ROULER COUVERT

Data Mining

Paul Vidal & Gabin Sengel

2024-06-25

Sommaire

Introduction

Dans le cadre de cette étude, nous examinerons une base de données qui compile des informations sur les membres d'une compagnie d'assurance santé indienne. Cette entreprise envisage d'élargir son offre en introduisant des contrats d'assurance automobile destinés à ses clients actuels. Notre objectif est de développer le modèle le plus efficace pour prédire la probabilité qu'un client soit intéressé par un contrat d'assurance automobile.

Description de la base de données

Pour ce faire, nous disposons de la base de données **Health Insurance Cross Sell Prediction** créé par **Anmol Kumar**. Cette base de données regroupe des informations sur **381109 clients** au travers de **11 variables**, elle ne possède pas de données manquantes. Ces variables sont présentées dans le tableau suivant :

Nom de la variable	Type de la variable	Description
Gender	factor	Le genre du client
Age	integer	L'age du client
Driving_License	factor	Vaut 1 si le client a le permis de conduire, 0 sinon
Region_Code	factor	Code de la région du client
Previously_Insured	factor	Vaut 1 si le client a déjà une assurance auto, 0 sinon
Vehicle_Age	factor	L'age du véhicule
Vehicle_Damage	factor	Vaut 1 si sa voiture a déjà été accidentée, 0 sinon
Annual_Premium	numeric	Ce que le client paye pour son assurance sur 1 an (en Roupie)
Policy_Sales_Channel	factor	Code qui indique le canal de communication du client (ex:Mail, Telephone, en personne etc...)
Vintage	numeric	Nombre de jours depuis la souscription de son premier contrat
Response	factor	Vaut 1 si le client est interessé pour prendre une assurance auto, 0 sinon

Problèmes et solution

L'un des premiers problèmes que nous avons rencontrés est le nombre d'observations dans la base de données. Avoir beaucoup d'information peut être un avantage, mais l'estimation de certains modèles peut prendre un certain temps et être très gourmand en puissance de calcul.

Pour économiser du temps de calcul tout en restant performant, nous avons eu comme première idée de faire un échantillonnage de **100000** individus en gardant la même proportions de **Response = 1** que dans la base de donnée d'origine, c'est à dire **12.26 %**.

Mais nous nous sommes vite confronté a un déséquilibre de classe, nos modèles avaient tendances à favoriser la classe majoritaire (dans notre cas, prédire majoritairement des 0)

Pour pallier a ce problème deux solutions se proposait à nous:

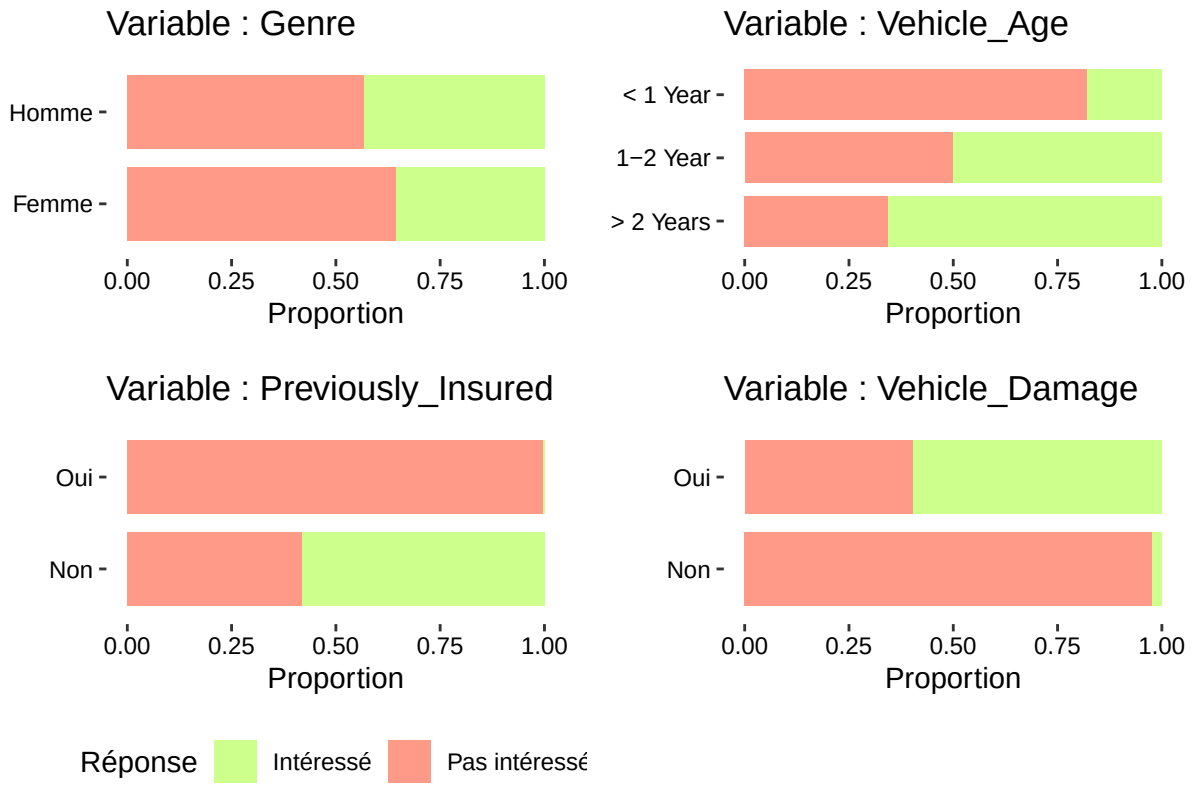
- Ajuster le seuil de classification, qui de est fixé à 0.5 par défaut. En le diminuant nous "forçons" les modèles à prédire plus de **Response = 1**. Mais cette méthode n'est pas toujours efficace et peut être très arbitraire.
- Augmenter la proportions de **Response = 1** dans notre échantillon pour que cette classe soit moins sous représenté.

Etant donné que nous disposons d'un nombre conséquent d'observations, nous avons alors choisi la deuxième option. Nous avons alors gardé l'idée de prendre un échantillon de **381109** individus, mais cette fois-ci avec une proportion de **Response = 1** de **40 %**

Analyse descriptive

Avant de procéder à l'estimation des modèles, nous souhaitons introduire une section consacrée aux statistiques descriptive, portant sur les variables quantitatives et certaines variables qualitatives.

Variables qualitatives



Nous pouvons observer plusieurs tendances parmi les clients de la compagnie d'assurance :

- La proportion de clients intéressés est un peu plus élevée chez les hommes que chez les femmes, bien que les chiffres soient assez proches (43.36 % pour les hommes contre 35.69 % pour les femmes).
- Il semble que plus le véhicule du client est ancien, plus il y a de chance pour qu'il soit intéressé par l'assurance (17.87 % pour les véhicules de moins d'un an, contre 65.62 % pour ceux de plus de deux ans).
- Les clients qui ont déjà une assurance véhicule ont tendance à ne pas être intéressés par cette nouvelle offre (99.57 %).
- Par ailleurs, une plus grande proportion de clients ayant déjà subi un sinistre sont intéressés par l'assurance (59.75 %), contrairement à ceux n'ayant pas eu de sinistre (2.42 %).

Variables quantitatives

Les variables quantitatives de notre de base de données sont l'Age (en années), Annual_premium (montant de la prime payé chaque année en roupie) et Vintage (ancienneté de l'assuré en jours).

	Age	Annual_Premium	Vintage
Minimum	20	2630	10
Maximum	85	540165	299
Moyenne	40	30951	154
Médiane	40	32062	154
Ecart_type	15	17887	84

Les statistiques descriptives des variables quantitatives montrent que l'âge des individus varie entre 20 et 85 ans avec une moyenne et une médiane similaire suggérant une distribution symétrique de l'âge parmi les individus.

La prime moyenne payé fluctue énormément allant de 2 630 roupies à 540 165, il y a une grande disparité dans les montants payés par les différents clients. La moyenne vaut 30 951 roupies et la médiane est légèrement supérieur à 32 000 roupies.

L'ancienneté, exprimée en jours, varie de 10 à 299 exprimant une variabilité relativement élevé. On note cependant que tous les clients de notre base de données ont souscrit leur contrat depuis moins d'un an. Ainsi, tous les clients sont récents. La moyenne et la médiane valent toute deux 154 jours.

Les écarts-types des variables quantitatives indiquent une dispersions relativement modéré autour de la moyenne.

Pour avoir une première intuition des variables influentes, on décide de vérifier si les moyennes de chaque variable sont significativement différentes entre les individus intéressés à souscrire un contrat auto et les autres.

Variable	μ_NI	μ_I	IC inf à 95%	IC sup à 95%	p-value
AGE	38.18	43.47	-5.46	-5.12	0.00
A_PREMIUM	30513.44	31607.02	-1323.13	-864.03	0.00
VINTAGE	153.59	154.18	-1.65	0.47	0.27

L'âge moyen des individus non intéressés est inférieur (38,18) à celui des individus intéressés, qui est de 43,47. L'intervalle de confiance à 95 % n'inclut pas 0, et la p-value est proche de 0,00, indiquant une différence statistiquement significative entre les âges des individus intéressés et non intéressés.

Les primes annuelles moyennes sont également différentes entre les groupes, les individus non intéressés payant légèrement moins en moyenne que ceux intéressés. Les IC pour cette comparaison n'incluent pas 0, et la p-value est proche de 0,00, montrant une différence très significative dans les primes entre les groupes.

Les moyennes pour les individus non intéressés et intéressés sont très proches (153,59 contre 154,18), et les IC incluent 0. La p-value est de 0,27, ce qui n'est pas statistiquement significatif, suggérant que l'ancienneté ne diffère pas significativement entre ceux intéressés et ceux non intéressés par l'assurance véhicule. Un résultat peu étonnant quand on sait que tout les clients sont très récent (inférieur à un an).

Hypothèses :

- L'âge pourrait être un facteur dans l'intérêt pour l'assurance véhicule, les individus plus âgés montrant plus d'intérêt.
- Des primes annuelles plus élevées pourraient être associées à une probabilité plus élevée d'intérêt pour des produits d'assurance supplémentaires.
- L'ancienneté d'un client peut ne pas être un bon prédicteur de leur intérêt pour souscrire à une assurance véhicule.

Présentation des modèles

Découpage train/test:

Nous allons à présent aborder la présentation et l'analyse des modèles que nous avons pu entraîner et tester. Pour cela, nous avons divisé notre échantillon en deux parties : un ensemble d'entraînement, df_train, et un ensemble de test, df_test. Nous avons respecté la répartition classique, attribuant les deux tiers de l'échantillon à l'entraînement 66666 observations) et le tiers restant au test 33334 observations).

Pour chaque modèle, nous allons d'abord présenter la matrice de confusion ainsi que les mesures de performances et ensuite nous commenterons les résultats

Il existe plusieurs indicateurs qui nous permettent de déterminer si un modèle est performant ou non, en voici quelques un :

$$\text{Sensibilité (Recall)} = \frac{TP}{TP + FN}$$

$$\text{Spécificité} = \frac{TN}{TN + FP}$$

$$\text{Précision} = \frac{TP}{TP + FP}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Pour affiner notre cible, nous devons porter une attention particulière à notre objectif de prédiction.

Dans le cadre de notre étude, nous cherchons à identifier avec la plus grande exactitude les clients susceptibles de souscrire au nouveau contrat auto proposé par la compagnie d'assurance.

L'accent est donc mis sur une juste prévision des vrais positifs – autrement dit, nous aspirons à concevoir un modèle caractérisé par une forte sensibilité, minimisant ainsi le risque d'erreurs de type II.

Cependant, se fier exclusivement à la sensibilité pourrait conduire au choix d'un modèle qui prédit systématiquement une réponse positive.

Il est donc essentiel de considérer également la précision, un indicateur qui évalue la proportion de prédictions positives qui sont effectivement correctes.

La spécificité, en revanche, sera reléguée au second plan, partant du principe qu'un faux positif – soit la proposition d'un contrat à un client non intéressé – n'engendre pas de conséquences désastreuses pour l'entreprise, du moins pas dans une mesure significative.

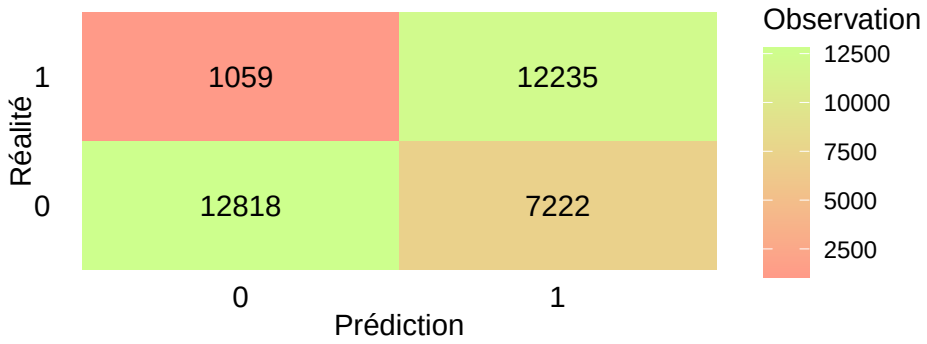
En revanche, manquer de reconnaître un client potentiellement intéressé se traduirait par une occasion manquée, affectant directement le potentiel de croissance de la compagnie.

Bien que la sensibilité et la précision soient nos critères principaux, l'accuracy reste un indicateur que nous surveillerons attentivement, car il reflète la proportion globale de prédictions correctes du modèle, tant pour les classes positives que négatives.

LDA

La LDA, ou Analyse Discriminante Linéaire, est une technique statistique utilisée pour la classification et la réduction de dimensionnalité. Elle vise à séparer les différentes classes (ou groupes) en trouvant une combinaison linéaire des caractéristiques qui maximise la séparation entre les classes tout en minimisant la variation au sein de chaque classe. La LDA se distingue par sa facilité de compréhension et de mise en œuvre, faisant d'elle un modèle de base particulièrement efficient. Peu gourmande en ressources de calcul, elle présente l'avantage de ne pas nécessiter une optimisation préalable des paramètres.

Matrice de confusion



Mesure des performances

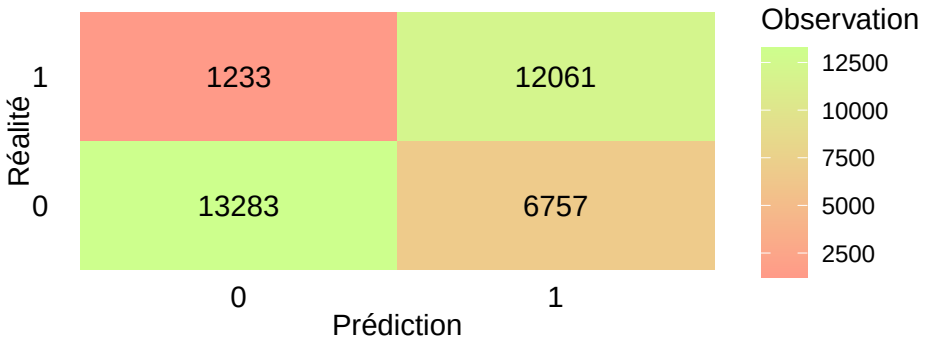
Indicateur	Accuracy	Erreur global	Spécificité	Erreur Type 1	Sensibilité	Erreur Type 2	Précision
Valeur	75.16 %	24.84 %	63.96 %	36.04 %	92.03 %	7.97 %	62.88 %

Le modèle affiche une sensibilité élevée à 92.03 %, indiquant une forte capacité à détecter les vrais positifs. L'accuracy, qui mesure la justesse globale des prédictions, est raisonnable à 75.16 %. La précision s'établit à 62.88 %, suggérant que lorsque le modèle prédit un résultat positif, il est correct un peu plus de 6 fois sur 10. Collectivement, ces indicateurs montrent que le modèle est plutôt efficace pour identifier les cas positifs, tout en maintenant un niveau acceptable de prédictions correctes globales.

QDA

La QDA, ou Analyse Discriminante Quadratique, est une méthode de classification statistique qui, contrairement à la LDA, prend en compte la covariance propre à chaque classe et utilise des séparateurs quadratiques plutôt que linéaires. Cela permet à la QDA de mieux s'adapter aux structures de données plus complexes où la relation entre les variables n'est pas nécessairement linéaire. LA QDA présente les même avantages que la LDA en terme de facilité de compréhension et de temps de calcul.

Matrice de confusion



Mesure des performances

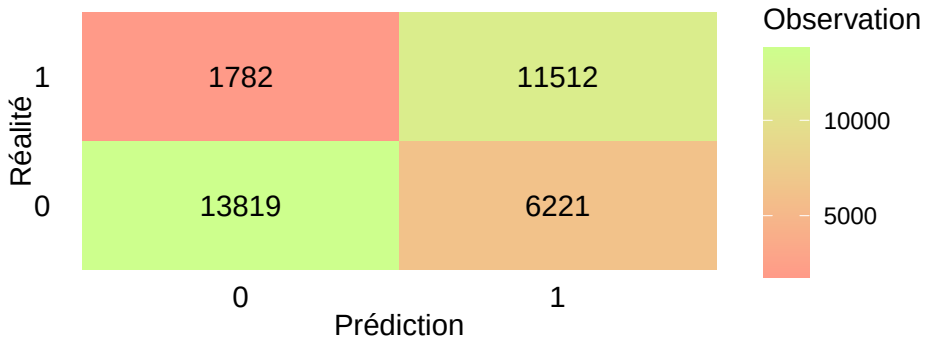
Indicateur	Accuracy	Erreur global	Spécificité	Erreur Type 1	Sensibilité	Erreur Type 2	Précision
Valeur	76.03 %	23.97 %	66.28 %	33.72 %	90.73 %	9.27 %	64.09 %

Ce modèle présente une accuracy légèrement améliorée à 76.03 %, comparé à 75.16 % précédemment, suggérant une légère augmentation de la justesse globale des prédictions. La sensibilité diminue un peu, passant de 92.03 % à 90.73 %, ce qui indique une petite réduction dans la capacité du modèle à identifier tous les vrais positifs. La précision augmente de 62.88 % à 64.09 %, reflétant une meilleure exactitude des prédictions positives. En résumé, ce modèle montre une amélioration de l'accuracy et de la précision, tout en maintenant une sensibilité élevée, ce qui peut indiquer un équilibre légèrement meilleur entre les différents types d'erreurs par rapport au modèle précédent.

Logit

Le modèle logistique, ou Logit, est une technique de régression utilisée pour prédire la probabilité d'appartenance à une catégorie ou classe binaire en fonction de l'une ou plusieurs variables indépendantes. Il modélise la relation entre les variables indépendantes et la log-odds de la variable dépendante, fournissant des coefficients qui représentent le changement logarithmique dans les odds pour une unité de changement dans les variables prédictives. Etant donné que la variable que nous chercons à prédire est binaire, nous avons trouvé pertinent d'estimer ce modèle dans notre étude.

Matrice de confusion



Mesure des performances

Indicateur	Accuracy	Erreur global	Spécificité	Erreur Type 1	Sensibilité	Erreur Type 2	Précision
Valeur	75.99 %	24.01 %	68.96 %	31.04 %	86.6 %	13.4 %	64.92 %

Pour ce modèle, l'accuracy est à 75.99 %, se situant de façon comparable aux deux modèles précédents. La sensibilité est de 86.6 %, ce qui est légèrement inférieur aux résultats antérieurs. La précision est de 64.92 %, montrant une légère amélioration par rapport au premier modèle mais pas par rapport au deuxième.

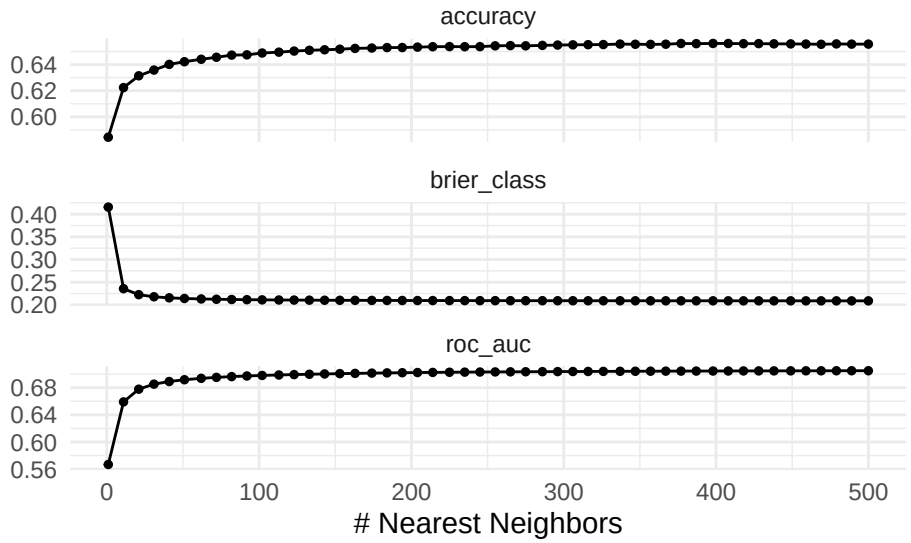
Ainsi, tandis que la sensibilité est un peu réduite, la précision est légèrement meilleure dans ce modèle comparativement au premier, ce qui peut indiquer un équilibre différent entre les types d'erreurs.

KNN

Le modèle KNN, ou k-Nearest Neighbors (k-plus proches voisins), est une méthode de classification qui assigne une classe à une observation en se basant sur les classes des k observations les plus proches dans l'espace des caractéristiques. Il s'agit d'une méthode intuitive, basée sur la similarité des caractéristiques. Avant d'appliquer le modèle KNN, il est crucial de déterminer la valeur optimale de k, c'est-à-dire le nombre de voisins les plus proches à considérer. Cette valeur sera choisie de manière à optimiser l'accuracy du modèle, assurant ainsi les meilleures prédictions possibles.

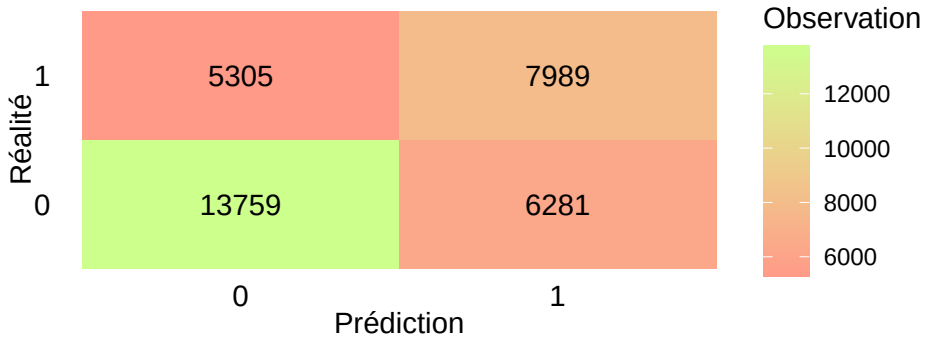
Optimisation des paramètres

Nous avons utilisé une technique de validation croisée à 5 plis pour identifier les paramètres optimaux. La validation croisée est une méthode d'évaluation et de comparaison des modèles statistiques qui divise les données en un nombre prédéfini de 'plis' ou sous-ensembles. Le modèle est alors entraîné sur k-1 plis et testé sur le pli restant, et ce processus est répété k fois, avec chaque pli utilisé une fois comme donnée de test. Ici, k vaut 5, signifiant que nous avons divisé nos données en 5 sous-ensembles distincts.



Nous choisirons k = 398 - plus proches voisins

Matrice de confusion



Mesure des performances

Indicateur	Accuracy	Erreur global	Spécificité	Erreur Type 1	Sensibilité	Erreur Type 2	Précision
Valeur	65.24 %	34.76 %	68.66 %	31.34 %	60.09 %	39.91 %	55.98 %

Pour le modèle KNN, l'accuracy s'établit à 65.24 %, ce qui est inférieur aux performances observées avec les modèles précédents. La sensibilité, à 60.09 %, suggère également une baisse dans la capacité du modèle à détecter tous les vrais positifs. Quant à la précision, elle est de 55.98 %, ce qui est moins élevé que ce que nous avons vu avec les autres modèles.

En somme, comparé aux modèles LDA, QDA et Logit précédemment évalués, le modèle KNN semble présenter des performances moindres en termes d'accuracy, de sensibilité et de précision.

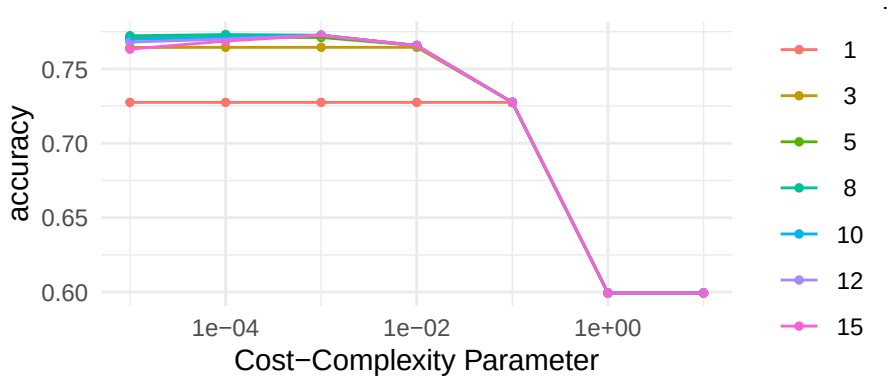
Decision tree

Le modèle Decision Tree est un algorithme qui sépare les données en branches pour aboutir à des décisions sous forme d'arbres. Partant d'un nœud racine, l'algorithme choisit la meilleure caractéristique pour diviser les données en sous-groupes plus homogènes. Cette opération est répétée récursivement pour chaque branche, formant un arbre jusqu'à ce que les feuilles (les nœuds terminaux) correspondent aux classes de classification.

Parmi les paramètres que nous avons ajustés, il y a la complexité de coût (cost complexity). Ce paramètre pénalise la complexité de l'arbre pour prévenir le surajustement : plus la valeur de la complexité de coût est élevée, plus l'arbre de décision sera simple.

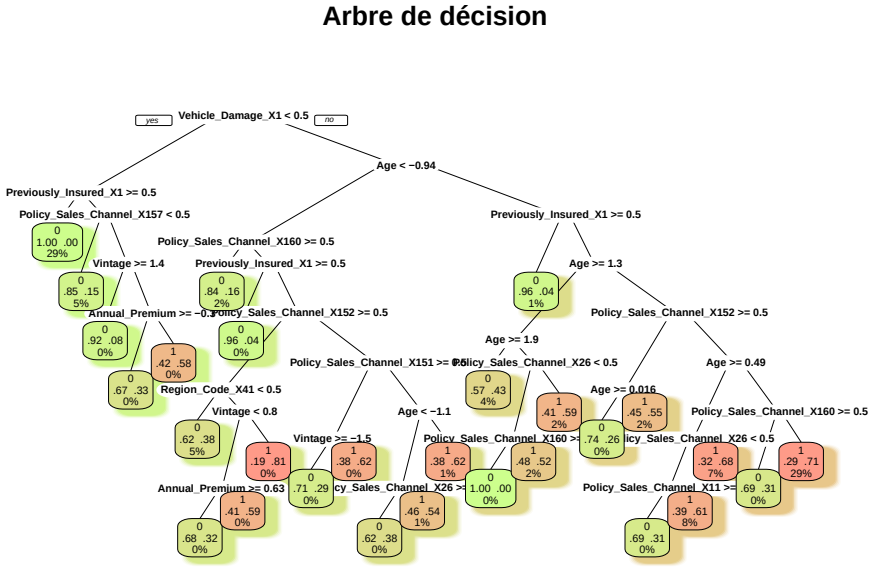
L'autre paramètre est la profondeur maximale de l'arbre, qui détermine combien de divisions (ou 'niveaux') l'arbre peut faire avant de s'arrêter. Limiter la profondeur peut aussi aider à éviter le surajustement en réduisant la complexité du modèle. Dans notre analyse, nous avons choisi la complexité de coût et la profondeur d'arbre qui maximisent l'accuracy, c'est-à-dire la proportion de prédictions correctes faites par le modèle sur les données de validation croisée.

Optimisation des paramètres

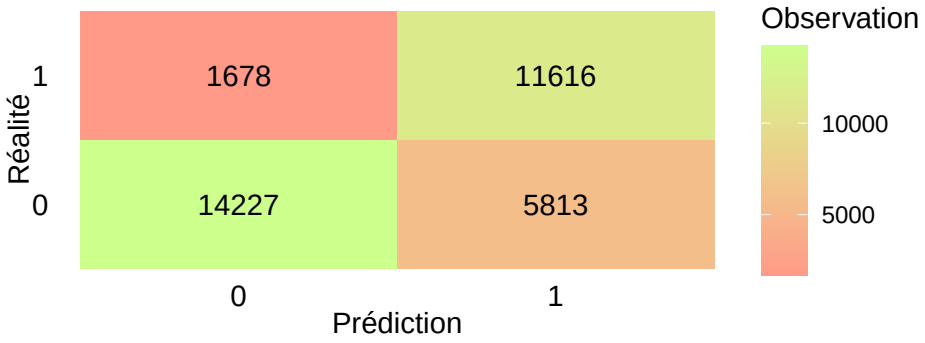


L'arbre obtenu à un coût de compléxité de 10^{-4} et une profondeur de 8.

Arbre obtenu



Matrice de confusion



Mesure des performances

Indicateur	Accuracy	Erreur global	Spécificité	Erreur Type 1	Sensibilité	Erreur Type 2	Précision
Valeur	77.53 %	22.47 %	70.99 %	29.01 %	87.38 %	12.62 %	66.65 %

Dans environ 77.53 % des cas, l'arbre de décision a correctement prédit si un client était intéressé ou non par l'assurance véhicule donc se trompe dans 22.47 % des cas. La performance global du modèle est moyenne. Cependant, La sensibilité est élevée, à 87.38 %. La précision est de 66.65, la meilleur précision pour l'instant parmi les modèles étudiés précédemment. Cela montre que l'arbre de décision est bon pour identifier les clients qui sont effectivement intéressés par une assurance véhicule. L'enjeux principal pour la compagnie d'assurance est de ne pas louper les clients potentiellement intéressé par un contrat auto. Proposer une offre a un client non intéresser comporte moins de risques.

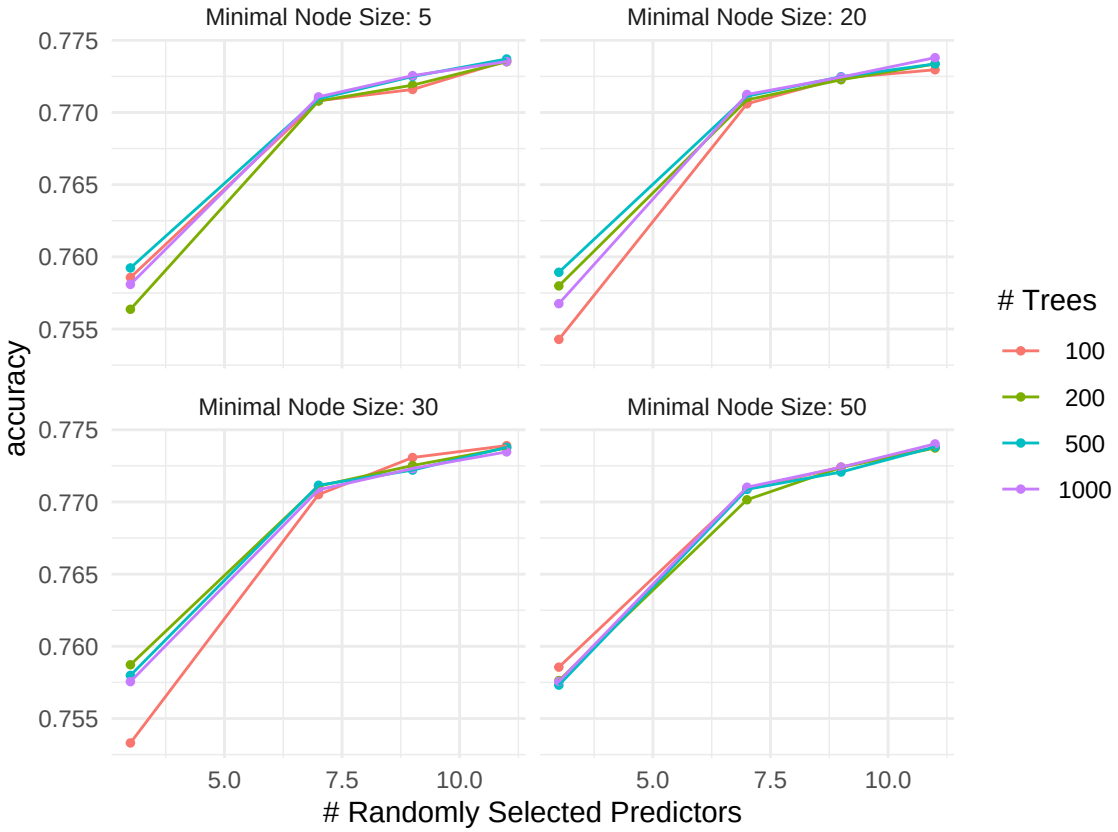
Random Forest

Le principe derrière une forêt aléatoire est qu'un ensemble d'arbre de décision peuvent ensemble former un modèle robuste et puissant. Les forêts aléatoires ajoutent de l'aléatoire à la construction de chaque arbre, ce qui les rend moins sujettes au surajustement. Pendant la construction d'un arbre, au lieu de chercher le meilleur critère de séparation parmi toutes les caractéristiques, l'algorithme recherche le meilleur critère parmi un sous-ensemble aléatoire des caractéristiques. Cette méthode est appelée "bagging" (Bootstrap Aggregating), et elle est combinée avec une sélection aléatoire de caractéristiques pour améliorer la robustesse de la forêt. Pour notre modèle de forêt aléatoire, nous avons commencé par optimiser plusieurs paramètres importants, en utilisant une fois de plus la validation croisée à 5 plis pour évaluer la performance du modèle avec différents paramètres. Cela nous aide à généraliser mieux et à éviter le surajustement.

Optimisation des paramètres

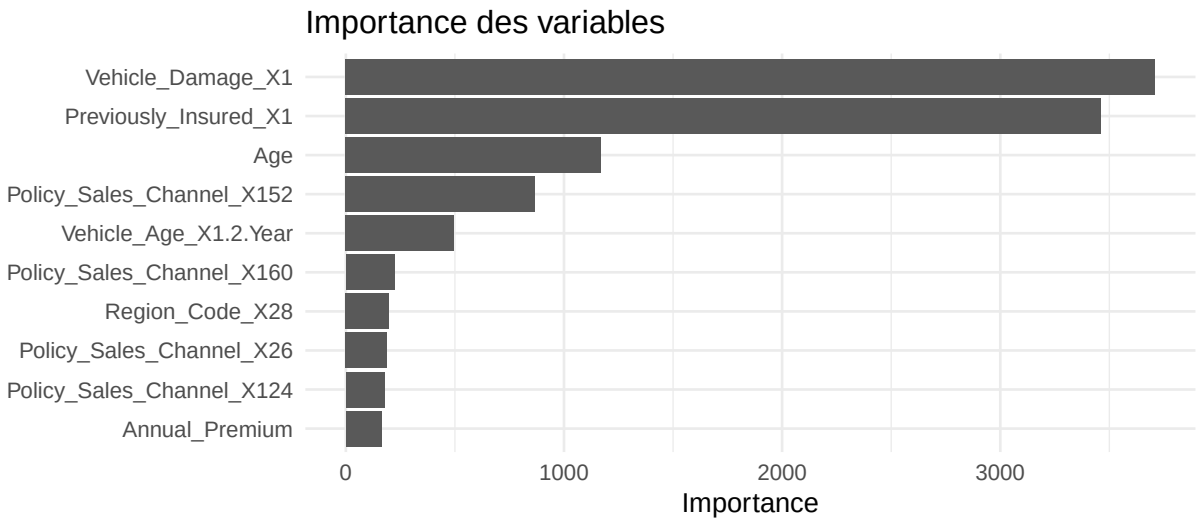
Les paramètres que nous avons optimisés sont :

1. **mtry**: Le nombre de variables à considérer pour la séparation à chaque nœud. Un mtry élevé augmente la chance que les arbres soient similaire et peut conduire à un surajustement, tandis qu'un mtry trop faible peut conduire à des arbres faibles et donc à une forêt moins puissante.
2. **ntrees**: Le nombre d'arbres dans la forêt. Plus il y a d'arbres, plus les prédictions seront stables et précises, mais après un certain point, des gains supplémentaires seront marginaux, et cela augmentera les coûts de calcul.
3. **min_n (Minimal Node Size)**: La taille minimale des nœuds terminaux. Une taille de nœud minimal plus grande peut lisser le modèle (moins de surajustement), tandis qu'une taille plus petite peut capturer davantage de détails dans les données, mais au risque de surajustement.



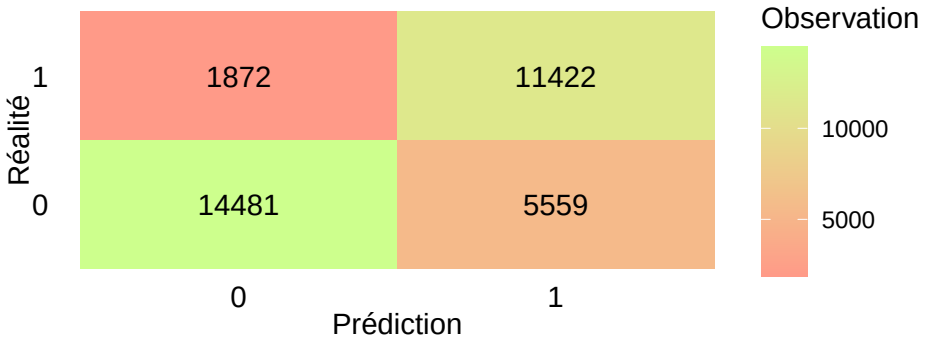
Meilleurs hyperparamètres : **mtry** = 11 et **trees** = 1000 et **min_n** = 50

Importance des variables



L'importance des variables indique les variables ayant le plus d'impact sur les prédictions du modèle. Les deux variables les plus importantes sont, sans grande surprise, Previsously_insured_X1, qui indique que le client a déjà été assuré et Vehicle_damage_X1 qui indique quel le véhicule du client a déjà été accidenté.

Matrice de confusion



Mesure des performances

Indicateur	Accuracy	Erreur global	Spécificité	Erreur Type 1	Sensibilité	Erreur Type 2	Précision
Valeur	77.71 %	22.29 %	72.26 %	27.74 %	85.92 %	14.08 %	67.26 %

La forêt aléatoire a prédit correctement l'intérêt des clients pour l'assurance véhicule dans environ 77.71 % des cas. Comparativement à l'arbre de décision (77.53 %), l'exactitude est très faiblement améliorée. La sensibilité est de 85.92 %. Bien que cette valeur soit un peu plus basse que celle de l'arbre de décision (87.38 %), elle reste élevée. La précision est de 67.26. La random forest est le modèle qui nous donne la meilleur précision. Cela indique que la forêt aléatoire est également un bon modèle pour identifier les clients réellement intéressés par l'assurance véhicule, essentiel pour la compagnie d'assurance. Notre Random forest n'est pas vraiment meilleur qu'un simple arbre de décision.

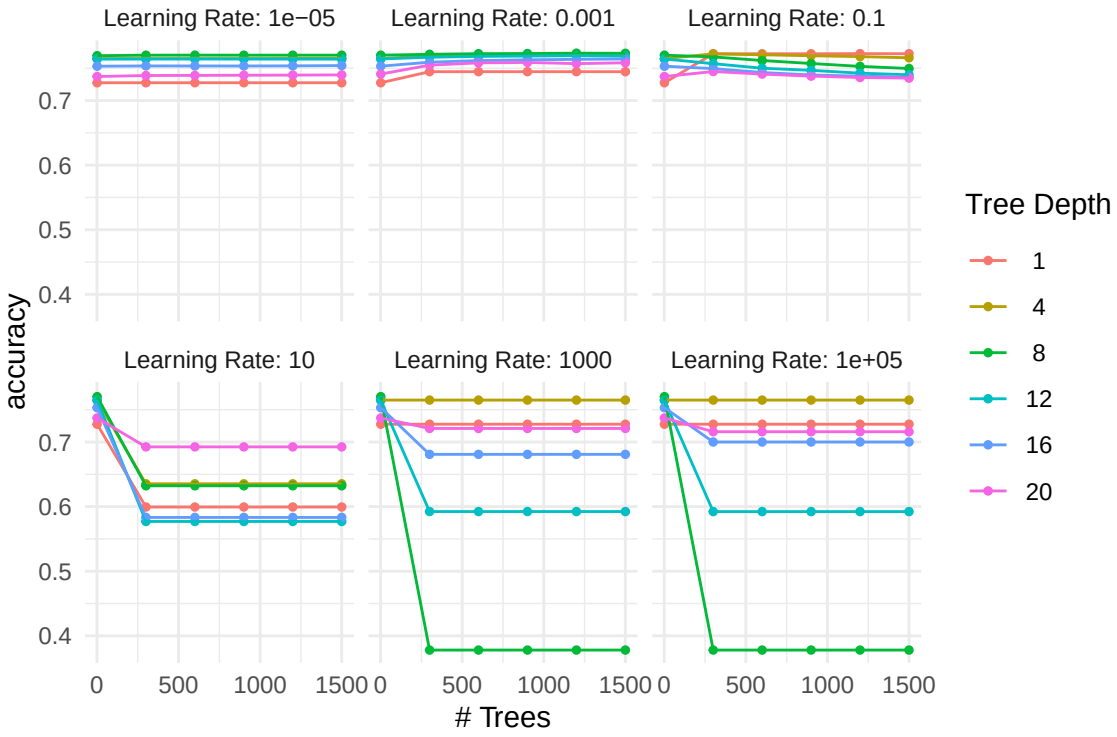
Boosting

Le Boosting est une méthode d'ensemble qui combine les prédictions de plusieurs modèles de base, souvent des arbres de décision, pour améliorer la robustesse et la précision des prédictions. Le principe est d'entraîner plusieurs modèles peu robustes les uns après les autres, en essayant à chaque fois de corriger les erreurs du modèle précédent, pour à la fin obtenir un seul et même modèle robuste. Il s'agit de l'un des modèles les plus récent et efficace, mais il a tendance à être gourmand en temps de calcul, notamment lors de l'optimisation de ses paramètres.

Optimisation des paramètres

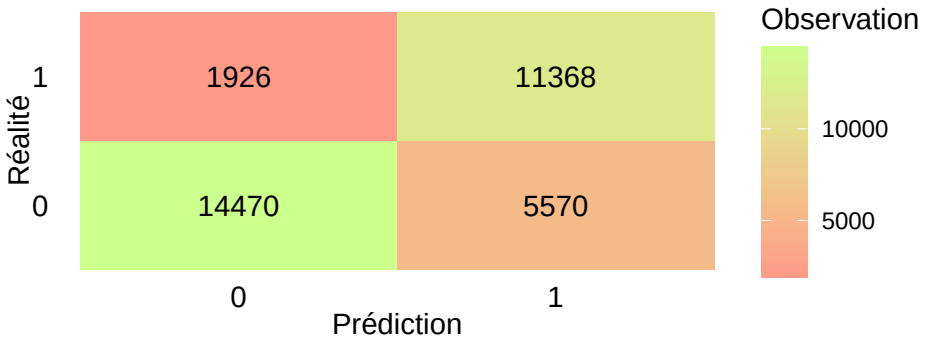
Les paramètres que nous avons optimisés sont :

1. **n_trees** : Ce paramètre détermine le nombre total d'arbres de décision à construire dans le modèle de Boosting. Une valeur plus élevée peut améliorer la performance du modèle mais aussi augmenter le risque d'overfitting et le temps de calcul.
2. **tree_depth** : Il s'agit de la profondeur maximale pour chaque arbre de décision. Une profondeur plus grande permet aux arbres de capturer des interactions plus complexes entre les variables, mais elle peut aussi conduire à de l'overfitting.
3. **learning rate** : Ce paramètre contrôle la contribution de chaque nouvel arbre ajouté au modèle. Un taux d'apprentissage plus faible nécessite plus d'arbres pour construire le modèle final, mais peut améliorer la généralisation du modèle en évitant un apprentissage trop rapide qui pourrait ignorer les subtilités des données.



Nous choisirons ntrees = 1200, tree_depth = 8 et learning_rate = 0.001

Matrice de confusion



Mesure des performances

Indicateur	Accuracy	Erreur global	Spécificité	Erreur Type 1	Sensibilité	Erreur Type 2	Précision
Valeur	77.51 %	22.49 %	72.21 %	27.79 %	85.51 %	14.49 %	67.12 %

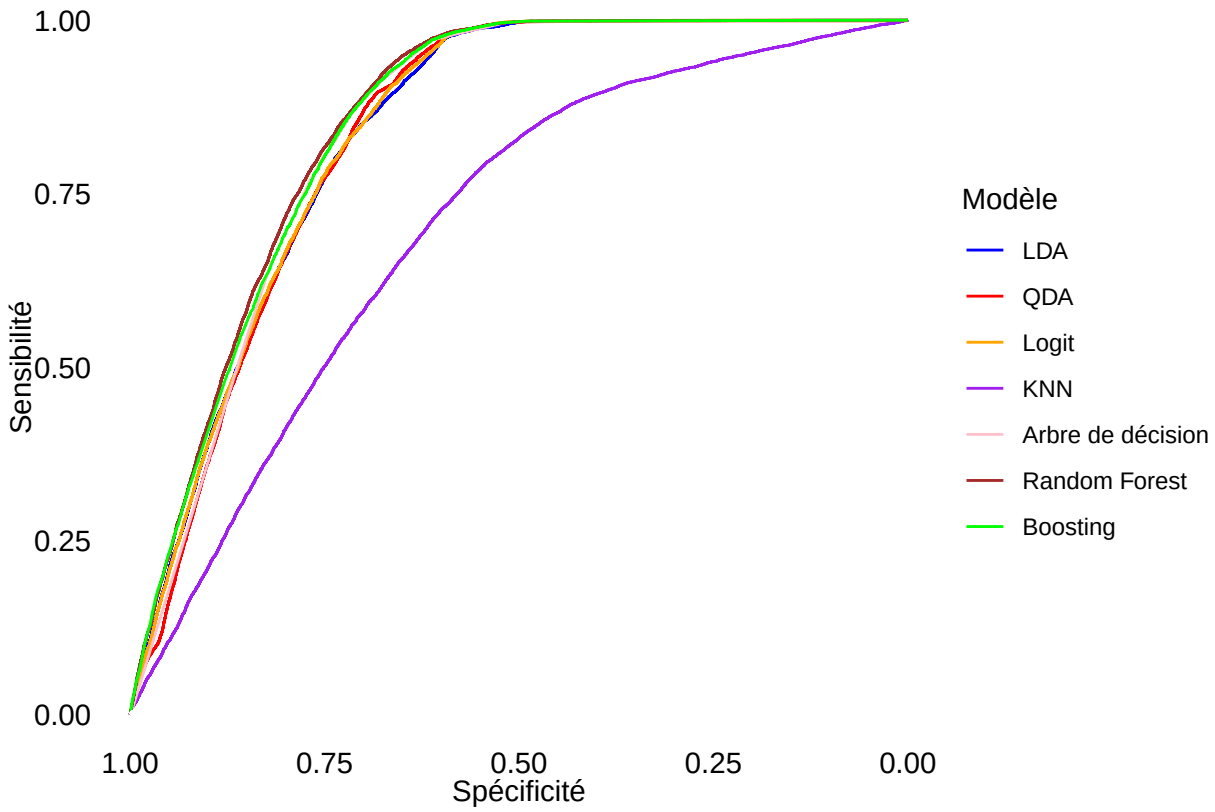
Mesure des performances

Le modèle Boosting présente une accuracy de 77.51 %, ce qui le place entre les modèles Decision Tree et Random Forest pour cet indicateur. Pour la sensibilité, elle est de 85.51 %, ce qui représente une légère diminution par rapport à Random Forest (85.92 %) et plus significative par rapport au Decision Tree (87.38 %).Et sur la Precision, elle est de 67.12 %, légèrement inférieure à Random Forest (67.26 %) mais supérieure à Decision Tree (66.65 %).

En comparant ces résultats avec ceux des modèles précédents, le modèle Boosting offre une performance globale compétitive, avec une accuracy proche du meilleur modèle Random Forest. Toutefois, il présente une légère baisse en sensibilité, indiquant une petite concession dans la capacité à détecter tous les vrais positifs. La précision du modèle Boosting reste solide, bien qu'elle soit légèrement inférieure à celle de Random Forest, suggérant une qualité légèrement moindre dans la prédiction correcte des vrais positifs.

Comparaison des modèles

Courbe ROC



Aire sous la courbe :

LDA	QDA	LOGIT	KNN	Arbre_décision	random_forest	boosting
0.839	0.837	0.841	0.703	0.844	0.855	0.851

La courbe ROC (Receiver Operating Characteristic) est un outil graphique très utilisé pour évaluer la qualité des modèles prédictifs en classification binaire. Elle représente la sensibilité (taux de vrais positifs) en fonction de 1 - spécificité (taux de faux positifs) à différents seuils de décision. L'aire sous la courbe ROC (AUC - Area Under the Curve) permet de quantifier la performance globale du modèle : une AUC de 1 indique une performance parfaite, tandis qu'une AUC de 0,5 correspond à une performance non meilleure qu'un choix aléatoire. La courbe ROC est particulièrement utile parce qu'elle est indépendante du seuil de classification et donne un aperçu de la performance du modèle sur l'ensemble des seuils possibles.

En comparant les courbes ROC des différents modèles, il est évident que la plupart des modèles ont des performances relativement similaires, avec des AUC très proches les unes des autres, à l'exception des KNN qui semblent avoir une performance significativement plus basse.Nous pouvons observer que le modèle Random Forest est celui qui offre la meilleure performance, avec une AUC de 0.855 . Le modèle Boosting arrive en deuxième position, avec une AUC de 0.851 , suivi par le modèle DT avec une AUC de 0.844 . Les modèles LOGIT LDA, et QDA présentent des performances très légèrement inférieures, avec des AUC respectives de 0.841, 0.839 et 0.837. Le modèle KNN est le moins performant avec un AUC de 0.703.

F1_Socre

Pour sélectionner le modèle le plus adapté à nos besoins, il est important de trouver un équilibre entre deux aspects principaux :

1. Bonne Précision : Un modèle qui prédit correctement les vrais positifs (TP - True Positives) tout en minimisant les faux positifs (FP - False Positives) est essentiel pour ne pas perdre de temps et de ressources à proposer des contrats d'assurance auto à des clients non intéressés.
2. Bonne Sensibilité : Un modèle avec un faible taux d'erreurs de seconde espèce (faux négatifs, FN) est important pour ne pas omettre de prédire les clients réellement intéressés. Ne pas identifier ces clients peut entraîner une perte d'opportunités de vente.

Le F1-score est une métrique qui aide à équilibrer la précision et la sensibilité (rappel), et sa formule est :

$$F1\ Score = \frac{2 \times (\text{precision} \times \text{recall})}{\text{precision} + \text{recall}}$$

Modèles	LDA	QDA	LOGIT	KNN	Decision tree	Random Forest	Boosting
F1_score	74.72 %	75.12 %	74.21 %	57.97 %	75.62 %	75.45 %	75.21 %

En se basant sur le tableau des scores F1 pour les différents modèles :

- Arbre de décision : Avec un F1-score de 75.62 %, l'arbre de décision offre une bonne balance entre précision et sensibilité. Les arbres de décision sont également faciles à comprendre et à expliquer, ce qui peut être rassurant et plus accessible pour le grand public.
- Forêt aléatoire (Random Forest) : Avec un F1-score similaire de 75.45 %, la forêt aléatoire est également un bon choix. Elle offre l'avantage de la robustesse grâce à la combinaison de plusieurs arbres et est généralement plus performante en termes de gestion des erreurs et de la variance. Tout comme l'arbre de décision, la forêt aléatoire peut être visualisée et expliquée relativement facilement.

Les deux modèles (arbre de décision et forêt aléatoire) présentent des F1-scores convenables et sont parmi les meilleurs modèles du tableau. Ils sont également parmi les plus interprétables, ce qui les rend adaptés pour une présentation ou une utilisation où la transparence et la compréhension du modèle sont importantes, notamment lorsqu'il s'agit de communiquer avec des non-experts.