Projet Machine Learning & Forecasting

Projet : Prediction solaire

Bastien Dussap // Antoine Faul
Année universitaire 2020-2021
M2 Mathématiques de l'aléatoire

Contents

1 Introduction

Dans le cadre du projet de machine learning pour la prévision, nous avons choisi d'étudier la prévision de radiation solaire reçue en un point donné.

Dans le marché énergétique, il est très important de réguler la production énergétique en fonction de la demande en raison de la difficulté de stocker l'énergie durablement.

Afin d'ajuster la production de ses centrales électriques notamment, il est donc intéressant pour un fournisseur d'énergie de prévoir la production en énergie renouvelable (énergie solaire, énergie éolienne) à un horizon de quelques heures ou quelques jours. Dans notre projet nous avons choisi de limiter la prévision à un horizon d'une heure car il s'agit du type de prévision nécessaires en pratique aux industriels. On peut voir le lien de ce

sujet avec la problématique de prédiction de la consommation électriques à l'échelle nationale souvent abordée comme exemple dans le cours.
En effet, si l'on dispose d'un estimé de cette consommation et de la production d'énergie renouvelables, ici solaire, alors l'on peut savoir quelle quantité d'énergie il reste à produire dans les centrales électriques ou fossiles.

Dans notre projet, nous allons chercher à prévoir l'irradiation solaire, exprimée comme une densité surfacique de puissance, reçue dans la station météorologique HI-SEAS se situant sur l'île d'Hawaï à environ 2500 mètres d'altitude sur les pentes du volcans Mauna Loa. Les données ont été récoltée sur une période de 4 mois entre Septembre et Décembre 2016. Les données sont fournies par des satellites de la NASA qui utilise cette base pour simuler les conditions de vies sur Mars et ont été mis à disposition de tous sur le site Kaggle.

Nous avons choisi de nous intéresser à ce sujet car le fait prédire la production d'énergie par une centrale photovoltaique peut permettre des économies considérables de production d'énergies fossiles. Il y a donc au delà de l'aspect économique pour les fournisseurs d'électricités qui veulent éviter des pertes énergétiques due à une sur-production et assurer une production suffisante pour satisfaire ses clients, un enjeu environnemental important. Cependant une sur-estimation de la capacité de production des centrales photovoltaiques peut mener dans le pire des cas à des pénuries locales d'électricité ou simplement à un achat d'énergie de dernière minute sur les marchés étrangers à des prix plus élevés. Il est donc essentiel de disposer de prévisions fiables et non biaisées.
De plus, dans le but d'augmenter la part des énergies renouvelables sur notre production totale la capacité à prédire la production d'énergie solaire est un atout important et peut pousser les acteurs du secteur de l'énergie à accélérer ce mouvement.

Il serait également intéressant d'effectuer des prévisions à plus long terme pour connaître les tendances globales par exemple dans l'optique de choisir un lieu d'installation d'une centrale photovoltaique.
Cependant nos données se limitent à un seul emplacement géographique et un laps de temps assez restreint donc nous avons choisi de se focaliser sur la prévision à cour terme de l'énergie solaire produite.

2 Analyse et traitements des données

2.1 Description des données

Il y a 32686 points de mesures répartis entre le premier septembre 2016 et le premier janvier 2017 soit un total de 4 mois. Cela correspond à environ 250 points de mesures par jour en moyenne, soit une mesure toutes les 5 minutes environ. Cependant, on ne dispose pas d'un échantillonage régulier et le nombre de mesures par jour est variable, cela posera un problème notamment si nous voulons traiter ce jeu de donnée comme une série temporelle, ce que l'on fera lors de l'étude des résidus de nos modèles.

Les variables explicatives fournies sont essentiellement météorologiques comme la température, l'humidité, la pression mais aussi des informations sur le vent notamment sa vitesse et sa direction.

On dispose aussi d'une variable UNIXTime, correspondant au nombre de secondes écoulées depuis le début de l'année 1970, permettant d'ordonner facilement les mesures mais aussi de la date en format yyyy-mm-dd et de l'heure locale. Enfin nous disposons de la date de lever et de coucher du soleil à Hawaï pour chaque jour.

2.2 Transformation des données

Tout d'abord nous avons remarqué que les unités utilisées étaient des unités de mesures américaines non standard. La première étapes de la transformation des données était donc de convertir ces unités. Par exemple :

obtenir des températures en degrés, des pressions en HPa ou des vitesses en km/h.

Nous avons à notre disposition l'heure de la journée ainsi que l'heure de levée et du couché du soleil, cependant ces variables ne sont que des procurations de l'information qui nous intéresse réellement : la position dans le ciel du soleil. A l'aide de la fonction *getSunLightPosition* du package *suncalc* nous avons obtenus les deux variables d'intérêt : Altitude et Azimuth, qui misent ensembles permettent d'obtenir précisément la position du soleil au temps t.

Nous avons également créé une variable Index qui permet d'indexer les données et des variables qui récupèrent l'heure de la journée, le jour dans le mois et la date totale.

2.3 Interpolation

Comme mentionné précédemment, il est important pour nous de disposer d'un échantillonage régulier de points de mesures afin de pouvoir appliquer des méthodes de séries temporelles, notamment sur les résidus de nos modèles. Pour cela nous avons choisi de créer artificiellement un point de mesure exactement toutes les 5 minutes en sélectionnant à chaque fois la valeur la plus proche dans le temps dont nous disposons. Pour les nombreuses valeurs manquantes nous avons utiliser une technique d'interpolation.

Sur l'ensemble de nos données il y avait 307 sauts. La majorité sont de petit saut (5min ou 10min manquantes) cependant certain saut sont plus conséquent comme le montre le graphique (1) :

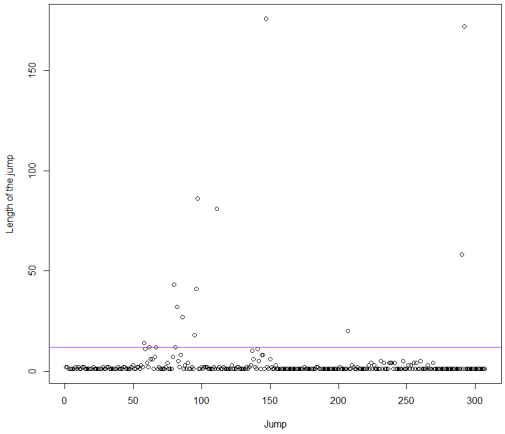

Figure 1: Longueur de chaque saut de données (ici une unité représente 5min). Les points au dessus de la ligne violette représente les sauts de plus d'une heure, soit 12 sauts de plus d'une heure

Pour les sauts de moins d'une heure, interpoler linéairement chaque variables est suffisant. En revanche pour les sauts plus long (notamment les deux plus long de 12h) cela conduira à des interpolations de très mauvaises qualités (voir le graphique 2). Cela risque d'avoir des répercussions sur la qualité de nos prévisions, même si la quantité de données interpolées ne représente que 3.8% de nos données totales. Pour résoudre ce problème nous choisissons d'interpoler les données des sauts de plus d'une heure, simplement en recopiant les données précédentes, ce faisant nous obtenons des doublons mais cela reste préférable à une simple interpolation linéaire.

Enfin il n'est pas nécessaire d'interpoler certaine variables explicatives comme Altitude et Azimuth qui sont entièrement déterminées par la date.

Notre objectif est de faire de la prévision à 1h, formellement cela signifie que, étant données un ensemble de donnée explicative à un instant t, nous souhaitons déterminer la radiation au temps $t+12$. Ainsi nous décalons notre jeu de donnée pour que la ligne i contienne les variables explicatives au temps i ainsi que la radiation au temps $i+12$. Enfin, nous pouvons ajouter une nouvelle variable appelée lag1h, celle-ci contient la radiation observée au temps i.

2.4 Statistiques descriptives

Pour expliquer la variable Radiation nous gardons qu'une petit nombre de variable explicatives : la pression, la température, le lag, l'humidité, l'altitude et l'azimuth, la vitesse du vent. Le climat Hawaïen est sub-tropical avec deux saisons: l'été entre Mai et Octobre qui est ensoleillé et chaud, et l'hiver entre novembre et avril pendant lequel les températures restent assez hautes mais le temps est généralement plus pluvieux. On peut donc s'attendre à une tendance décroissante de la radiation solaire entre les mois de septembre et de décembre. La variable Index devra dont être pris en compte dans nos modèle. Nous ne gardons pas les heures de levées et couché du soleil ainsi que l'heure car elles sont redondantes avec Altitude et Azimuth. Nous nous débarrassons

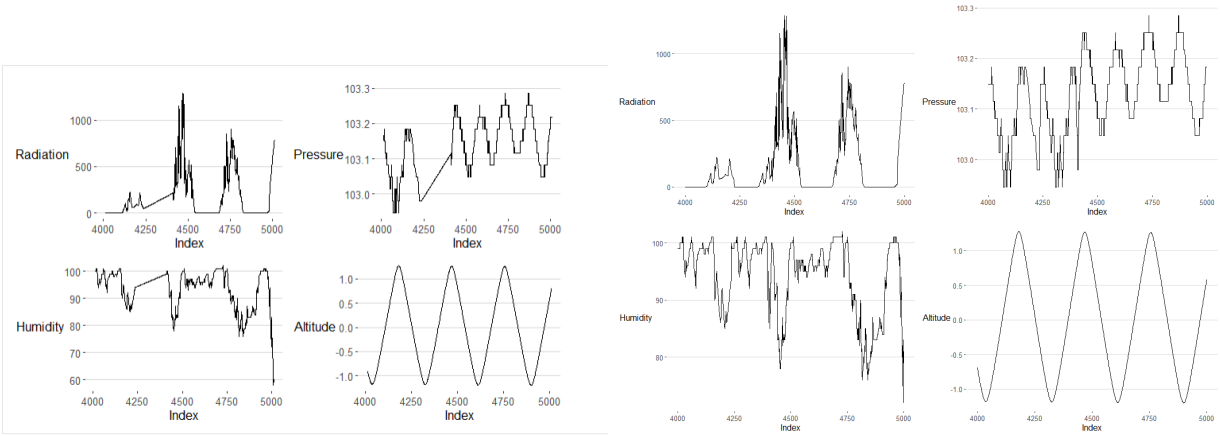

Figure 2: Comparaison de l'interpolation linéaire et du dédoublement au niveau du plus grand saut.

aussi de la variable direction du vent car nous la jugeons inutile. A partir de la nous pouvons tracer la matrice de corrélation (figure 3).

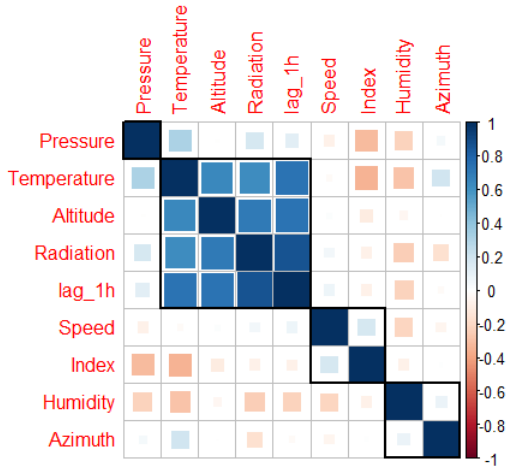

Figure 3: Matrice de corrélations des différentes variables

Cette première analyse nous permet de voir que les variables les plus importantes sont l'altitude, la température ainsi que le lag. L'humidité semble aussi joué. C'est donc ces variables que nous utiliserons en priorité dans nos modèles.

2.5 Jeu de donnée et modèle témoin

Nous séparons le jeu de données en deux. Nous utiliserons les données allant du 1er septembre 2016 au 5 décembre inclus comme jeu de données d'entraînement et nous testerons nos modèles sur le reste des données, soit du 6 décembre au 27 décembre. Nous avons choisi une durée de 3 semaines pour le jeu de donnée test afin d'avoir assez de temps pour faire de l'agrégation de modèle.

La figure 4, montre la première semaine du jeu de donnée test. Comme l'on peut voir la radiation solaire en un point est un phénomène très périodique. L'heure de la journée, représenté dans notre jeu de donnée par les variables altitude et azimuth, auront un impact déterminant sur la prédiction. En réalité, la date et l'heure, détermineraient complètement la radiation en un point si il n'y avait pas les nuages. Comme on peut le voit sur la figure, les radiations souffrent d'une très grande variance du à ces phénomènes météorologiques, qui sont ici accentués par la position de la balise, en effet elle se trouve sur le flanc d'un volcan, proche de la mer, et dans une zone à climat tropical, ainsi la météo est beaucoup plus changeante qu'en Île-De-France par exemple. Le minimum que l'on espère de nos modèles est qu'ils arrivent à prédire la périodicité de la radiation et donc de bien performer les jours de beaux temps (cf figure 5) . Les meilleurs modèles quand à eux devront être capables, à partir des données explicatives, de prédire les fluctuations météorologique. La prévision météo est un secteur à part entière et les données que nous exploitons ne sont pas suffisantes pour obtenir de bonne prévision de la météo.

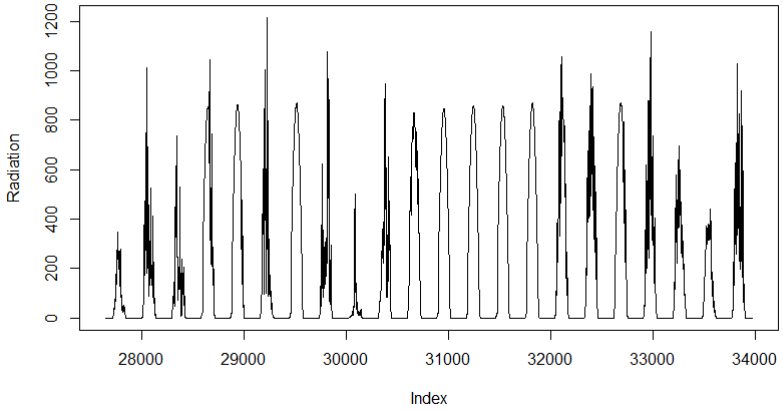

Figure 4: Jeu de données de test, allant du 6 décembre au 27 décembre.

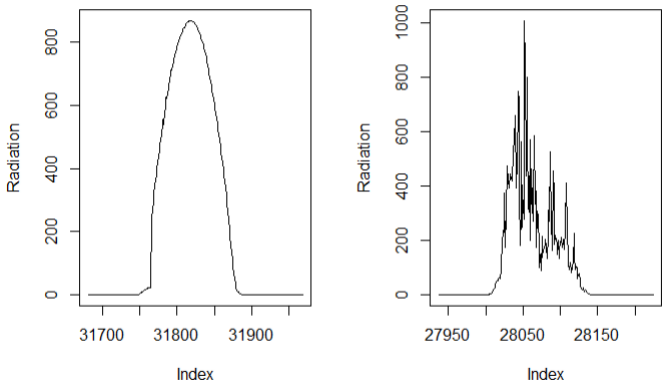

Figure 5: Comparaison entre une journée sans grandes fluctuations météorologique à gauche et une journée avec, à droite. Tous les modèles doivent êtres capable de bien prédire les journées comme celle de gauche, les meilleurs quand à eux seront capable de prédire celle de droite.

2.6 Choix de la métrique d'évaluation des performances

Afin d'évaluer les performances de nos différents modèles sur les ensembles de tests ou par validation croisée, il faut choisir une métrique nous permettant de comparer de manière efficace plusieurs modèles. En raison des très faibles valeurs de radiation solaire enregistrées la nuit, il semble inapproprié de choisir une métrique qui pénalise le pourcentage d'erreur sur la prédiction telle que le Mean Absolute Percentage Error (MAPE) souvent utilisé dans le cours. Dans notre cas, nous allons utiliser l'erreur quadratique moyenne comme métrique d'évaluation de nos modèles.

Pour avoir un point de repère nous utilisons comme modèle témoin, un modèle naïf consistant à prédire en la radiation au temps $t + 12$ comme étant la radiation au temps t. La figure 6 représente la prédiction ainsi que le RMSE de ce modèle.

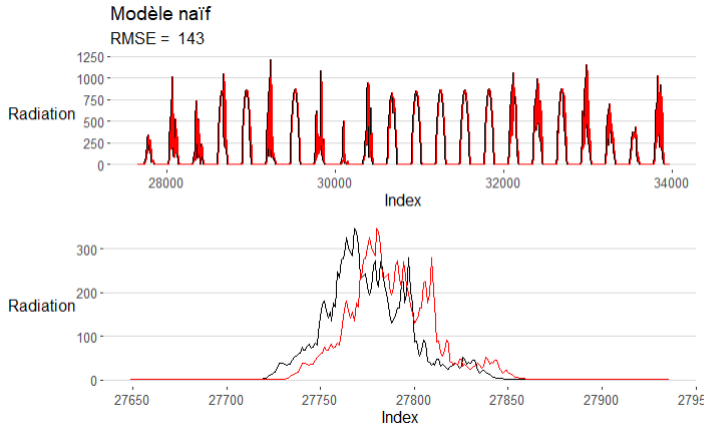

Figure 6: Prédiction et RMSE du modèle naïf.

3 Premiers modèles

3.1 Arbres de régression

Le premier modèle que nous utiliserons est un modèle d'arbre de regression. Celui-ci tentera d'expliquer la radiation en fonction des variables : Temperature, Pressure, Humidity, Altitude, Azimuth, Speed et lag1h. Pour ce faire nous commençons par entraîner un arbre maximal que nous "taillons" (prunning) afin d'obtenir un plus petit arbre. Notons que l'arbre d'origine obtient un RMSE de 141 ce qui est à peine meilleur que notre modèle naïf. Malheureusement, l'étape de prunning n'a pas améliorée l'erreur, en effet l'arbre "taillé" est trop simple, voir figure 7, pour pouvoir correctement capturer la complexité de la radiation solaire.

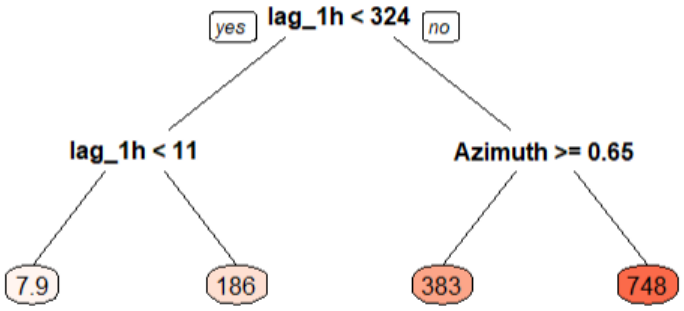

Figure 7: Arbre de décision

3.2 Bagging d'arbre

Pour obtenir un meilleur modèle nous choisissons de recourir à une technique de bagging. Pour ce faire, nous entraînons plusieurs arbres sur des sous-échantillons du jeu de donnée d'entraînement, puis nous estimons en prenant la moyenne. L'idée du bagging est de corriger la variance de nos arbres, c'est pourquoi il est préférable d'avoir des arbres avec une grande variance. Nous choisissons d'entraîner 10 arbres, sans les "tailler", afin d'avoir des arbres de grande variances. La question en suspend reste la taille des sous-échantillons à choisir pour le modèle, pour cela nous testons plusieurs valeur possible : 10% des données, 20%, 30% etc... On obtient de meilleurs résultats en prenant un faible nombre de données (10%).

La figure 8 montre les résultats obtenus pour le meilleur modèle, c'est à dire celui on l'on sélectionne 10% des données. Comme on peut le voir ajouter des arbres permet bien d'améliorer nos résultats mais le modèle finit quand même par converger.

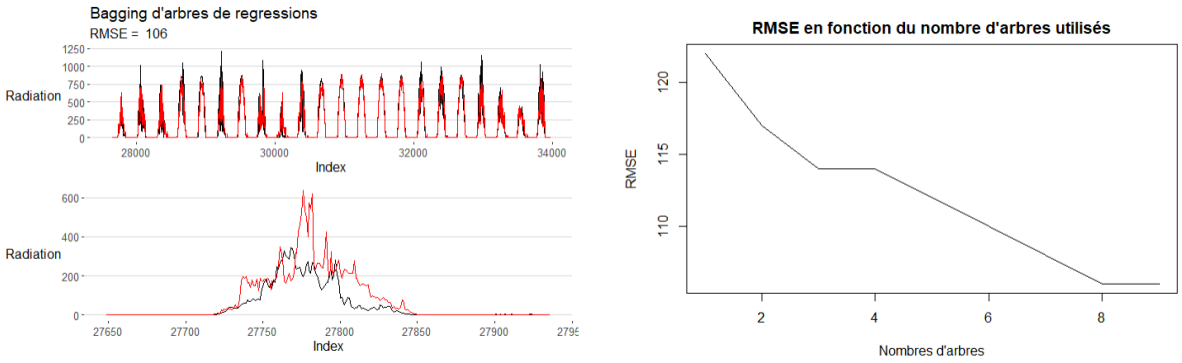

Figure 8: A gauche la prédiction du bagging d'arbres. A droite la RMSE en fonction du nombre d'arbres utilisés.

Le modèle ainsi obtenu prédit une courbe plus réaliste que la fonction constante par morceaux, trop simple, obtenue avec un arbre de régression taillé. Remarquons que le modèle arrive bien à capturer la périodicité du phénomène sans avoir eu besoin de lui expliciter cette périodicité.

3.3 Forêt aléatoires

L'étape suivante est donc d'utiliser une forêt aléatoire composée de 500 arbres. La figure 9 montre les résultats obtenues. L'utilisation d'une forêt surpasse le bagging aussi bien en terme de résultats (bien que ce soit très léger) que de temps de calculs. L'utilisation de la librairie *ranger* permet d'extraire du modèle l'importance qu'a chaque variables dans la prédiction. Comme on pouvait si attendre, le lag et l'altitude ont le plus d'impact sur le modèle, suivie de près par l'azimuth et la température.

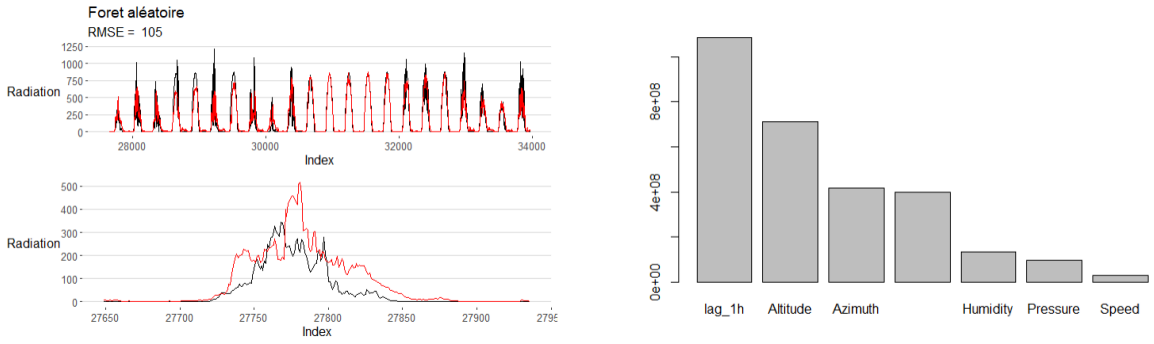

Figure 9: A gauche la prédiction de la forêt. A droite l'importance de chaque variable explicatives dans le modèle.

Pour mieux comprendre les résultats et l'importance de chaque variables, nous pouvons utiliser un "Accumulated Local Effects plots", voir figure 10. Cette figure nous montre comment la prédiction varie lorsque que l'on modifie légèrement une variable. Ce graphique permet de voir que la variable speed n'a presque aucun impact sur le modèle. Le modèle accorde une grande importance à l'heure de la journée comme nous pouvons le voir avec les variables altitude et azimuth. Ainsi une forêt aléatoire réussit à inférer l'aspect périodique de la radiation à l'aide de cette variable. L'aspect météorologique du problème est lui aussi bien inféré via notamment l'humidité : une forte humidité et le modèle semble l'avoir comprit aussi. Cependant certains résultats peuvent sembler perturbants. Par exemple, lorsque que la température est élevée, le modèle à tendance à prédire à la baisse, alors qu'une haute température est généralement associée à une absence de nuage et au soleil. De même lorsque la température est basse le modèle prédit à la hausse. Ce que nous supposons est que comme le modèle prédit à une heure, alors quand le température est au maximum de la journée (en après-midi), la radiation va avoir tendance à diminuer dans l'heure car on se rapproche de la nuit. C'est un peu le même raisonnement que l'on peut faire sur le graphique de la radiation avec un lag.
D'une manière générale, ces graphiques nous confortent dans l'idée que la forêt aléatoire est un bon modèle, car nous seulement elle rend de bon résultat, mais en plus elle les obtient pour de bonnes raisons, c'est à dire qu'elle n'a pas exploitée un quelconque biais dans nos données pour donner un bon résultat.

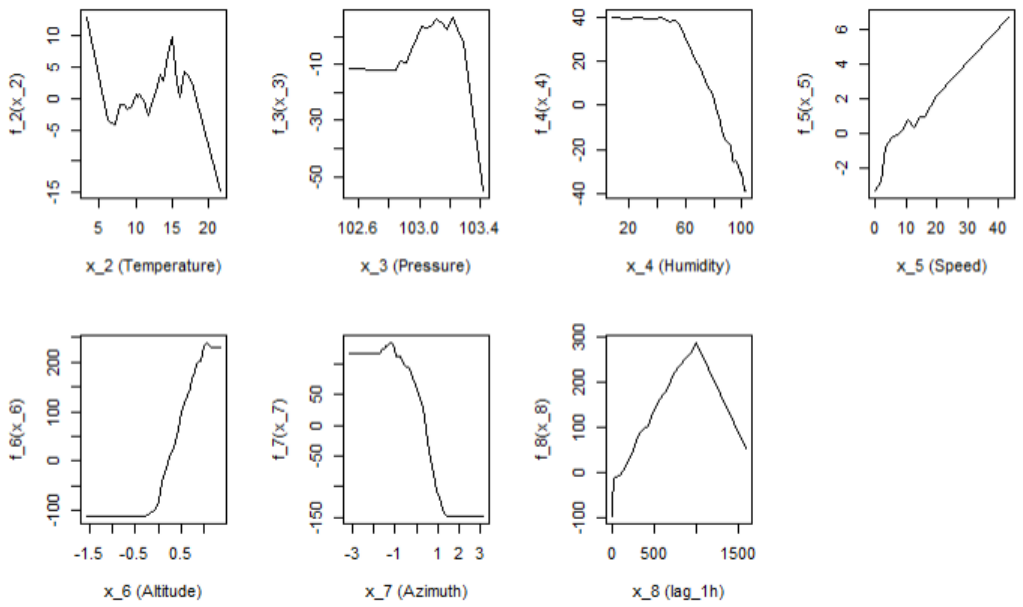

Figure 10: Accumulated Local Effects plots

3.4 Méthode ARIMA sur les résidus

La figure 11 représente les résidus de la forêt aléatoire, c'est à dire la différence entre la valeur prédite et la vrai radiation. On remarque que les résidus ont la forme d'une série temporelle. On suppose donc que l'utilisation d'un modèle de type ARIMA sur les résidus permettrait d'obtenir de bien meilleur résultats. Les paramètres du modèles sont calculer en utilisant un algorithme de sélection de modèle, ici "AIC".

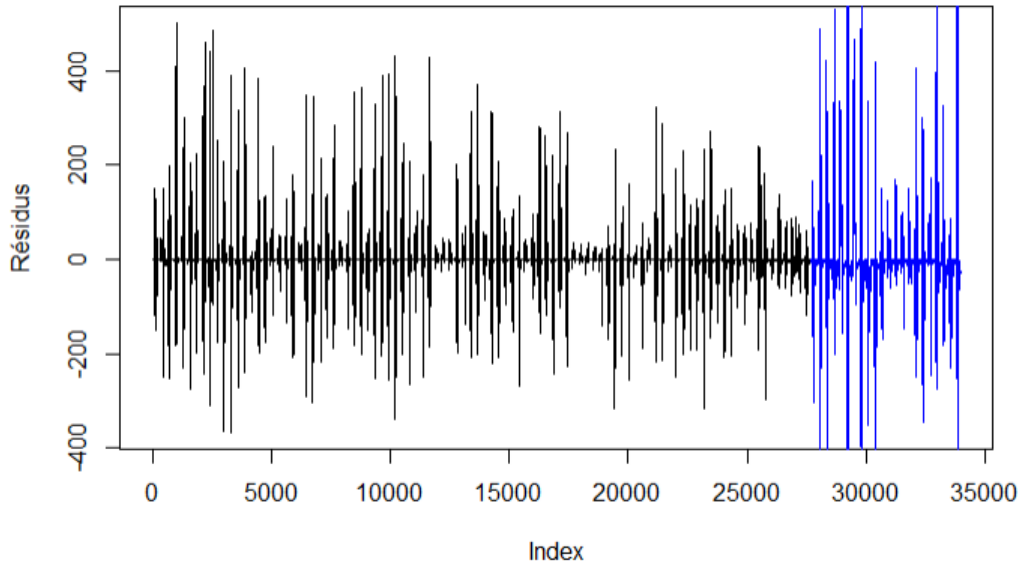

Figure 11: Les résidus de la forêt aléatoire. En noir, les résidus sur le jeu d'entraînement et en bleu, sur le jeu de test

Puisque les forêt aléatoire sont plus rapides à calculer que le bagging d'arbre tout en renvoyant des résultats équivalents nous avons choisi d'appliquer ce modèle uniquement sur la forêt aléatoire. La figure 12, représente les résultats du modèle. Comme on peut le voir sur cette figure,l'utilisation d'un modèle ARIMA sur les résidus améliorent grandement la RMSE. Ce modèle est identique à la forêt aléatoire les bonnes journées mais le surpasse les mauvaises. Ce modèle à cependant deux défauts, premièrement le modèle ARIMA est long à calculer notamment car on utilise de la sélection de modèle pour trouver les bon paramètres, le modèle est moins bon la nuit, parfois le modèle prédit même des radiations négatives, ce qui n'est pas fondamentalement

un problème car nous n'avons pas besoin d'un modèle de machine learning pour prédire la radiation solaire la nuit, on peut donc se contenter de regarder les résultats de la journée.

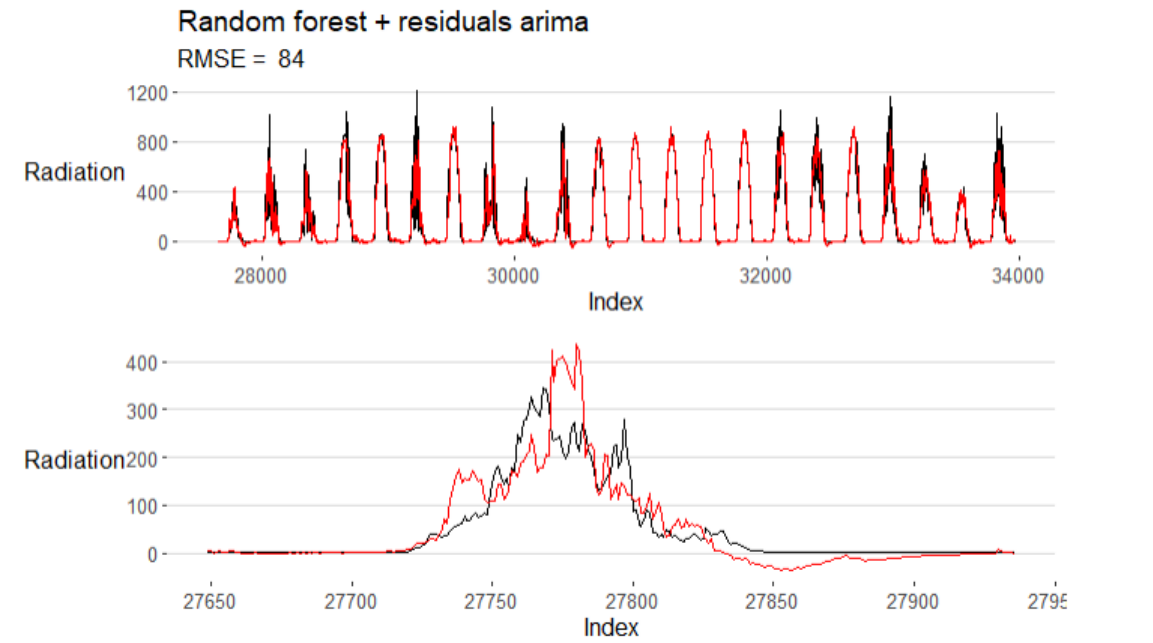

Figure 12: Résultats de la forêt aléatoire couplé avec un modèle ARIMA sur les résidus

4 Modèles additifs

Après avoir remarqué que les modèles purement linéaires ne permettaient pas une bonne prédiction de l'irradiation solaire notamment en raison de la forte périodicité quotidienne de la valeur cible, une idée naturelle peut être d'étendre ces modèles linéaires par des modèles additifs dans lesquels la non-linéarité des fonctions de la base de spline peut permettre de capter cette périodicité des données.

4.1 Choix des variables à sélectionner

Avec le diagramme d'importance effectué dans la partie sur les arbres de régression, on a l'avantage d'avoir déjà une bonne idée des variables qui vont être déterminantes pour la prédiction.

Les principales variables identifiées sont la température, la position angulaire du soleil par rapport aux panneaux photovoltaiques (altitude et azimuth) ainsi que lag1h qui représente la radiation solaire décalée d'une heure.

De plus, nous pouvons également prendre en compte le fait que les données proviennent toutes d'une période de 4 mois entre Septembre et Décembre. Donc la périodicité annuelle que l'on peut attendre n'est pas présente et on peut remarquer une légère tendance de baisse de la radiation entre Septembre et Décembre en raison de la saison hivernale qui débute vers Novembre à Hawai. On a donc décidé d'intégrer la variable Index qui va prendre en compte cette tendance.

Nous avons choisi dans un premier temps d'intégrer au fur et à mesure ces différentes variables explicatives dans nos modèles additifs. Il reste ensuite pour chaque variable un paramètre représentant le nombre de fonction dans la base de spline à optimiser ainsi que le choix du type de fonctions dans base de spline.

Il s'agit donc pour chacune de nos 4 variables identifiées à trouver le nombre de fonction optimal dans la base (noté K) et la base optimale. Pour cela, on a effectué sur chaque variable prise séparément une sélection de ces 2 paramètres en choisissant ceux qui minimisent l'erreur RMSE sur les données de test.

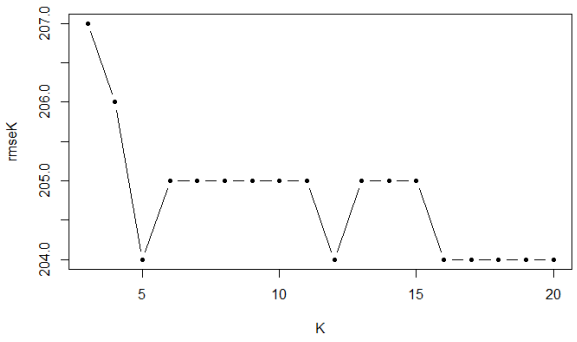

Figure 13: RMSE sur les données de test pour des modèles additifs utilisant la température comme unique variable explicative avec différentes valeurs de K

La figure 13, il s'agissait de trouver le meilleur paramètre K pour un modèle additif avec la température comme variable explicative. On remarque qu'il y a plusieurs valeurs de K qui obtiennent la même performance sur les données de tests, on va donc choisir la plus petite (K=5) afin d'avoir le modèle le plus simple possible. Toutes les bases de fonctions ont des résultats assez similaires donc on ne va pas prendre en compte ce paramètre.

Cependant avec cette approche, nous avons oublié de prendre en compte les effets de dépendances entre plusieurs variables explicatives sur la radiation. Par exemple, il serait intéressant d'étudier l'effet combiné de la température et de l'humidité ou de l'altitude et de l'azimuth. Une approche pour trouver les variables à rajouter dans le modèle est d'analyser les résidus par blocs de notre modèle et de voir si la variable que l'on souhaite rajouter est en mesure d'expliquer ces résidus. Si c'est le cas, alors on rajoute cette variable à notre modèle comme le montre 14.

Dans tous les cas les résulats obtenus avec des GAM ne sont pas satisfaisant en comparaison des résultats obtenus avec une forêt aléatoires (cf figure 15)

4.2 Modèles additifs avec poids exponentiels décroissants

Un moyen supplémentaire de prendre en compte la dépendance temporelle des données est de donner davantage d'importance aux données récentes qu'aux données anciennes pour prédire l'irradiation future. Cela revient à

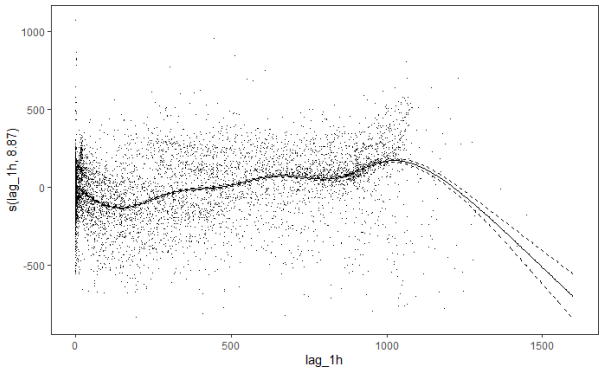

Figure 14: Nous avons entrainé un modèle et nous voulons savoir si nous pouvons entraîner un GAM sur les résidu en fonction de la variable lag1h. A en juger par la courbe obtenue, il semblerait qu'une partie des résidus puissent s'expliquer à partir de la lag1h et donc nous devons changer notre modèle pour y incorporer cette variable.

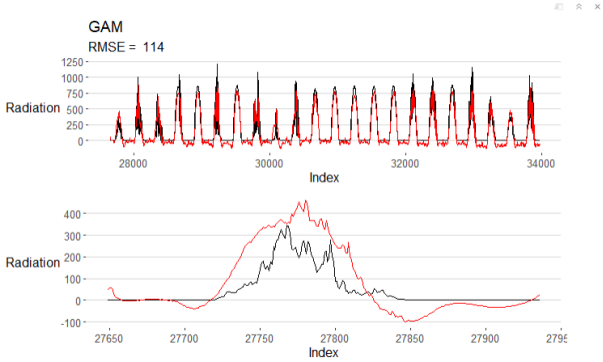

Figure 15: Prédiction et RMSE de notre GAM

prendre en compte la stabilité relative des variables météorologiques qui disposent d'une certaine "inertie" et ne peuvent pas changer de valeurs de manière instantanée. Pour traduire cette idée dans notre code, une solution est de donner des poids décroissants aux données pour que les données récentes disposent de plus d'importance dans la prévision.

Une méthode classique pour cela est d'utiliser des poids exponentiels décroissants puis d'ajuster un paramètre appelé mémoire qui correspond en quelque sorte à la durée de la mémoire pendant laquelle les données antérieures ont encore de l'importance dans la prévision future. On peut utiliser un modèle de validation croisée pour optimiser la valeur de ce paramètre. Les résultats obtenus ne sont pas très satisfaisants car ils ne sont pas meilleurs que le modèle additif construit précédemment.

4.3 Transfert learning avec un GAM

L'idée de ce modèle consiste à utiliser un modèle additif couplé à une forêt aléatoire. Pour ce faire on commence par entrainer un GAM sur nos données. la forêt aléatoires sera entraînée sur les variables et les coefficients du GAM, dans l'espoir que ceux ci donneront de l'information en plus. Malheureusement, le modèle obtenue est moins bon qu'une forêt aléatoires classique (cf figure 16)

4.4 Modèles additifs avec ARIMA

On reprend l'idée développée dans la partie sur les forets aléatoires consistant à essayer d'appliquer un modèle de série temporelle ARIMA sur les résidus de nos modèles additifs afin de capter les périodicités et d'améliorer les performances.

On remarque que comme pour les forêts aléatoires, cette méthode permet d'améliorer significativement les performances de nos modèles additifs, cependant ils restent tout de même largement moins efficaces que les meilleurs méthodes de forets aléatoires (cf17.
A priori nous pensions qu'un modèle de type GAM (et ses dérivées) serait le meilleur. Ce type de modèle fonctionnais bien avant jusqu'à ce qu'on décide de décaler les données de 1H depuis lors les résultats sont tous inférieure à ceux d'une forêt aléatoire.

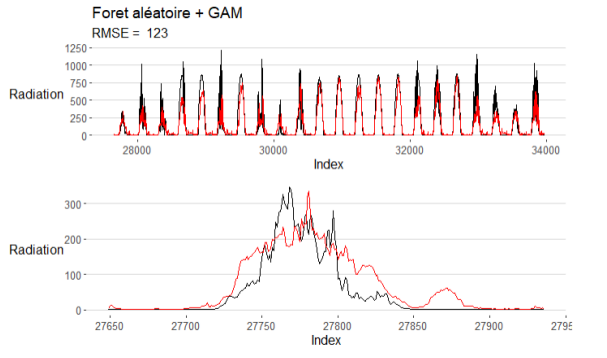

Figure 16: Prédiction et RMSE du modèle RF + GAM

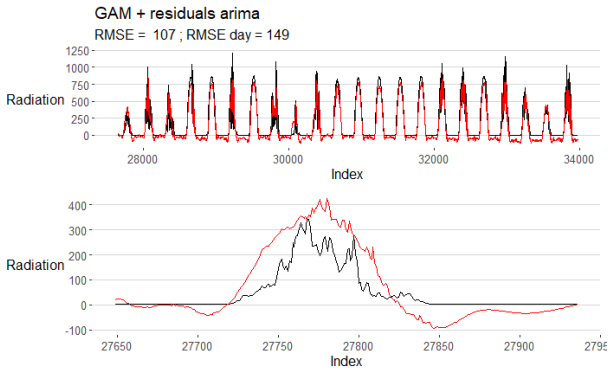

Figure 17: Performances de notre modèle additif amélioré avec un modèle ARIMA sur les résidus

5 Boosting et agrégation séquentielle de modèles

5.1 Gradient boosting

Dans cette partie, nous allons nous intéresser aux modèles de gradient boosting avec différents types de "weak learners" ou prédicteurs de base pour lesquels on va utiliser une partie des modèles (arbres,GAM) présentés précédemment. Le principe du gradient boosting est d'aprédgréger séquentiellement des prédicteurs de base sur des sous-ensembles (éventuellement aléatoires pour rajouter un effet de bagging) du jeu de données. L'idée de cette technique est d'ajouter successivement des prédicteurs de bases en les entraînant sur les résidus du modèle précédemment entraîné. Cependant en raison du coût computationnel trop important du choix du meilleur prédicteur à chaque étape l'idée est d'effectuer une descente de gradient

Ce type de méthode permet théoriquement à la fois de réduire la variance et le biais des prévisions. Deux paramètres sont importants dans ce type d'algorithmes: - le pas de descente du gradient, que l'on notera v - le nombre de "weak learners" utilisés ou nombre d'itérations de la descente de gradient, que l'on notera M

En utilisant des arbres de régression CART comme "weak learner" avec le package GBM, l'on obtient des résultats assez satisfaisants. Pour cela, on a optimisé le nombre d'arbres dans le modèle en utilisant une méthode d'Out-Of-Bag error, qui correspond à du boostrapping pour calculer les erreurs de validations de différents modèles. Avec le cours, on sait que le pas de gradient avec des arbres comme weak learners doit être très petit (typiquement entre 0.05 et 0,2). On peut donc réaliser une grid search avec une méthode de validation croisée pour déterminer le pas de gradient optimal.

Dans ce cas, le modèle optimal utilise 195 arbres et un pas de 0,1. Les performances du modèles sont assez bonnes. Il est également possible d'appliquer des modèles ARIMA à un horizon d'une heure sur les résidus de notre modèle de gradient boosting, ce qui permet encore une amélioration des performances de l'algorithme.

Ensuite, nous avons choisi de sélectionner les modèles additifs avec la formule établie précédemment comme weak learners (en utilisant le package GAM boost) car ils bénéficient également de relativement bonnes performances sur notre problème de prédiction et semblent pouvoir être améliorés notamment la nuit. Le principe de l'algorithme de gradient boosting reste similaire, comme avant on optimise le nombre de modèles utilisés cette fois-ci par validation croisée. Pour le pas de gradient, on choisit la valeur 0.1 utilisée assez classiquement.

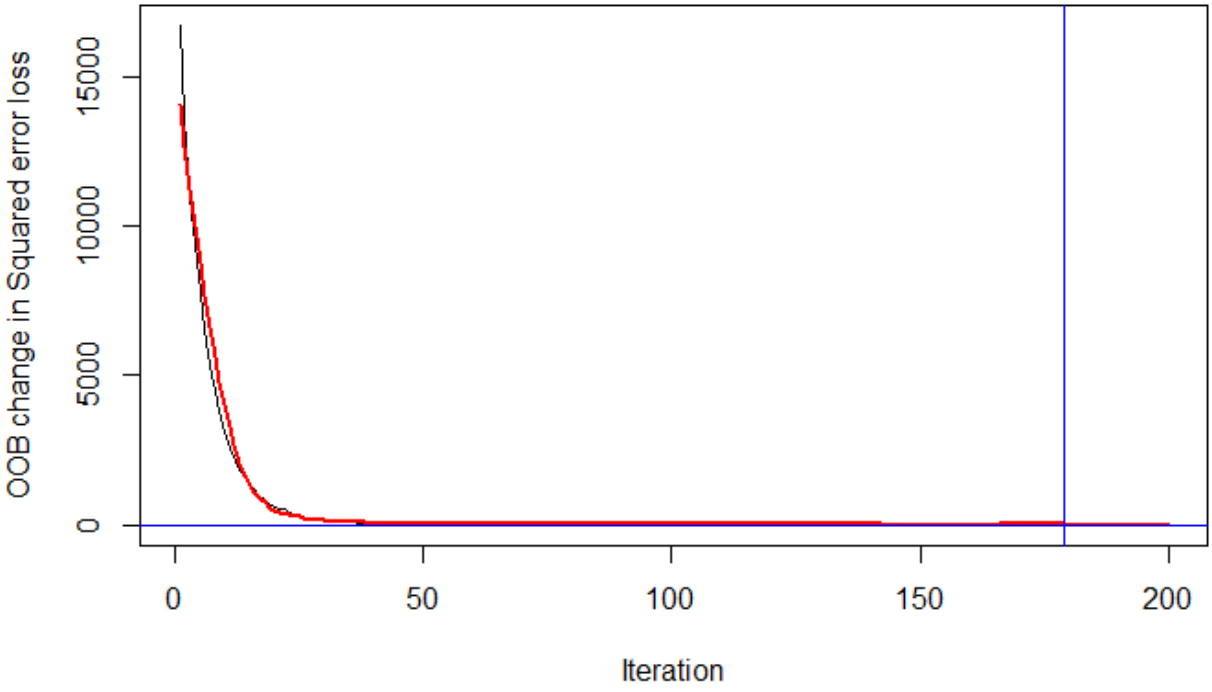

Figure 18: Évolution de l'erreur Out-Of-Bag en fonction du nombre d'itérations de notre algortihme de gradient boosting. Les pointillés bleus montrent que le minimum est atteint pour 195 itérations

5.2 Agrégations de modèles

Nous disposons désormais d'un grand nombres de modèles. Les modèles qui prennent en compte les résidus sont supérieurs aux autres et le modèle de forêt aléatoires avec étude des résidus par méthode ARIMA surpasse tous les autres. La dernière étape pour tenter d'améliorer nos résultats sera de combiner tous ses modèles avec un méthode d'agrégation de modèle. Nous utiliserons l'algorithme d'agrégation "MLpol".
Nous allons utiliser l'ensemble de nos modèles, même ceux dont les réultats sont médiocre, c'est à dire : le modèle naïf, un arbre de régression prunned, un bagging d'arbre, une fôret aléatoire, une forêt aléatoire et le modèle ARIMA, une méthode de gradient boosting avec et sans modèle ARIMA, un GAM boosted, un GAM, un GAM avec poids exponentielles, un GAM avec un modèle ARIMA et un GAM couplé avec une forêt aléatoire. Soit un total de 12 modèles. Bien une grande partie de ces modèles ont des résultats médiocres mais la plage de donnée est suffisamment longue pour que le modèle d'agrégations décide de lui même de mettre un poids de zéro sur ces modèles.
La figure 20 représente les résultats obtenus.

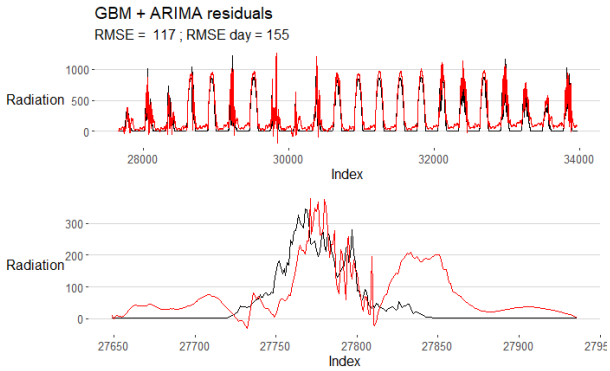

Figure 19: Performances du modèle de gradient boosting avec la librairie GBM en appliquant une méthode ARIMA sur les résidus

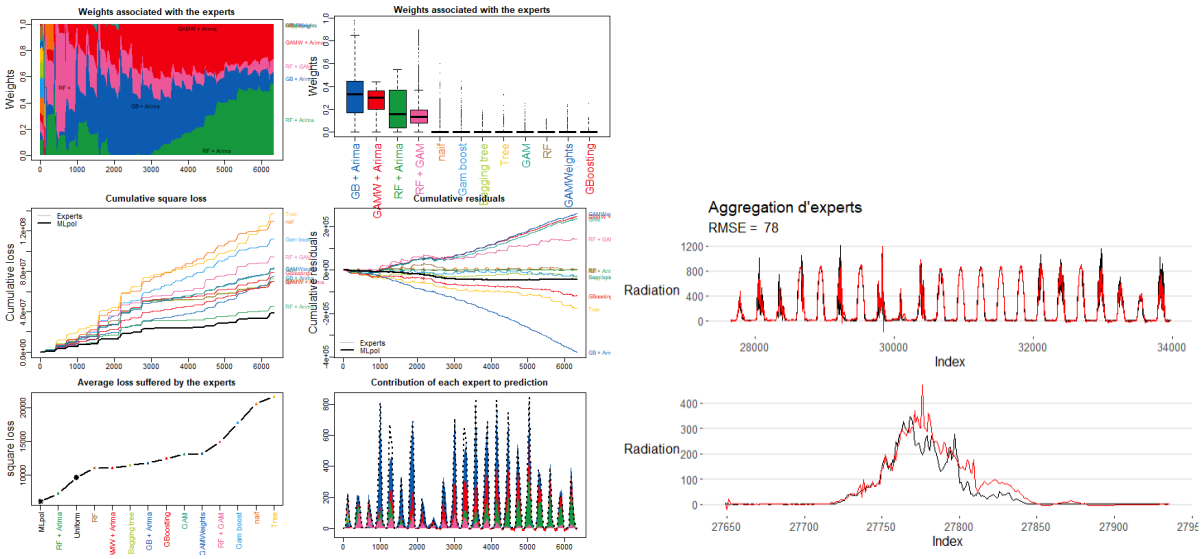

Figure 20: A gauche le résumé de notre modèle. A droite la prédiction du modèle d'agrégation.

Comme attendu l'algorithme n'utilise en réalité que les meilleurs et très vite ne mets plus aucun poids sur les moins bons. Le tableau suivant représente les poids mis sur 4 modèles (les autres ont un poids de 0)

RF + ARIMA	GB + ARIMA	RF + GAM	GAM + ARIMA
0.544	0.009	0.103	0.263

Ce modèle d'agrégation obtient une RMSE de 76, ce qui est meilleur que tous nos modèles précédent. On remarque cependant une curiosité, au vue de nos modèles il aurait été logique que l'algorithme mettent un gros poids sur le meilleur modèle (RF + ARIMA) très rapidement, ce n'est pourtant pas le cas. Après la première semaine la majorité du poids est partagé entre le modèle de gradient boosting + ARIMA et le modèle GAM + ARIMA, le meilleur modèle ayant quand à lui un poids de 0. Le choix d'avoir pris une période de 3 semaines pour le jeu de donnée test prend tout son sens ici.

6 Conclusion

Dans ce travail, nous avons choisi de nous restreindre à de la prévision de radiation solaire à un horizon très court (1 heure) ce qui est très demandé par les industriels du secteur de l'énergie pour ajuster la production des autres sources énergétiques. Pendant ce laps de temps très court, le variables météorologiques ne sont pas très difficiles à prévoir ce qui nous facilite la tache pour évaluer la radiation solaire, même si la localisation du site en montagne et proche de la mer favorise les changement météorologiques assez brutaux . On obtient ainsi des résultats très satisfaisants cependant une prévision à plus long terme et donc avec davantage d'incertitudes sur les données météorologiques serait sous doute plus délicate.

Au cours de ce projet, nous avons testé de nombreux modèles différents mais nous en avons uniquement sélectionnés quelques uns alors que d'autres tels que les régressions linéaires ont été éliminés. Les meilleurs résultats sont obtenus par l'agrégation séquentielle de différents modèles. Individuellement le meilleur modèle obtenu est une foret aléatoire sur laquelle on a utilisé un modèle ARIMA sur les résidus permettant de prendre en compte la périodicité des données.

Certains de nos modèles, comme les GAM, effectuaient une erreur importante la nuit, nous avons pensé initialement que cette erreur de nuit était la raison pour laquelle les GAM étaient moins performants que les forêts aléatoires, cependant après vérification l'erreur commise par les GAM le jour n'est pas meilleur que celle des forêts. Le fait d'agréger de manière séquentielle des modèles qui font des erreurs différentes permet de réduire l'erreur effectuée par le prédicteur final.

On peut remarquer que dans beaucoup de nos modèles l'ajout d'un modèle ARIMA sur les résidus permet une amélioration significative des performances. Nous pouvons être globalement très satisfait des performances de nos modèles de prévision à une horizon d'une heure. Il reste néanmoins d'autres pistes à exploiter pour améliorer nos modèles que nous n'avons pas développées dans ce travail. Par exemple une approche plus distincte entre les prévisions effectuées le jour ou la nuit, voire même une séparation des données pour chaque heure de la journée.

On peut regretter que notre jeu de données ne contiennent que des informations sur une durée de 4 mois. En effet, cela nous empêche d'exploiter une périodicité annuelle assez évidente de la radiation ainsi que les éventuels effets à plus long terme de phénomènes météorologiques. Ce qui aurait pu être assez intéressant dans des problématiques de choix d'emplacement pour des installations de centrales photovoltaiques si l'on avait également disposé de données étendues spatialement.

Finalement, nous avons trouvé pas mal de similarités entre ce problème de prévision de la radiation solaire et le problème type vu en cours de prévision de la consommation énergétique au niveau national. On retrouve globalement les mêmes modèles qui sont efficaces.