediction performance (R^2)		*Relative improvement over ordinary least squares by quintile of house value*				
Method	*Training sample*	*Hold-out sample*	1st	2nd	3rd	4th	5th
Ordinary least squares	47.3%	41.7% [39.7%, 43.7%]	–	–	–	–	–
Regression tree tuned by depth	39.6%	34.5% [32.6%, 36.5%]	−11.5%	10.8%	6.4%	−14.6%	−31.8%
LASSO	46.0%	43.3% [41.5%, 45.2%]	1.3%	11.9%	13.1%	10.1%	−1.9%
Random forest	85.1%	45.5% [43.6%, 47.5%]	3.5%	23.6%	27.0%	17.8%	−0.5%
Ensemble	80.4%	45.9% [44.0%, 47.9%]	4.5%	16.0%	17.9%	14.2%	7.6%

Tableau 1-Performance des différents algorithmes dans la prédiction des valeurs de la maison

A la vue résultats obtenus avec l'utilisation des différents algorithmes on peut clairement voir la performance de l'algorithme du foret aléatoire ; elle surpasse les moindres carrés ordinaires sur le hold-out (hors échantillon) de plus de 9 pour cent en termes de R^2 global. Le foret aléatoire est une moyenné sur plusieurs arbres (dans ce cas, 700). Chaque arbre est ajusté sur un échantillon bootstrap de l'ensemble d'apprentissage original et contraint à un sous-ensemble aléatoire de variables. Les prédictions des arbres sont ensuite moyennées. Arthur Charpentier, Emmanuel Flachaire, Antoine Ly, (2017) ont récemment montré dans leur étude comment appliquer les différentes techniques d'apprentissage automatique (ML) à des problèmes réels à tendances économiques, pour mieux les expliquer tout en donnant de meilleurs résultats concernant les différents phénomènes. Concernant notre travail nous prendrons deux (2) applications des applications qu'ils ont eu à travailler.

3.2.1 Première Application 1 :

Les ventes de sièges auto pour enfants (classification) Nous reprenons ici un exemple utilisé dans James et al. (2013). Le jeu de données contient les ventes de sièges auto pour enfants dans 400 magasins (Sales), ainsi que plusieurs variables, dont la qualité de présentation en rayonnage (Shelveloc, égal à « mauvais », « moyen », « bon ») et le prix (Price).12 Une variable dépendante binaire est artificiellement créée, pour qualifier une forte vente ou non (High=« oui » si Sales > 8 et à « non» sinon). Dans cette application, on cherche à évaluer les déterminants d'un bon niveau de vente.

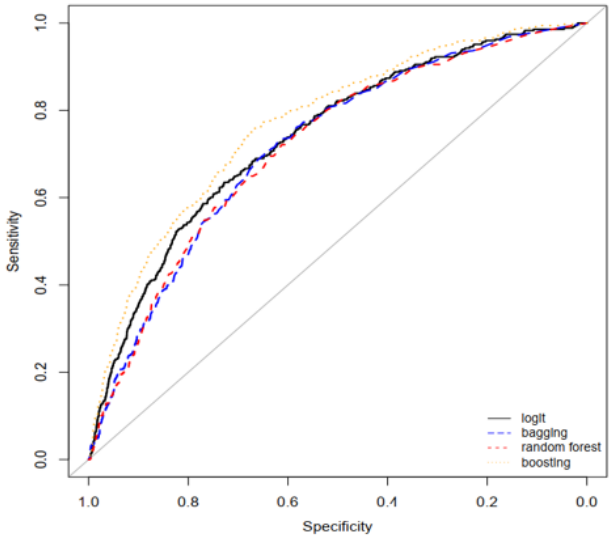


	AUC
logit	0.9544
bagging	0.8973
random forest	0.9050
boosting	0.9313

Dans cet exemple, on voit que la courbe ROC du modèle logit domine les autres courbes, et son aire sous la courbe est la plus grande (AUC=0.9544). Ces résultats indiquent que ce modèle fournit les meilleures prévisions de classification. N'étant dominé par aucun autre modèle, ce constat suggère que le modèle linéaire logit est correctement spécifié et qu'il n'est pas utile d'utiliser un modèle plus général et plus complexe c'est-à-dire l'utilisation d'un modèle algorithmique.

3.2.2 Deuxième Application 2 :

La deuxième application concerne l'achat d'une assurance caravane (classification) : Nous reprenons à nouveau un exemple utilisé dans James et al. (2013). Le jeu de données contient 85 variables sur les caractéristiques démographiques de 5822 individus.14 La variable dépendante (Purchase) indique si l'individu a acheté une assurance caravane, c'est une variable binaire, égale à « oui » ou « non



	AUC
logit	0.7372
bagging	0.7198
random forest	0.7154
boosting	0.7691

	0.5 cutoff		cutoff optimal	
	spécificité	sensitivité	spécificité	sensitivité
logit	0.9967	0.0057	0.7278	0.6351
bagging	0.9779	0.0661	0.6443	0.7069
random forest	0.9892	0.0316	0.6345	0.6954
boosting	0.9987	0.0000	0.6860	0.7385

Ils analysent leur étude par la construction de la courbe ROC (Receiver Operating Characteristics) qui est une représentation graphique présentant les performances d'un modèle de classification pour tous les seuils de classification car elle nous permet de tracer le taux de vrais positifs en fonction du taux des faux positifs. Sachant que la droite de Sensitivity représente le taux des vrais positifs(TVP) et Specificity celle du taux des faux positifs(TFP). La courbe du modèle boosting domine les autres courbes, son aire sous la courbe est la plus grande (AUC=0.7691). Ces résultats indiquent que le boosting fournit les meilleures prévisions de classification. Cet exemple comparé à l'exemple précédent, nous montre que les courbes sont assez éloignées de la forme en coude, ce qui suggère que la classification ne sera pas aussi bonne. [Charpentier et al., 2017]

3.2.3 Résumée des differents travaux :

Les différents travaux réalisés nous montrent qu'on pourra bien appliquer les techniques d'apprentissage automatique en économétrie afin d'estimer à mieux la variable endogène (la variable à expliquer) d'un phénomène économique et que les techniques d'apprentissage automatique fournissent des meilleures performances que les méthodes traditionnelles que la science économétrie utilise. Et vue la grande capacité des don-

nées il serait presque impossible pour les économistes de pouvoir réaliser une meilleure estimation ce qui n'est pas le cas des techniques d'apprentissage automatique car plus la capacité des données est grande plus la prédiction devient meilleure. Comme j'ai eu mentionné au début de ce travail ces deux domaines ont une similarité car leur but est de prédire une variable endogène (y) à partir d'une variable exogène (x) problème de régression simple ou par plusieurs variables explicatives (problème de régression multiple) ce qui nécessite de dire que les économistes doivent plus considérer les techniques d'apprentissage aux techniques traditionnelles et de plus l'attrait de l'apprentissage automatique est qu'il parvient à découvrir des modèles généralisés.

Chapitre 4

Solution Proposée

L'économétrie étant un ensemble de techniques appliquées à l'analyse des phénomènes économiques.L'analyse de ces phénomènes économiques vise essentiellement à mettre en évidence les mécanismes qui régissent ces phénomènes afin de mieux comprendre leur nature et leur fonctionnement, d'une part et de prévoir leur évolution d'autre part. L'économiste cherche dans sa démarche à caractériser les liens qui unissent les diverses variables intervenant dans l'explication des phénomènes économiques et, si possible, de dégager des lois de comportement sous–jacents. Au cours de cet processus d'analyse on peut être conforter a plusieurs obstacles dans la résolution d'un problème économique comme le problème de **choix d'une forme fonctionnelle**, **choix des variables descriptives** ou **explicatives**, **choix d'une distribution de probabilité** et **inclusion des interactions** entre les différentes variables explicatives qui sont les problemes majeurs que les économistes rencontrent dans la résolution des phénomènes de type économiques et l'avènement de **Big Data** rends ce problème encore plus complexe ce qui causera les limites de l'approche empirique traditionnelle de l'économétrie. Pour remédier a ces differents problemes les techniques de **Machine Learning** sont appropriées car elles sont efficaces et fournissent des meilleurs résultats en présence d'une grande quantité de données **Données massives** et elles sont meilleures dans le processus de sélection des variables explicatives expliquant au mieux le phénomène à traiter d'où l'importance pour les économistes de faire recourir aux techniques de **l'apprentissage automatique** pour traiter au mieux les phénomènes et effectuer une validation qui sera primordial pour la prise de décision.
C'est dans ce cadre qu'on cherchera à résoudre le problème avec les techniques de Machine Learning

4.1 Problématique :

Notre travail porte sur une étude menée en 2009 par **University of Minho, Guimarães Portugal** sous la supervision de **Portuguese "Vinho Verde" wine** une région spécialisée dans la culture des Vins qui a rendu publique le dataset sur la qualité des

Vins à savoir les deux types de Vins qui cultivent (Rouges et Blancs) comportant 6497 observations des différentes qualités de Vins et 11 variables explicatives pour prédire la qualité du Vin basée sur les facteurs physico-chimiques c'est-à-dire la quantité densité de l'eau , du sucre, de l'alcool, sulfate, l'acide , pH (phosphate d'hydrogène) etc... qui faut avoir avoir lors de la production d'un Vin pour qu'il soit le vin soit meilleur et aussi de chercher a savoir quels sont les facteurs pouvant influencer cette qualité de façon négative comme positive **(Meilleure ou mauvaise qualité)**.
Le dataset étant divisé en deux dataset différents :

- WhiteWine.csv : contenant toutes les observations sur les Vins blancs.
- RedWine.csv : contenant toutes les observations sur les Vins rouges.

4.2 Algorithmes et Outils utilisés :

4.2.1 Forets aléatoires (Random Forest)

Les forêts aléatoires ou les forêts de décision aléatoires sont une méthode d'apprentissage d'ensemble pour la classification, la régression et d'autres tâches fonctionnant en construisant une multitude d'arbres de décision au moment de la formation et en générant la classe qui est le mode des classes (classification) ou la prédiction moyenne (régression). Des forêts de décision aléatoires corrigent l'habitude de sur-adapter des arbres de décision à leur ensemble d'entraînement.
Le premier algorithme pour les forêts à décision aléatoire a été créé par **Tin Kam Ho** à l'aide de la méthode du sous-espace aléatoire, qui, dans la formulation de Ho, est un moyen de mettre en œuvre l'approche de "discrimination stochastique" proposée par **Eugene Kleinberg**. Une extension de l'algorithme a été développée par **Leo Breiman** et **Adele Cutler**, qui a enregistré "Random Forests" en tant que marque (à compter de 2019, propriété de Minitab, Inc.). Cette extension associe l'idée de "mise en sac" de Breiman et une sélection aléatoire de caractéristiques, introduites d'abord par **Ho**, puis de manière indépendante par **Amit** et **Geman**, afin de constituer une collection d'arbres de décision à variance contrôlée. [1]

Soit Y une variable cible ou à prédire et X une variable prédictive (ou Co variable), éventuellement de grande dimension. L'objectif général de l'analyse statistique est d'inférer (déduire), d'une manière ou d'une autre, la relation entre Y et X. Les forets aléatoires estiment une valeur μ(x) de la moyenne conditionnelle E (Y | X=x) de la variable cible Y, donnée X=x. Elles se développent un ensemble de forêts de plus de 500 arbres avec n observations indépendantes (Y_i, X_i), i=1,..., n. L'algorithme de forets aléatoires tient en compte plusieurs **paramètres** et **attributs** que l'utilisateur doit définir pour avoir une relation expliquant de mieux Y_i et X_i , Car Une forêt aléatoire est un méta-estimateur qui correspond à un certain nombre de classificateurs d'arbres de décision

1. https://en.wikipedia.org/wiki/Random_forest

sur divers sous-échantillons de l'ensemble de données et utilise la moyenne pour améliorer la précision prédictive et contrôler le sur-ajustement. La taille du sous-échantillon est toujours identique à la taille de l'échantillon d'origine, mais les échantillons sont dessinés avec remplacement si bootstrap = True (par défaut). :

- **n-estimators : integer, optional (default=10) :** Le nombre d'arbres à construire dans la forêt.
- **criterion : string, optional (default="gini") :** La fonction pour mesurer la qualité d'une scission. Les critères pris en charge sont "gini" pour l'impureté de Gini et "entropie" pour le gain d'information. Remarque : ce paramètre est spécifique à l'arbre.
- **max-depth : integer or None, optional (default=None) :** La profondeur maximale de l'arbre. Si aucun, les nœuds sont développés jusqu'à ce que toutes les feuilles soient pures ou que toutes les feuilles contiennent moins que min-samples-split samples
- **min-samples-split : int, float, optional (default=2) :** Le nombre minimal d'échantillons requis pour fractionner un nœud interne :
  * Si int, considérons min-samples-split comme nombre minimal.
  * Si float, min-samples-split est une fraction et ceil (min-samples-split * n-samples) correspond au nombre minimal d'échantillons pour chaque groupe.
- **min-samples-leaf : int, float, optional (default=1) :** Nombre minimal d'échantillons requis pour un nœud feuille. Un point de partage à n'importe quelle profondeur ne sera pris en compte que s'il laisse au moins des échantillons d'entraînement min-samples-leaf dans chacune des branches gauche et droite. Cela peut avoir pour effet de lisser le modèle, en particulier lors de la régression. * Si int, considérons min-samples-leaf comme nombre minimal. * Si float, min-samples-leaf est une fraction et ceil (min-samples-leaf * n-samples) correspond au nombre minimal d'échantillons pour chaque nœud.
- **min-weight-fraction-leaf : float, optional (default=0.) :** La fraction pondérée minimale de la somme totale des poids (de tous les échantillons d'entrée) devant être au niveau d'un nœud feuille. Les échantillons ont un poids égal lorsque sample-weight n'est pas fourni.
- **max-features : int, float, string or None, optional (default="auto") :** nombre de variables tire es aléatoirement pour constituer l'ensemble dans lequel sera sélectionnée la variable de segmentation de chaque nœud, souvent égal à n/3.
  * Si int, considère les fonctionnalités max-features à chaque division.
  * Si float, max-features est une fraction et les entités int (max-features * n-features) sont prises en compte à chaque division.
  * Si «auto», max-features = sqrt (n-features).
  * Si «sqrt», max-features = sqrt (n-features) (identique à «auto»).

* Si «log2», max-features = log2 (n-features).
* Si aucune, max-features = n-features

Remarque : La recherche d'un fractionnement ne s'arrête pas tant qu'il n'a pas trouvé au moins une partition valide des exemples de noeud, même si elle nécessite d'inspecter efficacement plus de fonctionnalités que max-features.

- **max-leaf-nodes : int or None, optional (default=None) :** Permet de sélectionner des arbres avec max-leaf-nodes de la meilleure façon qui soit. Les meilleurs nœuds sont définis comme une réduction relative de l'impureté. Si aucun, nombre illimité de nœuds terminaux.
- **bootstrap : boolean, optional (default=True) :** Si des échantillons bootstrap sont utilisés lors de la construction des arbres. Si False, l'ensemble de données entier est utilisé pour construire chaque arbre.
- **oob-score : bool (default=False) :** S'il faut utiliser des échantillons hors du sac pour estimer la précision de la généralisation
- **n-jobs : int or None, optional (default=None) :** Nombre de travaux(processus) à exécuter en parallèle pour les ajustements et les prévisions.
- **verbose : int, optional (default=0) :** Contrôle la verbosité lors de l'ajustement et de la prédiction.
- **min-impurity-decrease : float, optional (default=0.) :** Permet de diviser chaque noeud(variable) pour savoir quelle sont celles qui sont importantes au modèle tout en dressant un arbre traçant cette importance de la variable.
- **min-impurity-split : float, (default=1e-7) :** Seuil d'arrêt précoce de la croissance des arbres. Un nœud se divisera si son impureté est supérieure au seuil, sinon c'est une feuille. Ce seuil représente le nombre des variables qu'on veut sélectionner dans le modèle.

2

4.2.2 Fonctionnement

Considérons d'abord la formation d'un seul arbre, le partitionnement récursif standard commencerait par toutes les données et effectuerait une recherche exhaustive sur toutes les variables et tous les points de division pour trouver celui qui expliquerait au mieux l'ensemble de données en réduisant au maximum l'impureté du nœud. Alors les données sont dévissées en fonction du meilleur point de partage et le processus est répété de gauche à droite jusqu'à ce que certaines règles d'arrêts soient remplies notamment les nombres d'échantillons requit dans une feuille et dans un nœud, chaque fois que l'algorithme fait une division, toutes les variables sont incluses dans la recherche. Dans le cadre d'un problème de régression , un nouveau point de donnée (c'est à dire la

2. `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html`

variable la plus importante) X = x est prédite à partir d'un seul arbre de Random Forest comme étant la moyenne pondérée des observations originales Y_i, i=1,. . . ,n :

$$\hat{\mu}(\mathbf{x}) = \sum_{\mathbf{i=1}}^{\mathbf{n}} \omega_{\mathbf{i}}(\mathbf{x}, \theta)\mathbf{Y_i}$$

Avec ω_i(x, θ), vecteur de poids donné par une constante positive si l'observation Xi fait partie de la même feuille de l'arbre construite à partir du vecteur aléatoire des variables θ dans lesquelles la variable X a été abandonnée et donné par une valeur 0 sinon.

En utilisant les forets aléatoires, la moyenne conditionnelle E (Y|X=x) est approximée par la prédiction moyenne de k arbres uniques, chacun construit avec un vecteur indépendant et identique distribue t, t=1,. . . , k. Soit ω_i(x) la moyenne de ω_i(T) de l'ensemble de tous les arbres de la foret, et la prédiction de Random Foret est donnée par :

$$\hat{\mu}(\mathbf{x}) \sum_{\mathbf{i=1}}^{\mathbf{n}} \omega_{\mathbf{i}}(\mathbf{x})\mathbf{Y_i}$$

[Breiman, 2001].

4.2.3 Quelques détails sur la méthode choisie :

La méthodologie pour entraîner et évaluer notre modèle est présentée en figure ci-dessus. La base de données (le dataset) initiale détient toutes les données brutes (les observations) .Nous diviserons cette base des données en deux parties, d'une part une base de données d'évaluation qui servira à calculer les indicateurs de performance de référence pour tous les modèles construits dans cette étude. D'autre part, seront sélectionnées un pourcentage de cette bases de données qui seront utilisées pour entraîner notre modèle (qui serviront à la construction des modèles) qui seront ensuite testées pour évaluer le modèle.

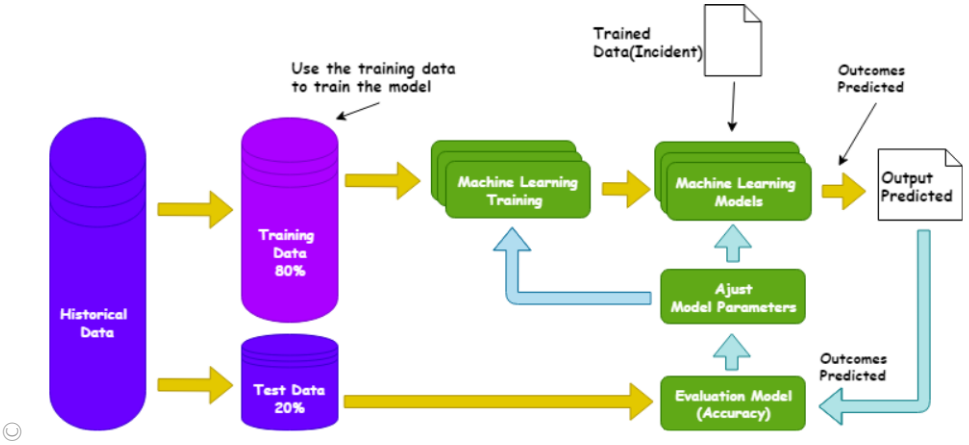


FIGURE 4.1 – Architecture du Modèle (Random Forest)

4.2.3.1 Pré-Traitement des variables

Cette phase consiste aux différentes étapes de traitement des variables explicatives de notre dataset car il y a des données manquantes, mais aussi des données de type catégoriques que nous traiterons pour que nous puissions les utilisées.

- **Données manquantes :** Pour traiter les données manquantes, nous remplaçons les données manquantes par la moyenne de l'ensemble des observations de la même colonne. Car c'est une méthode d'imputation très basique.Il s'agit de la seule fonction testée qui ne tire pas parti des caractéristiques de la série chronologique ou de la relation entre les variables. C'est très rapide, mais présente des inconvénients évidents. Un inconvénient est que l'imputation moyenne réduit la variance dans l'ensemble de données.
- **Données catégoriques :** Nous utiliserons la technique de **One-hot Encoding** pour convertir l'attribut type de Vin . le schéma de codage one-hot, code ou transforme l'attribut en m entités binaires qui ne peuvent contenir qu'une valeur de 1 ou 0. Chaque observation dans la catégorique l'entité est ainsi convertie en un vecteur de taille m avec une seule des valeurs égale à 1 (indiquant qu'elle est active) . Cette technique nous permettra d'éviter la piège des variables factice dans notre modèle à savoir le **Dummy variable trap**.

4.2.3.2 Sélection de toutes les variables utilisant Random Forest :

Les forêts aléatoires étant constituées de 4 à 1 200 arbres de décision, chacun d'eux étant construit sur une extraction aléatoire des observations de l'ensemble de données et une extraction aléatoire des entités. Tous les arbres ne voient pas toutes les caractéristiques ni toutes les observations, ce qui garantit que les arbres sont dé-corrélés et

donc moins enclins à trop s'ajuster. Chaque arbre est également une séquence de questions oui-non basées sur une ou plusieurs combinaisons de caractéristiques. À chaque nœud (c'est à chaque question), les trois divisent le jeu de données en 2 compartiments, chacun d'entre eux hébergeant des observations plus similaires entre elles et différentes de celles de l'autre compartiment. Par conséquent, l'importance de chaque caractéristique découle de la «pureté» de chacun des compartiments.
Pour la classification, la mesure de l'impureté est soit l'impureté de Gini, soit le gain / entropie d'informations. Pour la régression, la mesure de l'impureté est la variance. Par conséquent, lors de la formation d'un arbre, il est possible de calculer combien chaque caractéristique diminue l'impureté. Plus une caractéristique diminue l'impureté, plus elle est importante. Dans les forêts aléatoires, il est possible de faire la moyenne de la diminution d'impuretés de chaque entité pour déterminer l'importance finale de la variable. Les variables se trouvant en haut sont celles qui sont les plus importantes pour le modèle suivant le seuil d'impureté. [3]

4.2.4 Évaluations des performances des modèles :

Nous construisons deux modèles en appliquant l'algorithme de foret aléatoire sur notre base de données :

- **Premier modèle** :Avec toutes les variables explicatives du dataset.
- **Deuxième modèle** : Avec sélection des variables (Variables importantes).

Afin de répondre aux objectifs fixés du dataset mais aussi les objectifs fixés en chapitre 2 , plusieurs ajustements seront faits pour améliorer nos modèles afin d'avoir un résultat proche des observations réelles .
L'évaluation des performances de ces modèles s'effectue par le biais d'indicateurs d'incertitude, parmi lesquels, nous avons sélectionné les plus utilisés. Ils sont de deux types :

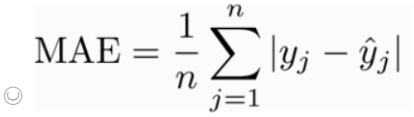
- **Les indicateurs fournis par le modèle :** l'erreur quadratique moyenne du modèle **(MSE)** qui mesure la moyenne des carrés des erreurs - c'est-à-dire la différence quadratique moyenne entre les valeurs estimées et ce qui est estimé.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2$$

FIGURE 4.2 – MSE

3. https://towardsdatascience.com/feature-selection-using-random-forest-26d7b747597f

- **Les indicateurs calculés à partir de l'ensemble de validation du modèle (Évaluation) :**
  *** Mean Absolute Error (MAE) :** qui mesure la magnitude moyenne des erreurs dans un ensemble de prévisions, sans tenir compte de leur direction. C'est-à-dire dire la moyenne sur l'échantillon test des différences absolues entre la prévision et l'observation réelle où toutes les différences individuelles ont un poids égal. [4]



$$\mathrm{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

FIGURE 4.3 – MAE

*** Coefficient de détermination** (R^2) **:** Le coefficient de détermination peut être considéré comme un pourcentage. Cela nous donne une idée du nombre de points de données compris dans les résultats de la ligne formée par l'équation de régression , c'est aussi une mesure de performance du modèle.
*** Coefficient de détermination** (R^2) **ajusté :** est une mesure de performance qui nous permet de savoir si on a besoin de plus des données pour un meilleur modèle. Car la partie qui explique la variabilité du modèle par rapport à l'ensemble des observations contenant dans le dataset. [5]

4.2.5 Détails Techniques :

Pour implémenter notre solution nous avons utilisé les bibliothèques et packages suivantes :
Bibliothèques Utilisés avec Anaconda Python 3 :

1. **Numpy :** Est une librairie du langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques.
2. **Pandas :** Fournit des structures de données rapides, flexibles et expressives conçues pour rendre le travail avec des données «relationnelles» ou «étiquetées» dans notre contexte à la fois facile et intuitif. Il a pour but d'être le bloc de construction fondamental de haut niveau pour l' analyse pratique et réaliste des données en Python . L'objet DataFrame permet de manipuler des données aisément et efficacement avec des index pouvant être des chaînes de caractères incluant les outils pour lire et écrire des données structurées en mémoire depuis et vers différent formats de fichier.

4. https://www.statisticshowto.datasciencecentral.com/absolute-error/
5. https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/coefficient-of-determination-r-squared/

3. **Sklearn.preprocessing** : est une bibliothèque libre Python dédiée à l'apprentissage automatique , nous l'utiliserons pour faire partitionner notre dataset en test et training set (Données d'entraînement et données d'évaluation), mais aussi pour encoder nos variables catégoriques.
4. **Matplotlib :**est une bibliothèque de traçage Python 2D qui produit des figures de qualité. Il permet aussi de générer des graphiques, histogrammes, spectres de puissance, graphiques à barres, diagrammes d'erreurs et des diagrammes de dispersion, etc.
5. **JupyterLab notebook :** Qui est beaucoup utilisé pour l'apprentissage car c'est un environnement qui est interactif, tu as la possibilité de voir directement l'effet des codes saisis.
6. **RandomForestClassifier package :** paquet pour les forets aléatoires.
7. **Anaconda Distribution Open Source** qui est un plateforme le plus simple et le plus utilisé pour les tests des algorithmes de machine learning il est aussi utilisé pour la science des données Python / R et l'apprentissage automatique sur Linux, Windows et Mac OS X.
8. **Python** Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions.

Environnement matériel :

— Ordinateur portable Lenovo ThinkPad W541 Processeur : Intel(R) Core(TM) i7-4810MQ CPU @ 2.80GHz, 2794 Mhz, 4 Core(s), 8 Logical Processor(s)

— RAM : 16.00 Go

— OS : UBUNTU 16.4

4.2.6 Description des Résultats Obtenus :

4.2.6.1 Méthode de validation :

A l'origine beaucoup de fonctionnalités peuvent nous donner un bon classifieur, mais cela peut finir par surclasser notre modèle et ne pas pouvoir généraliser de nouvelles données. L'un des approches de base que nous avons suivi est la réduction des fonctionnalités . Pour éviter les sur-ajustements dans notre modèle avec les forêts aléatoires, la principale chose à faire est d'optimiser un paramètre de réglage qui gouverne le nombre d'entités choisies de manière aléatoire pour développer chaque arbre à partir des données initialisées. Pour ce faire, vous allons effectué généralement cette opération via une validation croisée des plis multiples, où $k \in \{5,10\}$, et choisir le paramètre de réglage qui minimise l'erreur de prédiction de l'échantillon test.En outre, la croissance d'une forêt plus grande améliorera la précision des prévisions.

4.2.6.2 Facteurs déterminants à la qualité du vin :

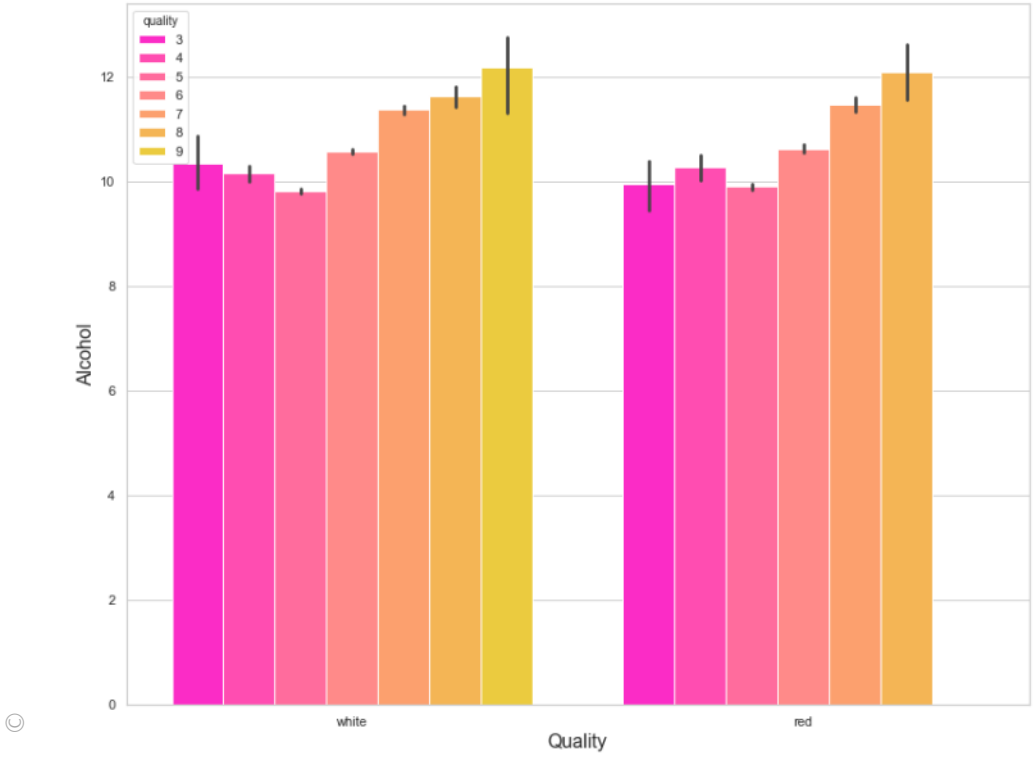


FIGURE 4.4 – Variable alcool dans la qualité du Vin

L'alcool se forme à la suite de la transformation du sucre par la levure au cours du processus de fermentation. Le pourcentage d'alcool peut varier d'un vin à l'autre. Nous interprétons l'alcool en utilisant différents récepteurs de goût, c'est pourquoi il peut avoir un goût à la fois amer, sucré, épicé et gras. Nos gènes jouent en réalité un rôle dans le goût de l'alcool amer ou sucré. Quoi qu'il en soit, nous pouvons tous percevoir l'alcool comme une sensation de réchauffement au fond de la bouche.Les vins plus alcoolisés ont tendance à avoir un goût plus gras et les vins moins alcoolisés ont tendance à avoir un goût plus léger et corsé Ce qui montre que la qualité du vin est directement liée à la quantité d'alcool dans le vin. Plus la quantité de l'alcool est importante dans le vin plus sera meilleure la qualité .

4.2.6.3 Facteurs déterminants à la qualité du vin :

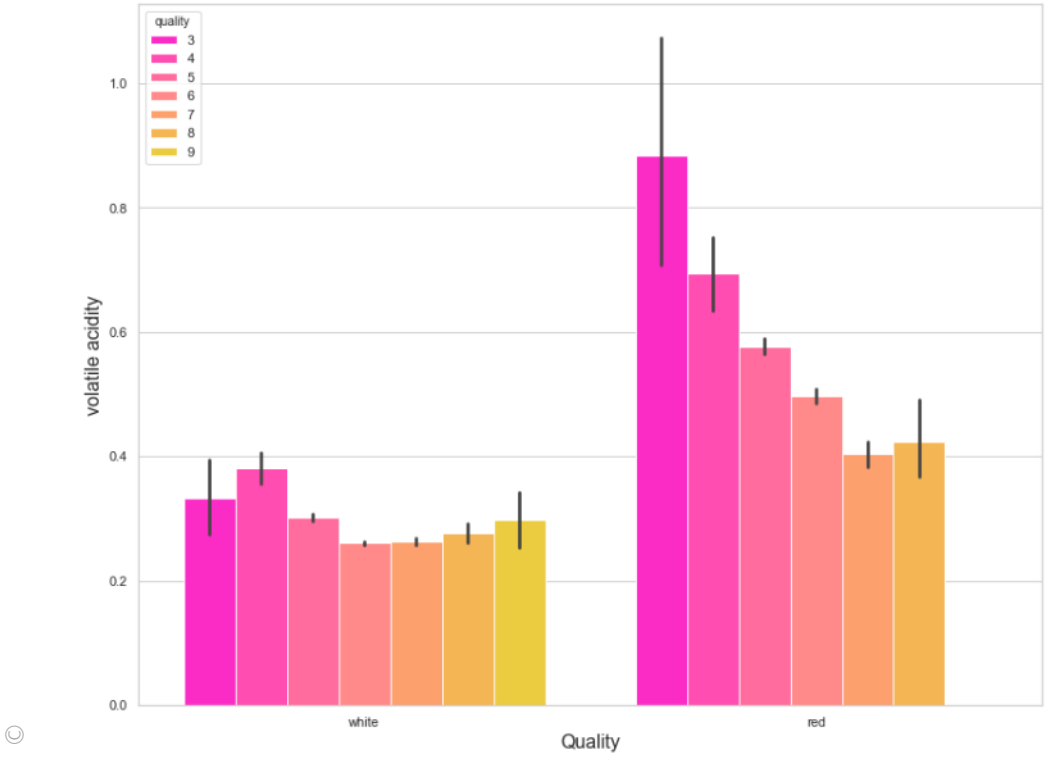


FIGURE 4.5 – Variable acide volatile dans la qualité du Vin

Ces acides doivent être éliminés du vin avant la fin du processus de production. Il est principalement constitué d'acide acétique, bien que d'autres acides tels que les acides lactique, formique et butyrique puissent également être présents. Un excès d'acides volatils est indésirable et conduit à un goût désagréable.La qualité du vin est directement liée à une diminution conséquente de la quantité d'acide volatile car plus la quantité d'acide est basse plus le vin aura un bon goût.

4.2.6.4 Facteurs déterminants à la qualité du vin :

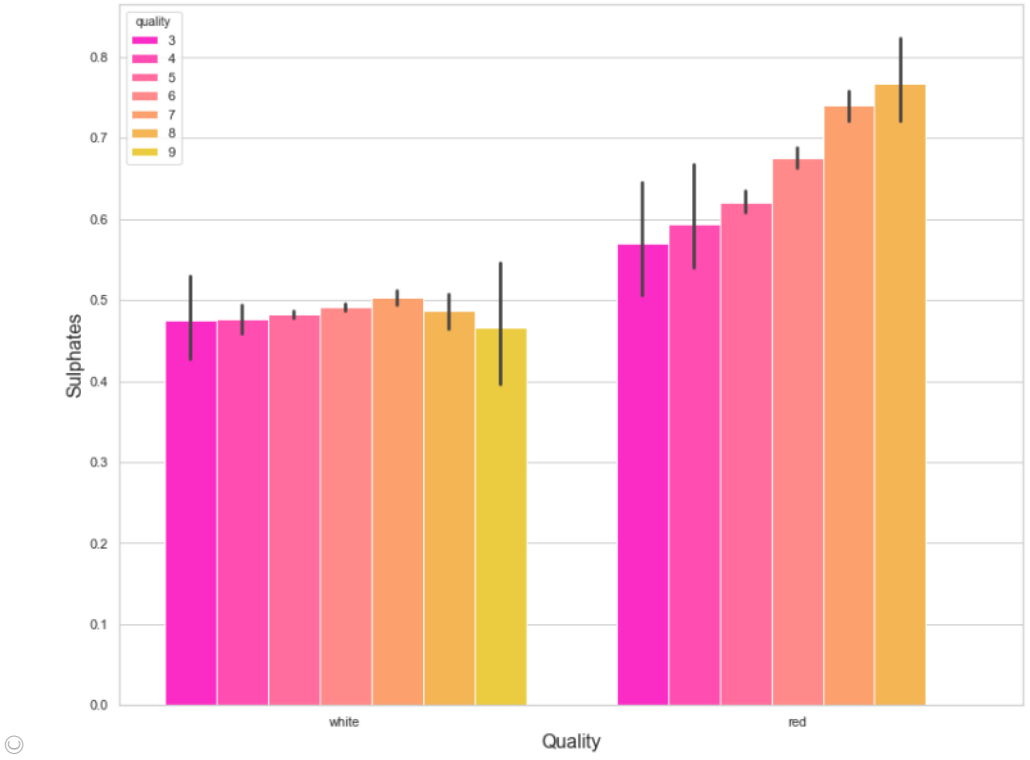


FIGURE 4.6 – Variable sulfates dans la qualité du Vin

Les sulfates dans le vin sont des composés chimiques (dioxyde de soufre ou SO_2) présents naturellement, à des degrés divers, dans tous les types de vin. Le dioxyde de soufre est naturellement présent dans les vins et est un sous-produit de la fermentation, mais la plupart des viticulteurs choisissent d'ajouter un peu supplémentaire pour prévenir la croissance de levures et de microbes indésirables, ainsi que pour se protéger contre l'oxydation.Le dioxyde de soufre inhibe les levures, empêchant les vins doux de se refermenter en bouteille. C'est un antioxydant qui garde le vin frais et exempt d'oxygène.Donc c'est un facteur important pour la qualité du vin car cette dernière est directement liée à la quantité de sulphates dans le vin. Plus de sulphates dans le vin sera meilleur, plus la qualité sera bonne. Cela nous pousse à conclure que les vins moins acides ont besoin de plus de sulphates que les vins plus acides. À **pH** 3,6 et plus, les vins sont beaucoup moins stables et les sulphates sont nécessaires à la conservation. Bien qu'il semble exister une certaine tendance indiquant de légères concentrations de sulfate de higer pour des échantillons de vin de qualité supérieure, la corrélation est assez faible. Cependant, nous voyons que cette tendance est due à une concentration plus élevée sur la qualité moyenne, et nous voyons clairement que les niveaux

de sulfate pour le vin rouge sont beaucoup plus élevés que ceux du vin blanc.Ce qui nous montre l'importance des sulphates dans la qualité des vins.

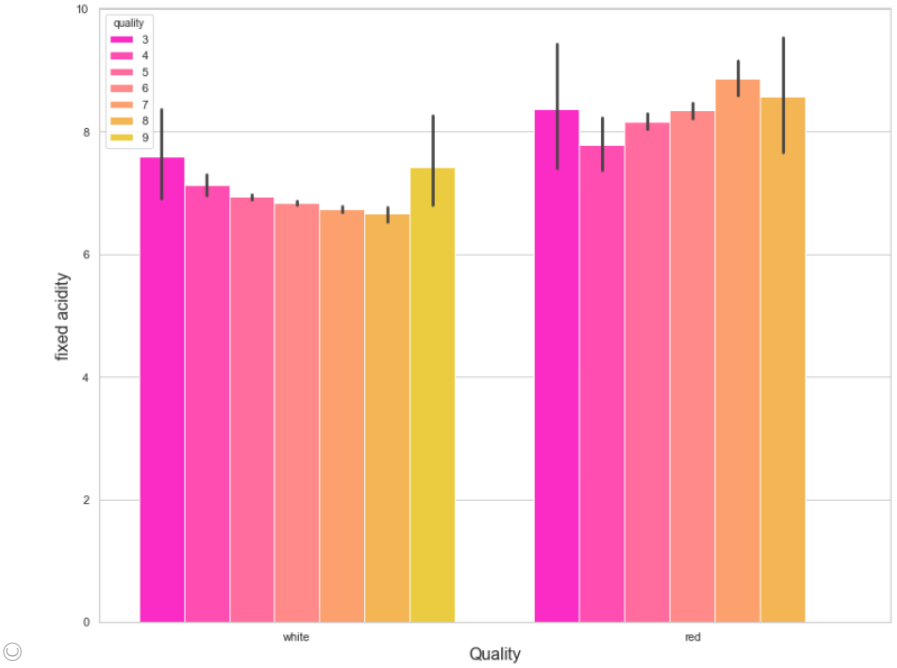


FIGURE 4.7 – Variable fixed acidity dans la qualité du Vin

Il est tout à fait évident que les échantillons de vin rouge ont une acidité supérieure à celle de leurs homologues de vin blanc. Nous pouvons également constater une diminution globale de l'acidité avec un vin de qualité supérieure pour les échantillons de vin rouge, mais pas autant pour les échantillons de vin blanc. C'est l'un des acides fixés qui donne au vin sa fraîcheur. Habituellement, la plus grande partie est consommée pendant le processus de fermentation et parfois, elle est ajoutée séparément pour donner au vin plus de fraîcheur, donc une augmentation de la quantité d'acide fixe donnera au vin une meilleure fraîcheur donc une meilleure qualité.

4.2.6.5 Résultats de la prédiction avec toutes les variables indépendantes(features) :

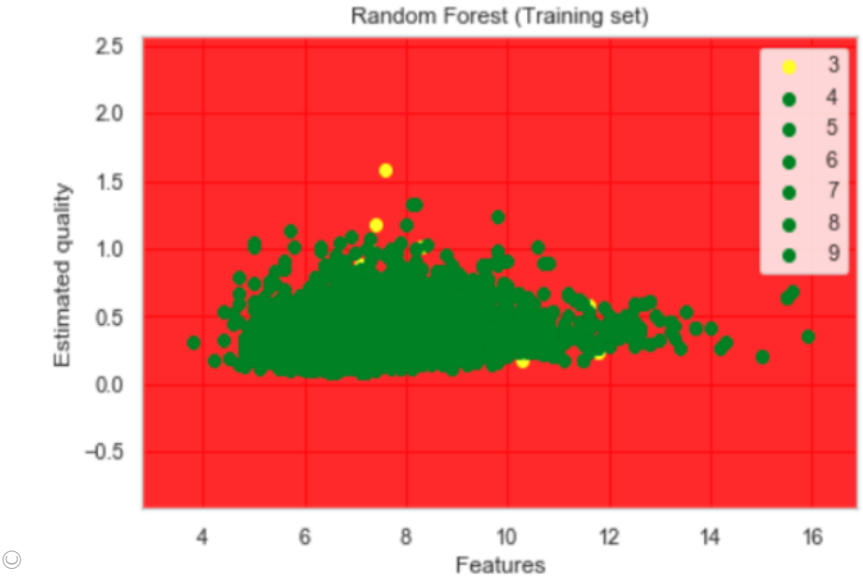


FIGURE 4.8 – Résultat des données d'entraînement

A la suite de l'observation du graphe de prédiction des données d'entraînement (training set) on peut voir que le modèle a mal classée certaines observations qui sont colorées au jaune. Ces observations mal classées peuvent être interprétées par la présence des données aberrantes dans notre jeu des données et la variable dépendante à savoir la qualité du vin . la figure suivante nous montre cette présence des données aberrantes qui sont loin des observations normales donc elles sont considérées comme des erreurs dans notre modèle.

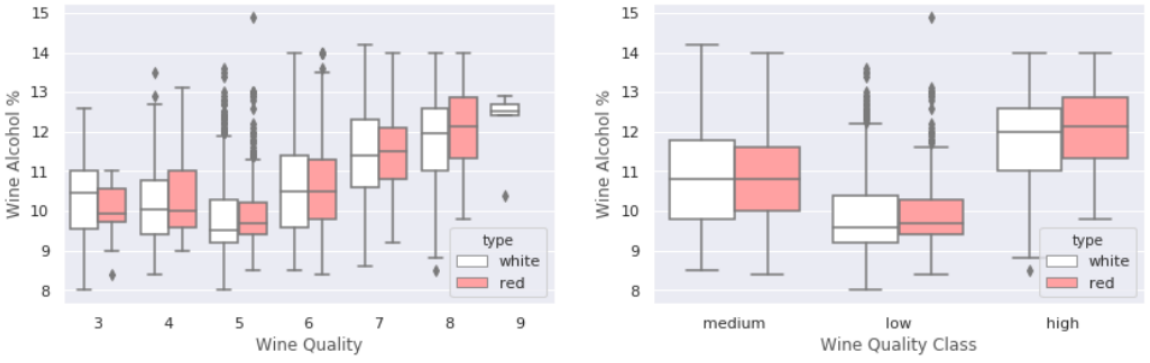


FIGURE 4.9 – Données aberrantes dans la variable Qualité du vin

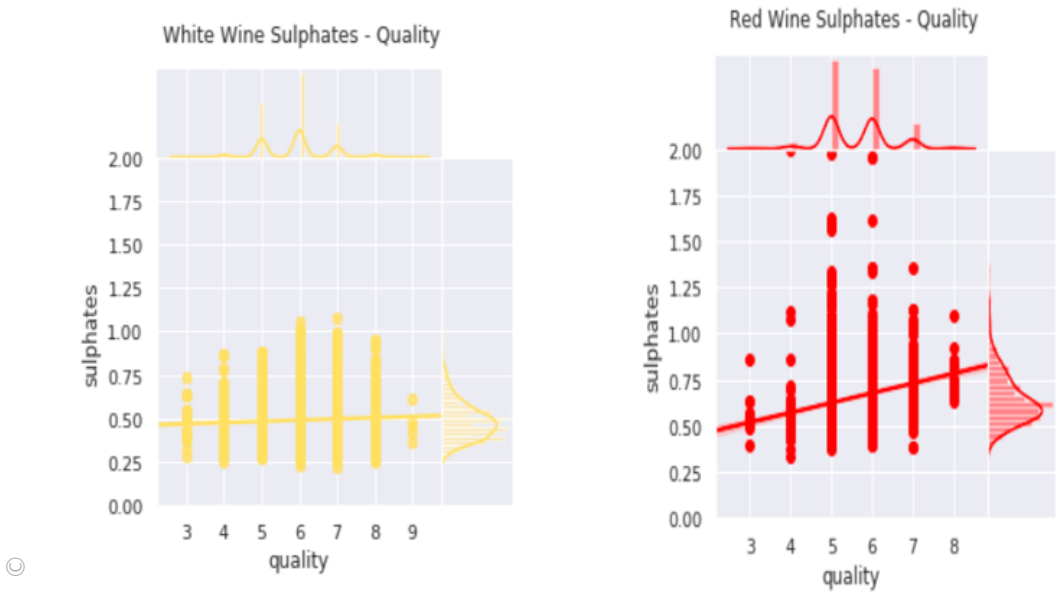


FIGURE 4.10 – Données aberrantes dans Sulfates

On voit une présence importante des valeurs aberrantes au niveau des observations du Vin rouge par rapport au Vin blanc cette présence conséquente des outliers va biaiser notre modèle et celles-ci affecteront le résultat final de notre modèle sur les données de validation(Test set) qui nous permettra d'évaluer en calculant le pourcentage de l'erreur que modèle aura trouvé c'est-à dire la variabilité non expliquée par notre modèle.

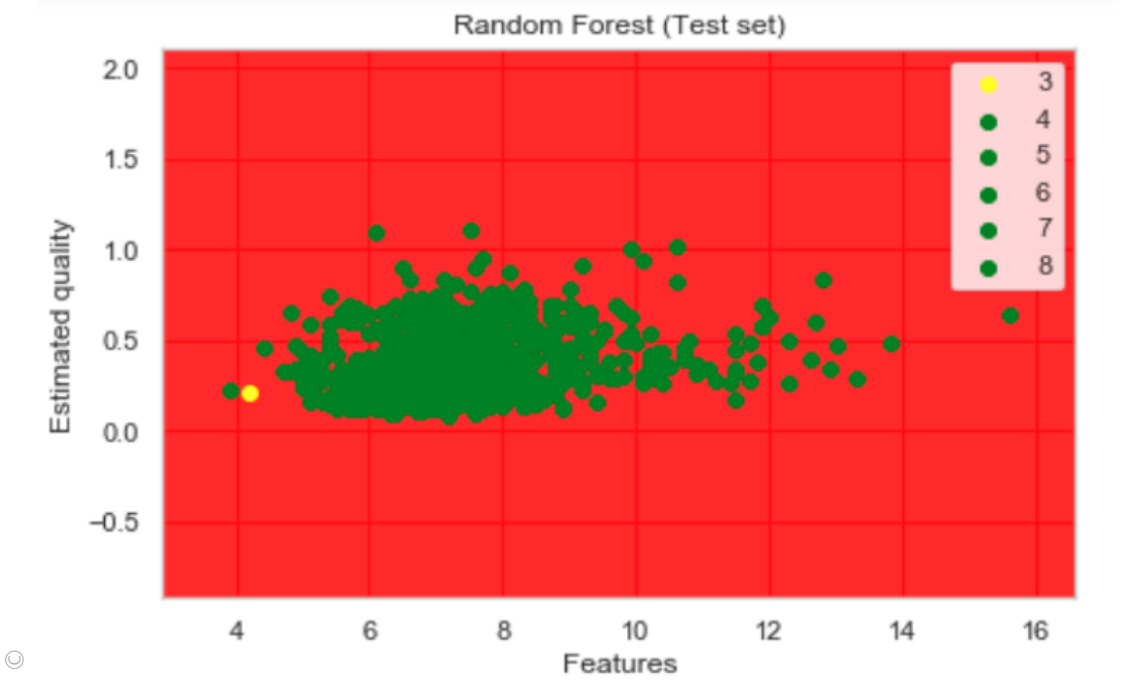


FIGURE 4.11 – Prédiction des données de test

À partir de ses résultats, le modèle obtient une précision de **91,14**% en utilisant toutes les variables de la base de données.Avec une erreur de **mae** de **0.35** degrees d'où une erreur de **0.086** On peut aussi voir ces erreurs sur la matrix de confusion c'est à dire les mal classifications faites par le modèle car sur un total des observations de 1523 Vins classés bons il a classé 174 Vins parmi cette portion comme pas bons et du cote des Vins pas classés bons d'un nombre de 205 il a classé 48 parmi celle-ci comme bons.

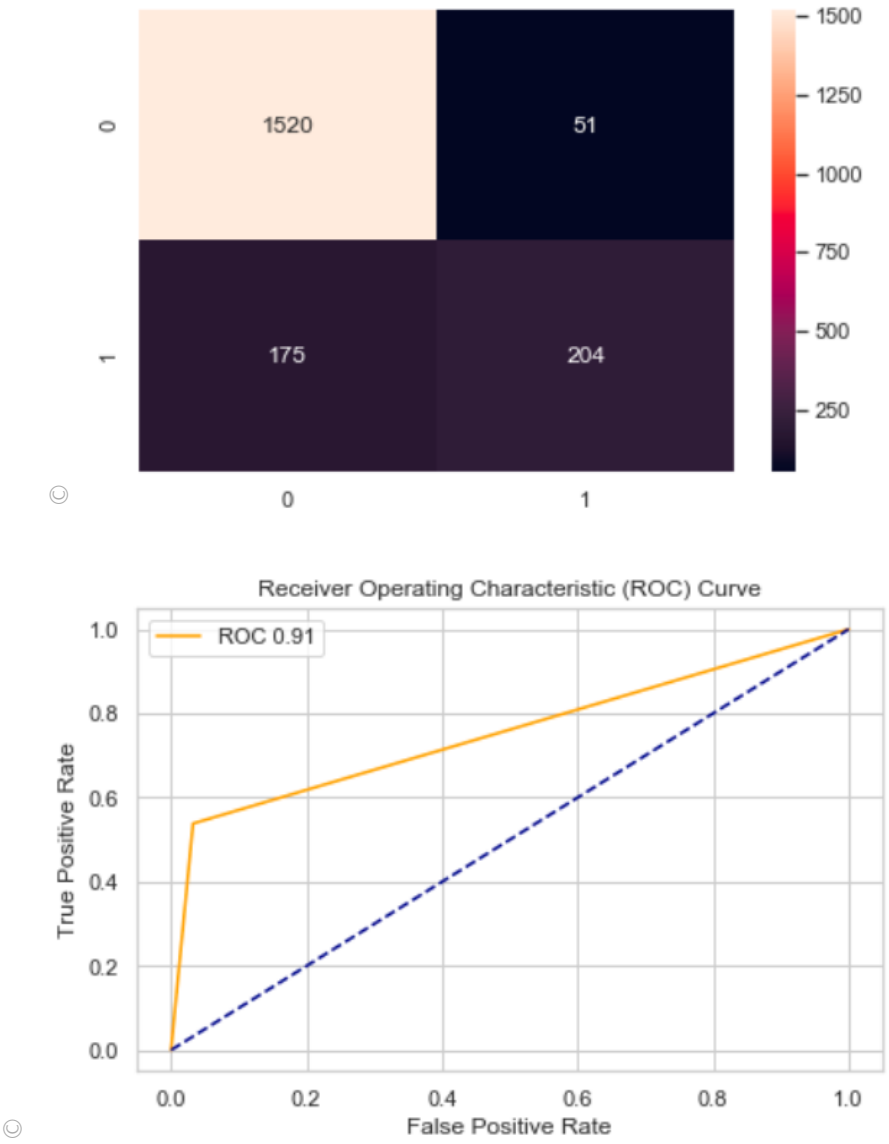


FIGURE 4.12 – Résultats

Cet résultat a été obtenu grâce à la fonction Grid search du package RandomForestClassifier, elle nous permet de trouver les meilleures hyper-paramètres du modèle avec un nombre d'arbres de 500 il y a toujours la possibilité d'améliorer le résultat.

```
params_dict={'n_estimators':[500],'max_features':['auto','sqrt','log2']}
clf_rf=GridSearchCV(estimator=RandomForestClassifier(n_jobs=-1),param_grid=params_dict,scoring='accuracy',cv=10)
clf_rf.fit(X_train,y_train)
```

```
GridSearchCV(cv=10, error_score='raise',
       estimator=RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=None, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=-1,
            oob_score=False, random_state=None, verbose=0,
            warm_start=False),
       fit_params=None, iid=True, n_jobs=1,
       param_grid={'n_estimators': [500], 'max_features': ['auto', 'sqrt', 'log2']},
       pre_dispatch='2*n_jobs', refit=True, return_train_score='warn',
       scoring='accuracy', verbose=0)
```

FIGURE 4.13 – Paramètres de random forest

4.2.6.6 Résultats avec les variables importantes :

Pour sélectionner les variables importantes a notre modèle nous utiliserons la technique de **backward elimination**.
L'importance de la sélection d'entités (variables) est très importante dans notre cas car nous avons à faire avec un jeu de données contenant un grand nombre d'entités. Ce type de jeu de données est souvent appelé un jeu de données de grande dimension. Maintenant, avec cette haute dimensionnalité, il y a beaucoup de problèmes tels que cette dimensionnalité augmentera considérablement le temps de formation de notre modèle d'apprentissage automatique, elle peut aussi rendre notre modèle très compliqué, ce qui peut conduire à un sur ajustement.Car le plus souvent, dans un ensemble de fonctionnalités de grande dimension, il reste plusieurs fonctionnalités qui sont redondantes, ce qui signifie qu'elles ne sont que des extensions des autres fonctionnalités essentielles. Ces fonctionnalités redondantes ne contribuent pas non plus efficacement à la formation du modèle. Il est donc clair qu'il est nécessaire pour nous d'extraire les caractéristiques les plus importantes et les plus pertinentes de notre ensemble de données afin d'obtenir les performances de modélisation prédictives les plus efficaces.

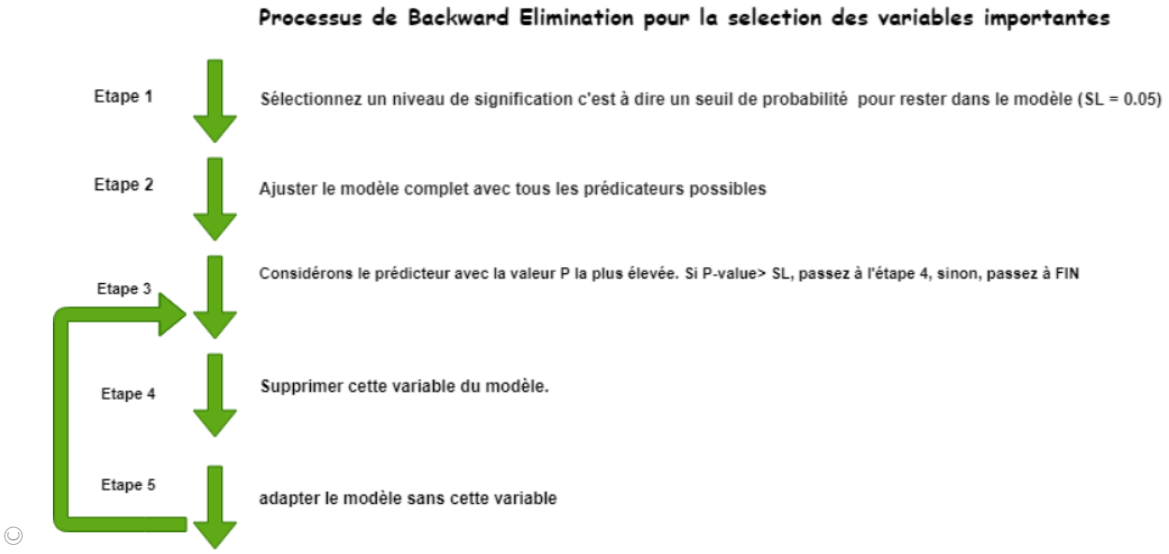


FIGURE 4.14 – Processus de Backward Elimination

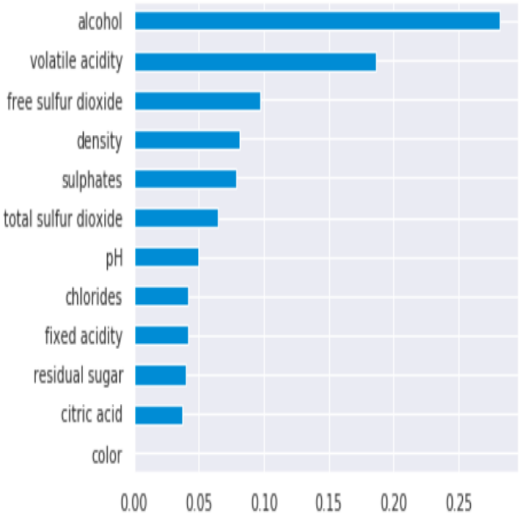


FIGURE 4.15 – Variables selon leurs importances

OLS Regression Results

Dep. Variable:	y	R-squared:	0.192
Model:	OLS	Adj. R-squared:	0.190
Method:	Least Squares	F-statistic:	139.7
Date:	Sat, 03 Aug 2019	Prob (F-statistic):	2.41e-289
Time:	21:14:49	Log-Likelihood:	-2532.3
No. Observations:	6497	AIC:	5089.
Df Residuals:	6485	BIC:	5170.
Df Model:	11		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.1966	0.004	44.299	0.000	0.188	0.205
x1	0.0663	0.010	6.778	0.000	0.047	0.085
x2	-0.0379	0.006	-6.119	0.000	-0.050	-0.026
x3	-0.0005	0.006	-0.084	0.933	-0.011	0.011
x4	0.1099	0.012	9.243	0.000	0.087	0.133
x5	-0.0086	0.006	-1.512	0.131	-0.020	0.003
x6	0.0294	0.006	4.541	0.000	0.017	0.042
x7	-0.0393	0.008	-5.171	0.000	-0.054	-0.024
x8	-0.1344	0.018	-7.623	0.000	-0.169	-0.100
x9	0.0430	0.007	6.114	0.000	0.029	0.057
x10	0.0477	0.006	8.667	0.000	0.037	0.058
x11	0.0949	0.010	9.811	0.000	0.076	0.114

Omnibus:	868.777	Durbin-Watson:	1.601
Prob(Omnibus):	0.000	Jarque-Bera (JB):	1261.359
Skew:	1.071	Prob(JB):	1.26e-274
Kurtosis:	3.260	Cond. No.	9.57

FIGURE 4.16 – Variables avec P-value

Ces deux figures nous donnent un rapport sur l'ensemble des variables constituant notre modèle mais aussi l'importance de chacune de ces variable par rapport à notre variable target (la qualité du vin) qu'on cherche a déterminer mais aussi comment chacune d'elles peut affecter la variabilité expliquée de notre modèle. Donc en tenant du

processus de **Backward Elimination** on sélectionnera les variables ayant un pouvoir d'explication sur notre variable de sortie sur la base d'un seuil de probabilité **(Le niveau de signification SL = 0.05)** tout en tenant compte de la variabilité de notre coefficient de determination ajustée (c'est a dire une variable est enlevé dans le modèle au cas ou sa suppression ne diminue pas la variabilité du modèle de façon significative.

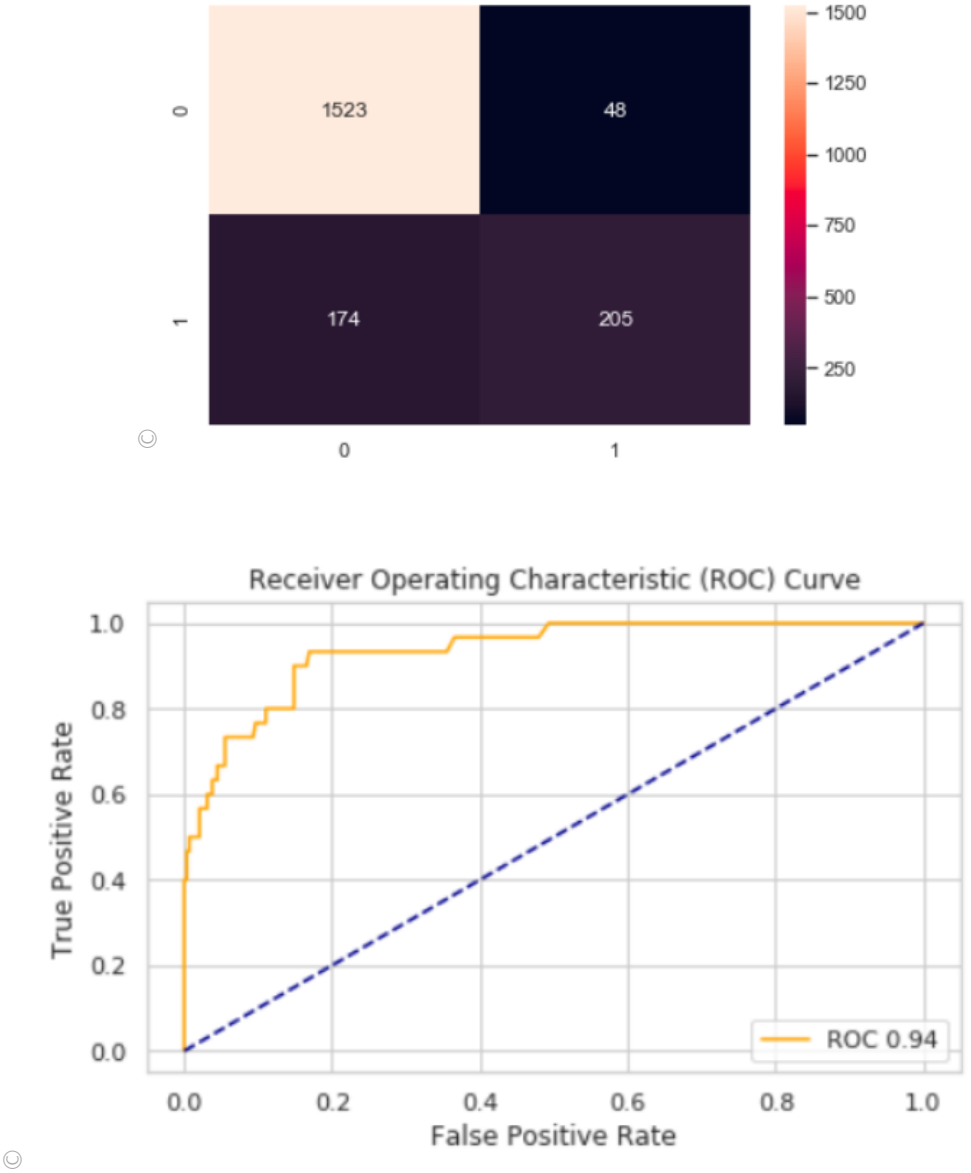


FIGURE 4.17 – Résultats

À partir de ses variables, le modèle obtient une précision de **94,87**% en utilisant

toutes les variables de la base de données.Avec une erreur de **mae** de **0.33** degrees d'où une erreur de **5.13**% avec possibilité d'améliorer les résultats en ajustant les hyper-paramètres du modèle.

4.3 CONCLUSION :

Le but de ce travail est montrer que l'utilisation des algorithmes d'apprentissage de machine learning peut être un plus et bien plus un outil puissant que les économistes doivent utiliser pour résoudre de façon plus efficacement à l'étude des problemes économiques. Au vue des différents travaux réalisés avec les techniques d'apprentissage automatique de Machine Learning pour résoudre les problématiques en économétrie à savoir l'explication d'un phénomène par rapport aux differents facteurs pouvant engendre la réalisation de cet phénomène (trouver une relation pouvant expliquer ou d'estimer à mieux la variable endogène (la variable à expliquer) d'un phénomène par rapport aux variables exogènes) et on voit clairement que les techniques d'apprentissage automatique fournissent des meilleures performances que les méthodes traditionnelles que la science économétrie utilise. Et vue l'avènement de **BIG DATA « Données massives»** la grande capacité des données il serait presque impossible pour les économistes de pouvoir réaliser une meilleure estimation ce qui n'est pas le cas des techniques d'apprentissage automatique car plus la capacité des données est grande plus la prédiction devient meilleure. A l'issu de notre étude nous avions remarqué que notre modèle algorithmique produisait des meilleurs résultats que celui des méthodes traditionnelles, implicitement il résout les trois obstacles de la modélisation économique à savoir : choix des variables explicatives, choix d'une forme fonctionnelle et le problème d'interactions des variables explicatives. Notre analyse souligne également que ces méthodes sont capables de produire des mode les qui peuvent être interprétés, ce qui est crucial du point de vue de l'économiste. Sur cette base, il semble possible de soutenir que ces outils sont très utiles pour les travaux statistiques en économie, en complément d'autres approches standard. Cette affirmation conduit naturellement à considérer l'utilisation en recherche économique des nombreuses extensions de ces algorithmes et d'autres méthodes issues du cadre d'apprentissage machine. Comme j'ai eu mentionné au début de ce travail ces deux domaines ont une similarité car leur but est de prédire une variable endogène (y) à partir d'une variable exogène (x) problème de régression simple ou par plusieurs variables explicatives (problème de régression multiple) ce qui nécessite de dire que les économistes doivent plus considérer les techniques d'apprentissage aux techniques traditionnelles et de plus l'attrait de l'apprentissage automatique est qu'il parvient à découvrir des modèles généralisés en donnant une meilleure formulation du problème en tenant compte de toutes les variables du problème ce qui est très important dans la résolution d'un problème économique.

- Perspectives : Les options pour améliorer et modifier cet algorithme pourrait être essayer en utilisant différents noyaux de mais aussi les paramètres d'optimisation **RANDOM FOREST**.

Bibliographie

[André et al., 2018] André, P., Wooldridge, J., Beine, M., Béreau, S., de la Rupelle, M., Durré, A., Gnabo, J.-Y., Heuchenne, C., Leturcq, M., and Petitjean, M. (2018). *Introduction à l'économétrie : une approche moderne*. De Boeck Supérieur.

[Besse, 2003] Besse, P. (2003). Pratique de la modélisation statistique. *Publications du laboratoire de statistiques et probabilités. Université Paul Sabatier, Toulouse. Disponiblea partir de l'URL http ://www-sv. cict. fr/lsp/Besse*.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine learning*, 45(1) :5–32.

[Charpentier et al., 2017] Charpentier, A., Flachaire, E., and Ly, A. (2017). Econom\'etrie et machine learning. *arXiv preprint arXiv :1708.06992*.

[Crépon, 2005] Crépon, B. (2005). Econométrie linéaire. *INSEE Franc (http ://www. ensae. fr/paristech/SE2C2/Cours_2005_06. pdf)*.

[Guyader, 2011] Guyader, A. (2011). Régression linéaire. *Université Rennes*, 2 :60–61.

[Michie et al., 1994] Michie, D., Spiegelhalter, D. J., Taylor, C., et al. (1994). Machine learning. *Neural and Statistical Classification*, 13.