HPC-Big Data 2021 : Apprentissage Statistique TP 3

Arbres binaires

Description des données :

Les outils testés dans ce TP seront les arbres binaires en régression puis discrimination.

Un producteur éolien utilise des prévisions de force de vent pour en déduire, par exploitation de la courbe de réponse des éoliennes, sa production électrique du lendemain. Afin d'assurer l'équilibre du réseau national de transport d'électricité, cette prévision de production est en effet exigée par le gestionnaire RTE du réseau français.
Non satisfait des prévisions de vent calculées par un modèle météorologique, ce producteur vous demande de lui fournir un modèle statistique capable d'améliorer les scores de prévisions de force de vent sur son parc éolien. Il aurait également besoin, pour son activité, de prévisions d'occurrence de dépassement du seuil de **13 m/s**.

Vous disposez d'une archive de prévisions de différentes variables, issues du modèle météorologique exploité jusqu'alors par votre client, ces prévisions constituant de potentiels prédicteurs pour vos modèles statistiques, ainsi que de l'archive correspondante des mesures de force de vent effectuées sur le parc éolien du producteur.

Le fichier ***DataTP.txt*** contient les 11 variables suivantes :

HU : humidité relative prévue en %

N : nébulosité (= couverture nuageuse) prévue en octas (entiers de 0 à 8)

P : pression prévue en hPa

u : composante zonale du vent prévue en m/s

v : composante méridienne du vent prévue en m/s

hel : hélicité prévue en m^2/s^2 (indice de vorticité)

DD : direction du vent prévue en rad

mois : mois de validité de la prévision

heure : heure de validité de la prévision

FFp : force du vent prévue en m/s

FFo : force du vent **observée** en m/s.

1. **Chargement des librairies et des données :**

Après installation, charger les packages : ***rpart, partykit***

Charger les données dans une data.frame :
data=read.table("DataTP.txt",header=TRUE)

2. **Arbres binaires :**

- La librairie ***rpart*** propose les techniques CART via sa fonction ***rpart***. Différents paramètres en contrôlent l'exécution : le coefficient de pénalisation pour la construction de l'arbre maximal (***cp*** paramètre de complexité, par défaut 0.01), le nombre minimal d'observations par nœud (***minsplit***, par défaut 20), la profondeur maximale des noeuds (***maxdepth*** , par défaut 30) → ?rpart.control.

- Estimer **un arbre de régression** avec un coefficient cp nul, permettant de prévoir le prédictand FFo (arbre maximal) :
rpartreg.out = rpart(FFo ~ . , data, cp=0)

  Tester les fonctions ***summary, print, plot, text, plot(as.partykit( ))***

  Procéder à un élagage de l'arbre maximal assurément sur-apprenti car présentant trop de feuilles. La fonction ***rpart*** intègre le calcul d'erreurs par validation croisée pour toute une séquence de coefficients de pénalisation de la complexité ***cp***. Tester les fonctions ***printcp*** et ***plotcp***, analyser le tracé de l'erreur relative sur test (erreur quadratique moyenne divisée par la variance du prédictand) en fonction du coefficient ***cp*** et en déduire un arbre élagué.

  Confronter l'arbre maximal et l'arbre élagué en terme de **RMSE** sur apprentissage et test avec votre procédure d'évaluation.

  Comparer les performances des arbres avec celles du meilleur modèle de régression obtenus lors des TP précédents. Conclure.

- De même, estimer **l'arbre de discrimination** maximal permettant de prévoir le prédictand OCC défini au TP1.

  Procéder à un élagage et évaluer les 2 arbres en terme de **PSS**, **BS** ou **ROC AREA**.

  Comparer avec une régression logistique ou une analyse discriminante. Conclure.