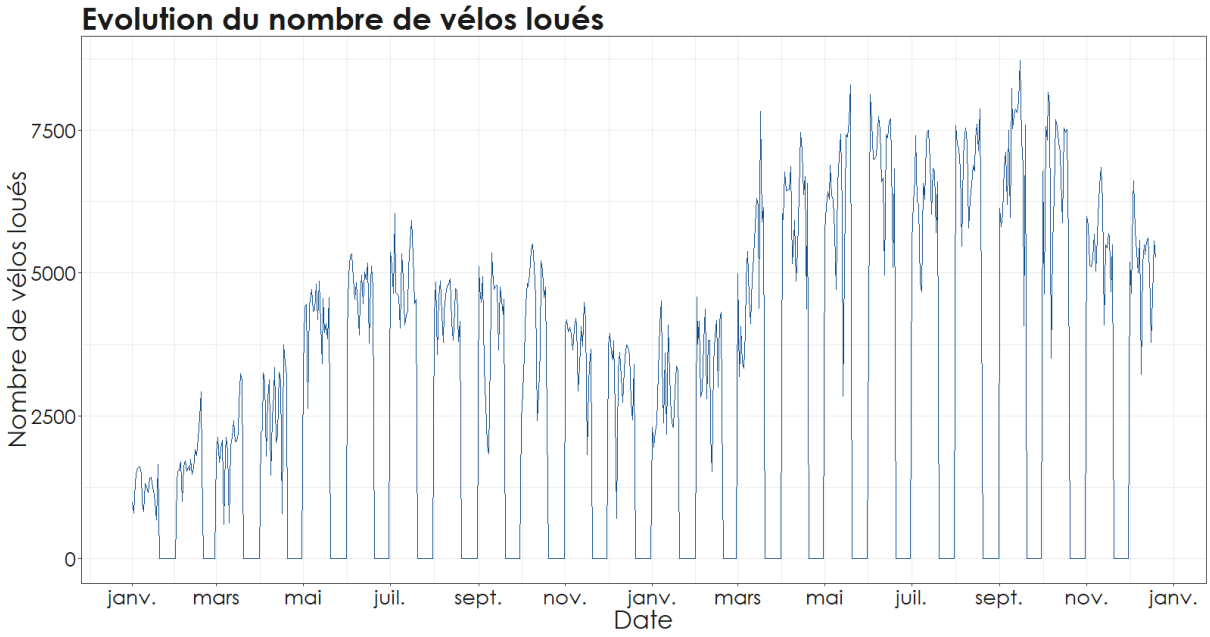
Test technique Quantmetry

Introduction

On cherche à modéliser le nombre de vélos loués par heure en se basant sur 2 années d'historique (2011 et 2012) de locations de vélos ainsi que des données météorologiques. La modélisation s'effectue en 2 parties : une analyse descriptive des données afin de détecter les facteurs les plus impactants puis la modélisation elle-même. La mise en place du modèle se fera en plusieurs étapes : préparation du dataset, test de plusieurs algorithmes, sélection du meilleur puis optimisation de celui-ci.

Partie I : Statistiques descriptives

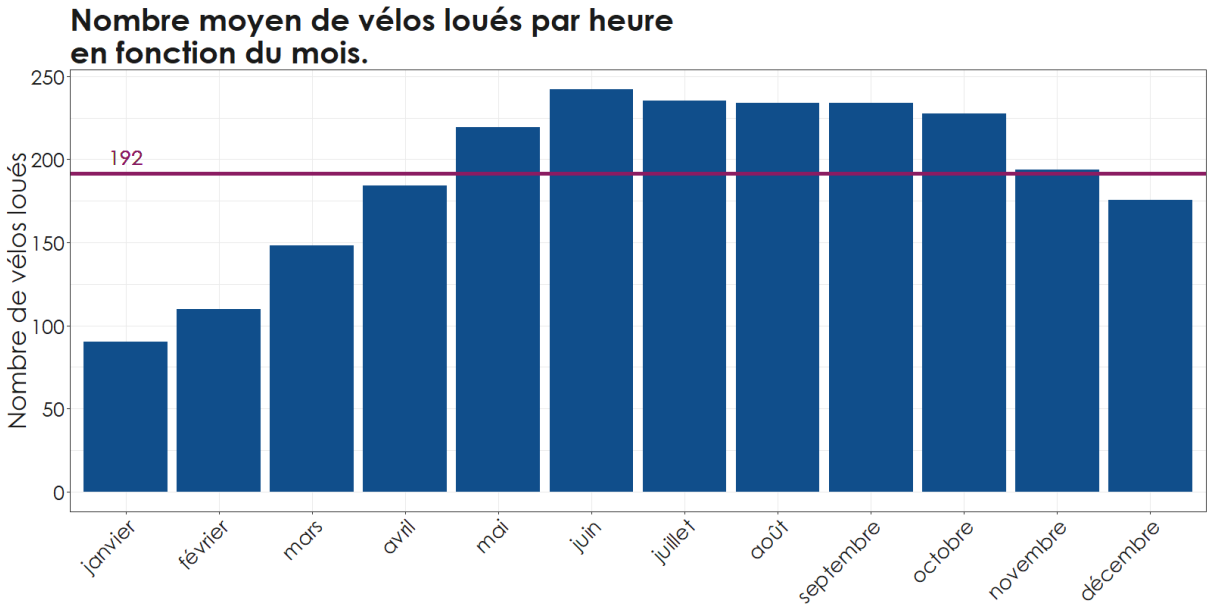
Pour commencer on regarde l'évolution du nombre de vélos loués dans le temps. Il s'agit d'un premier aperçu des données. Le nombre de vélos loués par jour est représenté sur la graphe suivant.



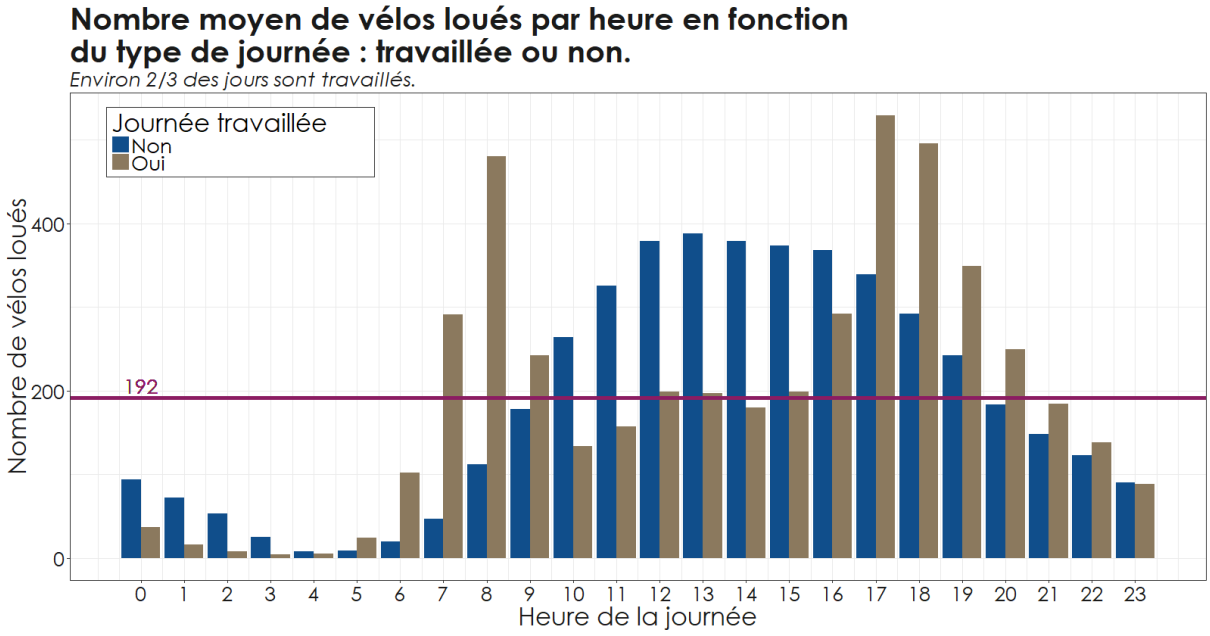
La première chose à remarquer est l'absence de données à chaque fin de mois. Il s'avère qu'il manque des données pour les 10 derniers jours de chaque mois. On peut noter une augmentation générale depuis janvier 2011 ainsi qu'une saisonnalité annuelle, probablement liée à la météo.

Pour étudier des différents facteur temporaux plusieurs variables ont été créées à partir de la date de location du vélo : l'année, le mois, le jour du mois, le jour de la semaine et l'heure de la journée.

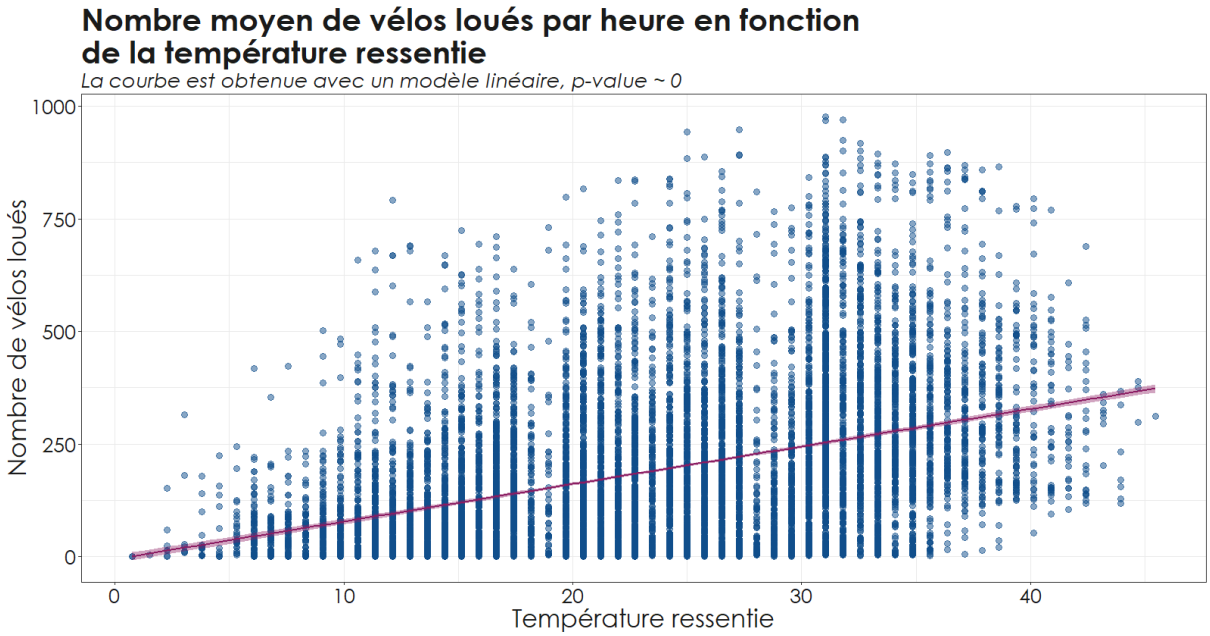
Le graphe suivant illustre l'importance du mois. Le trait horizontal correspond à la moyenne par heure du nombre de vélos loués sur l'ensemble de la période étudiée.



L'heure de la journée devrait également être un facteur très influençant du nombre de vélos loués. Les heures d'embauche et de débauche sont des heures de fortes affluences pour les jours travaillés tandis que le nombre de vélos loués les weekends par exemple est plus lissé sur la journée. Le graphe suivant illustre cette idée.



La météo est un facteur déterminant dans la location de vélos. Le graphe suivant montre le nombre de vélos loués en fonction de la température.



Le nombre de locations de vélos semblent augmenter avec la température. Un simple modèle linéaire confirme cela mais on peut aussi noter que pour les températures les plus élevées il y a moins de locations.

Les autres facteurs n'ont pas été retenus ici car ils présentent un lien avec le nombre de vélos loués moins important (jour du mois ou humidité par exemple).

Comparer deux populations

Dans le cas où le sexe et l'âge des utilisateurs serait fourni, une comparaison de la distribution des âges selon le sexe pourrait être faite. Pour cela on regarderait dans un premier temps les histogrammes des deux populations. Afin de tester si les deux distributions sont identiques on pourrait appliquer un test de Kolmogorov Smirnov.

Partie II : Modélisation

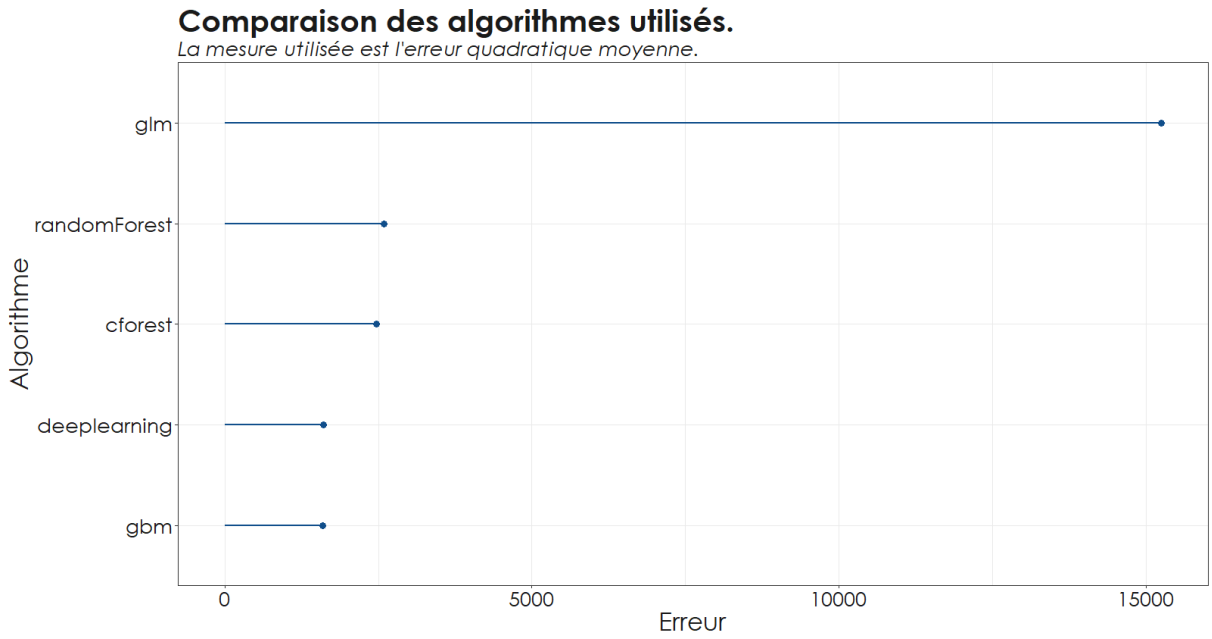
Après avoir préparé le dataset pour la modélisation, différents modèles ont été appliqués sur les données et leur performance comparée. Le meilleur algorithme à ce stade a ensuite été optimisé.

La préparation du dataset a consisté en la suppression de variables inutiles à la modélisation comme le timestamp de la location et les variables de nombres de vélos loués par type d'utilisateurs. Certaines variables ont été transformées en variables catégorielles pour être perçues comme telles par les modèles et non pas comme des variables numériques à cause de leur encodage (heure du jour, jour de la semaine).

Les algorithmes qui ont été appliqués sur le dataset sont les suivants : un modèle linéaire, une forêt, une forêt avec arbres conditionnels, support vector machine, gradient boosting et deep learning. Les paramètres par défaut de ces modèles ont été gardés pour la comparaison. Pour comparer les performances de ces algorithmes le dataset est coupé en un dataset d'apprentissage et un dataset de validation. Les modèles sont entrainés sur le premier et leur performance comparée sur le second.

La mesure de performance

La mesure de performance utilisée est l'erreur quadratique moyenne ainsi que le temps de calcul total. L'erreur quadratique est utilisée classiquement dans les problèmes de régression. Cette mesure est simplement la moyenne de la somme des carrés des erreurs. C'est un bon indicateur de la performance d'un modèle dont le but est de prédire le nombre de vélos loués, une erreur de prédiction à la hausse ou à la baisse ayant la même importance. Le temps de calcul est utilisé dans un second temps pour différencier deux algorithmes qui auraient des performances similaires. Un temps de calcul plus bas permettra par la suite de tester plus de paramètres pour optimiser le modèle.



Gradient boosting offre une performance légèrement meilleure que le deap learning. Comme il est également un peu plus rapide, c'est l'algorithme qui est retenu pour l'étape suivante.

Les trois paramètres qui vont être optimisés sont :

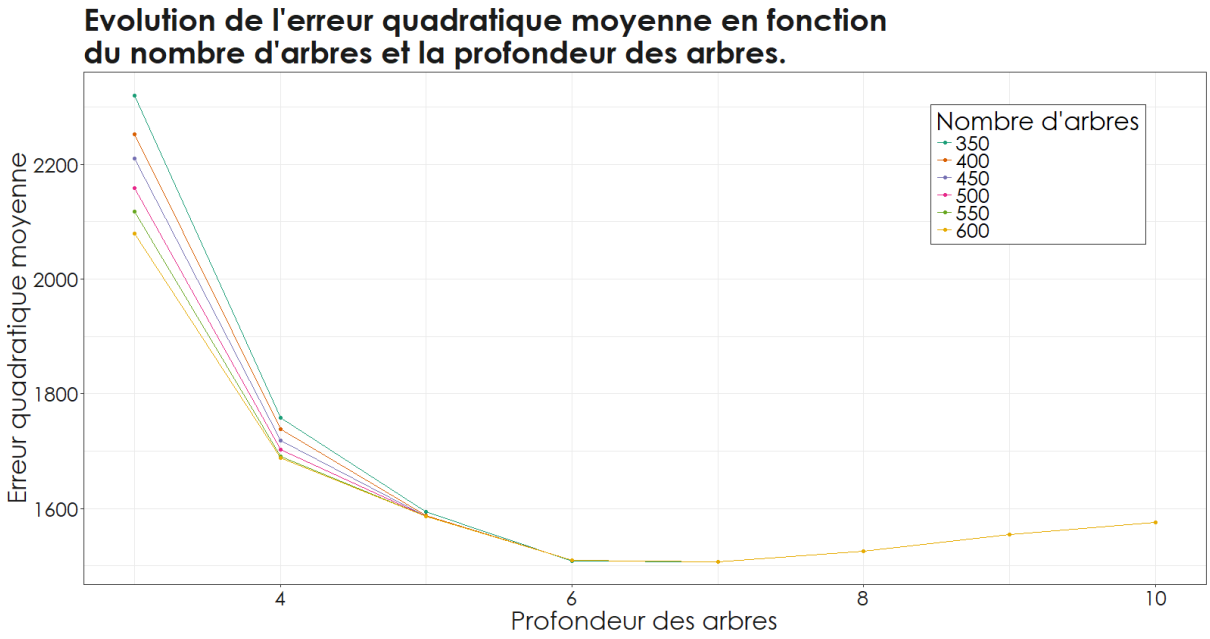
- Le nombre d'arbres : un nombre d'arbres plus important conduit à de meilleures performances mais peut aussi conduire à un sur-apprentissage.
- La profondeur des arbres : une profondeur élevée permet de mieux apprendre les relations entre les variables dépendantes et la variable cible mais peut aussi conduire à apprendre des relations trop spécifiques.
- Le coefficient de rétrécissement : il pénalise l'ajout d'un nouvel arbre dans le calcul final de la prédiction. Une faible valeur conduit à un modèle plus robuste car un arbre spécifique aura un impact moins important sur la prédiction finale mais une faible valeur nécessite aussi plus d'arbres pour garder une bonne précision.

Ces trois paramètres sont dans un premier temps optimisés un à un pour voir dans quelle mesure ils impactent la performance du modèle puis deux à deux pour voir quelle est la relation de dépendance entre chaque paire de paramètres.

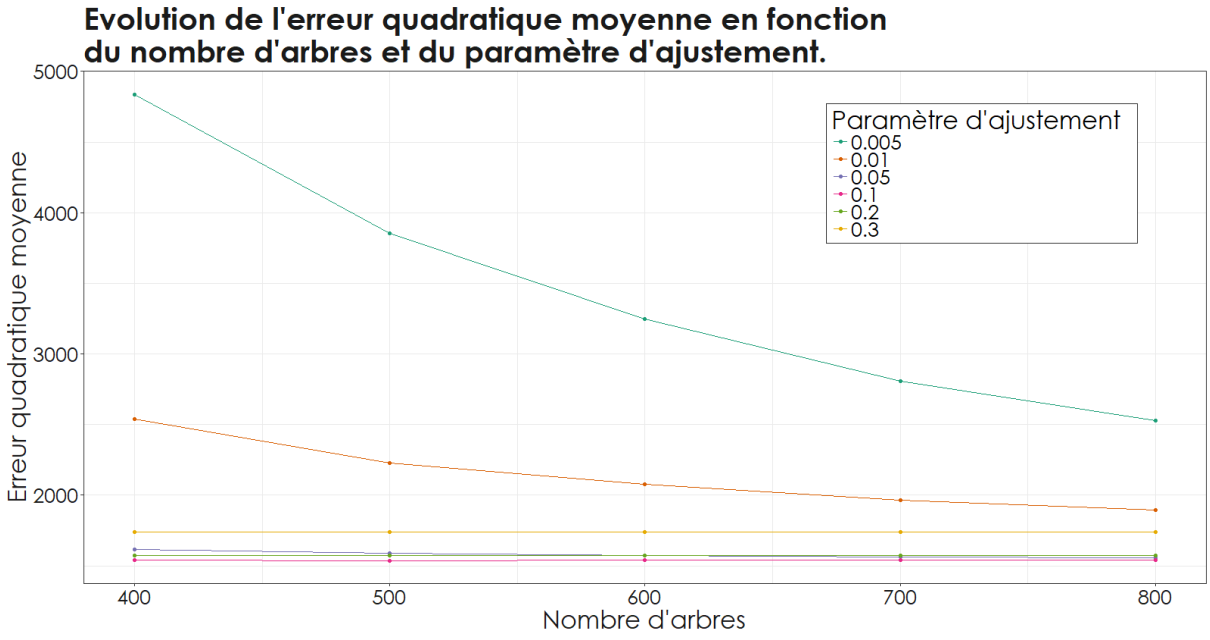
L'augmentation du nombre d'arbres fait baisser l'erreur mais à partir de 500 arbres on n'observe plus d'amélioration. De même l'augmentation de la profondeur des arbres contribue à une baisse de

l'erreur jusqu'à une profondeur de 6 puis l'erreur remonte légèrement. Le coefficient de rétrécissement est optimal à 0.2 mais les performances sont très similaires pour les valeurs supérieures à 0.1.

La profondeur des arbres n'impacte pas la valeur de nombre d'arbres à préférer, il s'agit toujours du plus grand nombre d'arbres.



Le graphe suivant montre bien la relation entre le nombre d'arbres et le coefficient de rétrécissement. On peut voir que la convergence vers l'optimum demande plus d'arbres pour les valeurs basses du coefficient de rétrécissement.



En affinant la grille de recherche on peut voir que l'optimal est obtenu pour un coefficient de 0.07 à partir de 500 arbres.

Après optimisation des trois paramètres en même temps, le meilleur résultat est obtenu avec 500 arbres, une profondeur de 7 et un coefficient de 0.06 pour une erreur quadratique moyenne de 1 470.

Vers une amélioration du modèle

Ce modèle pourrait être améliorer en affinant encore la grille de recherche des paramètres et en optimisant tous les paramètres du modèle utilisé (nombre minimal d'individus par feuille pour considérer une séparation ou pour accepter une feuille, nombre de variables à tester à chaque étape de la construction d'un arbre, …).

Il aurait aussi été possible d'appliquer ce processus d'optimisation à tous les algorithmes utilisés et de comparer la meilleure version de chaque algorithme et non pas la version classique.

Enfin prévoir le nombre de locations par des utilisateurs abonnés dans un premier temps et le nombre de locations d'utilisateurs non abonnés dans un second pour ensuite sommer les deux prévisions pourrait peut-être améliorer la prédiction.