Fiche TD avec le logiciel R : FAS

Forêts aléatoires sombres

Pr Jean R. Lobry

En guise de cadeau pour le départ à la retraite de ma collègue Anne-Béatrice Dufour, cette fiche propose d'explorer l'utilisation des forêts aléatoires sur un cas concret, la relation entre la composition des protéomes et la température optimale de croissance chez les bactéries, à des fins interprétatives ou prédictives.

Table des matières

1 Introduction

1.1 Source de la motivation

En tant que biométricien, j'ai longtemps fait la moue face aux méthodes de type forêt aléatoires (*Random Forest*) que nous allons explorer ici. J'ai fait mien l'aphorisme de René Thom [28] : « prédire n'est pas expliquer. » Je considérais ces méthodes comme des boites noires peu informatives, d'où le titre de cette fiche en forme de clin d'œil à la magnifique trilogie du *Problème à trois corps* de Liu Cixin [1].

Mon regard a commencé à changer quand je me suis intéressé au taux de couvert en chênes tempérés en France [2]. Dans les documents techniques expliquant l'exploitation des données satellitaires du service Copernicus de surveillance des terres de l'union européenne (CORINE Land Cover : CLC [3]), je me suis rendu compte que les forêts aléatoires étaient utilisées pour dresser une typologie de l'utilisation des sols [2, *e.g.*]. Il y avait donc une utilisation des forêts aléatoires dans un contexte opérationnel on ne peut plus sérieux.

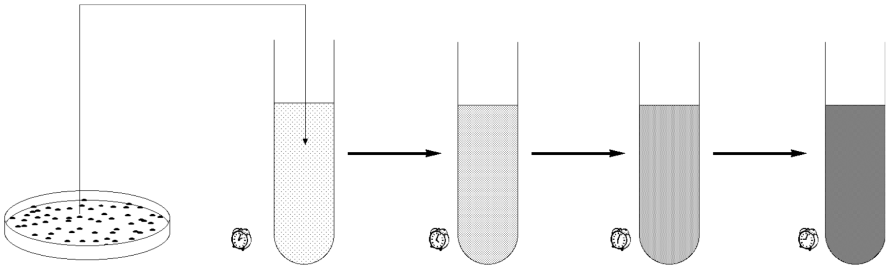
Un deuxième déclic est venu quand j'ai pris connaissance d'un rapport de la chambre régionale des comptes de la région Auvergne-Rhône-Alpes [1] utilisant des forêt aléatoires pour analyser la satisfaction générale des usagers des transports en commun de l'agglomération lyonnaise en fonction de divers caractéristiques de l'offre (*e.g.* fréquence de passage, accessibilité, confort). Il y avait donc une forte percolation des forêts aléatoires dans le monde des utilisateurs des méthodes statistiques.

Mais la bascule définitive eût lieu lorsque, à l'occasion de son départ à la retraite, ma collègue Anne-Béatrice Dufour m'a légué le livre [18] des actes des Journées d'Étude en Statistiques (JES) organisées par la Société Française de Statistique (SFdS) en 2016 sur le thème de l'*Apprentissage statistique et données massives*. Ce livre contient un article [9] de Robin Genuer et Jean-Michel Poggi exposant de façon particulièrement claire les forêts aléatoires et surtout proposant une méthode pour les utiliser à des fins de sélection de variables, et donc de sortir de la boite noire. Il me fallait donc absolument explorer les forêts aléatoires, en suivant une démarche classique en biométrie : que donne cette nouvelle méthode sur un problème déjà exploré avec des approches plus traditionnelles ?

1.2 Les données utilisées

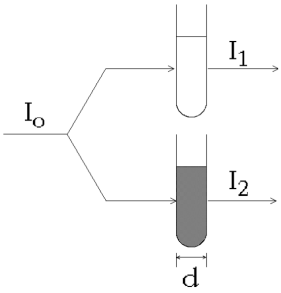
Lorsque l'on inocule un milieu de culture liquide avec une colonie de micro-organismes issue d'une boite de Petri, on observe une opacification progressive d'icelui.

1. *Le Problème à trois corps*, *La Forêt sombre* et *La Mort immortelle*.
2. Voir `https://pbil.univ-lyon1.fr/R/pdf/TCACPS.pdf`
3. `https://land.copernicus.eu/en/products/corine-land-cover`



La mesure de l'opacité du milieu de culture est une méthode très employée pour mesurer la biomasse, c'est-à-dire la masse sèche par unité de volume, parce qu'il y a une relation empirique bien établie [4], analogue à la loi de Beer–Lambert en chimie, entre l'absorbance, A, du milieu et la biomasse B :

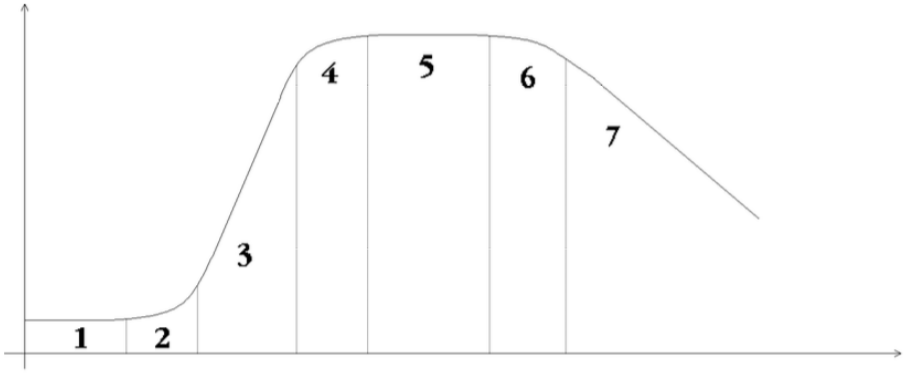
$$A = \log_{10}\frac{I_0}{I_2} - \log_{10}\frac{I_0}{I_1} = \log_{10}\frac{I_1}{I_2} = \alpha \mathrm{d} B \tag{1}$$



Comme représenté dans la marge, I_0 est l'intensité de la lumière incidente, I_1 l'intensité de la lumière transmise sans biomasse (blanc optique), I_2 l'intensité de la lumière transmise avec biomasse, d la longueur en cm du chemin optique et α une constante de proportionnalité. L'absorbance est souvent exprimée pour un chemin optique de 1 cm pour définir la densité optique (DO) du milieu :

$$\mathrm{DO} = \frac{1}{\mathrm{d}} A = \alpha B \tag{2}$$

Le suivi de la densité optique au cours du temps permet ainsi de suivre la croissance de la biomasse. La courbe de croissance standard représentée ci-dessous a été établie par Buchanan en 1918 [7]. C'est une représentation semi-logarithmique de sorte que la phase exponentielle notée **3** est linéaire.

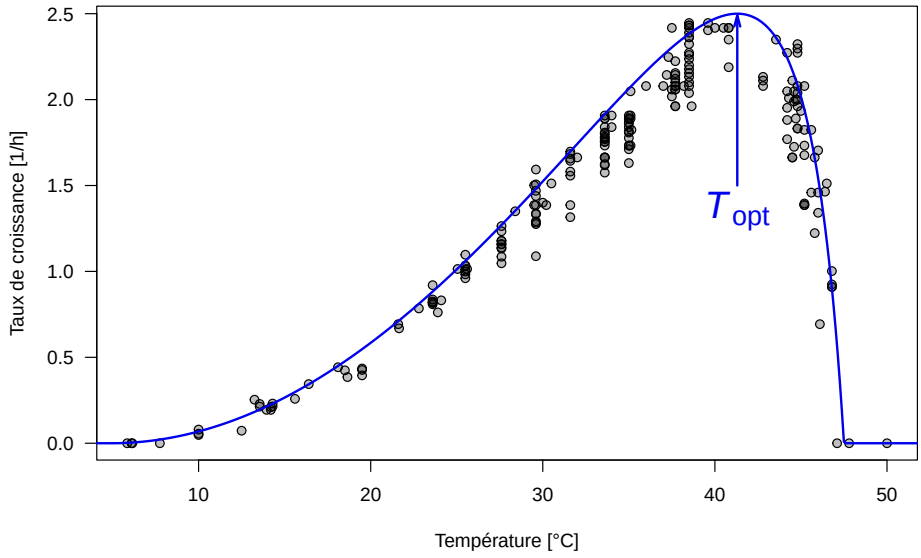


Depuis les travaux de Jacques Monod [22], la phase de croissance exponentielle est utilisée comme référence car c'est elle qui est la plus reproductible. La pente de la droite de la phase exponentielle dans la représentation semi-logarithmique est le taux de croissance, μ :

4. Voir par exemple [21, 29, 24, 17, 26, 8, 13, 12]

$$\mu = \frac{1}{B}\frac{dB}{dt} \tag{3}$$

Le taux de croissance est modulé par de nombreux facteurs environnementaux dont la température du milieu. La réponse du taux de croissance en fonction de la température est illustrée ci-après avec les données de Barber [3] de 1908 pour *Escherichia coli* et le lissage du modèle CTMI [23] [5]. On constate qu'il y a une *température optimale de croissance*, T_{opt}, décentrée vers la droite, ce qui donne un aspect asymétrique à la courbe de réponse.



C'est cette température optimale de croissance, T_{opt}, qui va jouer le rôle ici de variable à prédire, Y, chez 730 espèces bactériennes [6]. Mais en fonction de quelles variables prédictives ? Nous allons utiliser les fréquences en acides-aminés dans les protéomes de ces espèces, les données sont extraites de [16], les prétraitements effectués sont expliqués par ailleurs [7]. Les importer dans R :

```
chmin <- "https://pbil.univ-lyon1.fr/R/donnees/FAS/"
load(url(paste0(chmin, "tdr411.Rda")))
length(Y)
```

```
[1] 730
```

```
dim(X)
```

```
[1] 730  19
```

```
colnames(X)
```

5. C'est une fraction rationnelle définie par :

$$\mu(T) = \begin{cases} 0 & \text{si } T \notin [T_{\min}, T_{\max}] \\ \frac{\mu_{\text{opt}}(T-T_{\max})(T-T_{\min})^2}{(T_{\text{opt}}-T_{\min})[(T_{\text{opt}}-T_{\min})(T-T_{\text{opt}})-(T_{\text{opt}}-T_{\max})(T_{\text{opt}}+T_{\min}-2T)]} & \text{si } T \in [T_{\min}, T_{\max}] \end{cases}$$

6. *sensu lato* : Haeckel (1874)

7. Retrait de deux « outliers », *Eubacterium acidaminophilum* et *Cenarchaeum symbiosum*, ayant une composition particulière de leur protéome [15, 16]. Censure des psychrophiles trop peu documentés. Retrait du tryptophane pour avoir 19 variables indépendantes. Voir détails à `https://pbil.univ-lyon1.fr/R/pdf/tdr411.pdf`

```
 [1] "Ala" "Arg" "Asn" "Asp" "Cys" "Gln" "Glu" "Gly" "His" "Ile" "Leu" "Lys" "Met"
[14] "Phe" "Pro" "Ser" "Thr" "Tyr" "Val"
```

```
unique(mycol)
```

```
[1] "palegreen2" "red"        "orange"
```

Le vecteur `Y` contient les températures optimales de croissance de 730 espèces bactériennes en degrés Celcius. La table `X` contient les fréquences en pourcentage de 19 acides aminés des protéomes de ces mêmes espèces désignés par leur code à trois lettres (`Ala` pour l'alanine par exemple). Le vecteur `mycol` donne le codage couleur des classes de thermophilie [8] (mésophiles en `palegreen2`, thermophiles `orange` et hyperthermophiles en `red`).

Quand on fait de la régression linéaire simple, la variable quantitative dépendante Y est une fonction linéaire de la variable quantitative prédictive X plus du bruit :

$$Y = \beta_0 + \beta_1 X + \epsilon \tag{4}$$

Quand on fait de la régression linéaire multiple, on a plusieurs variables prédictives $X_1, X_2, \ldots, X_p \equiv \mathbf{X}$. En notant $\beta = (\beta_1, \beta_2, \ldots, \beta_p)^\top$ on a :

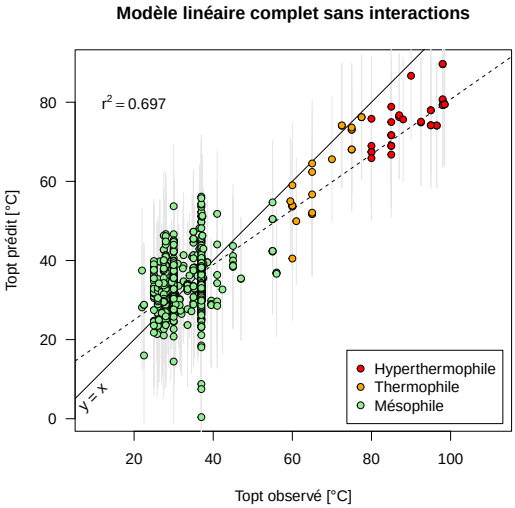
$$Y = \beta_0 + \mathbf{X}\beta + \epsilon \tag{5}$$

Pour visualiser le jeu de données on représente le résultat de modèle linéaire complet sans interactions. Notez qu'il se note de façon particulièrement compacte dans R avec simplement « `Y ~ .` » où le point est un alias pour désigner l'ensemble des variables prédictives [9].

```
lm1 <- lm(Y ~ . , data = X)
suppressWarnings(CI <- predict(lm1, interval = "predict"))
myplot <- function(ypred, main, pxy = c(10, 5), pl = "bottomright",
                   pr2 = c(20, 80), CI = NULL){
  plot.new() ; plot.window(xlim = range(Y), ylim = range(ypred), asp = 1)
  box() ; axis(1) ; axis(2, las = 1)
  if(!is.null(CI)) segments(Y, CI[,1], Y, CI[,2], col = grey(0.9))
  title(main = main, xlab = "Topt observé [°C]", ylab = "Topt prédit [°C]")
  abline(lm(ypred~Y), lty = 2)
  abline(c(0, 1))
  text(pxy[1], pxy[2], "y = x", srt = 45)
  legend(pl, inset = 0.02, legend = c("Hyperthermophile",
    "Thermophile", "Mésophile"), pch = 21,
    pt.bg = c("red", "orange", "palegreen2"))
  r2 <- signif(1 - var(Y - ypred)/var(Y), 3)
  text(pr2[1], pr2[2], bquote(r^2 == .(r2)))
  points(Y, ypred, pch = 21, bg = mycol)
}
myplot(lm1$fitted.values,
  main = "Modèle linéaire complet sans interactions", CI = CI[,2:3])
```

8. La discrétisation d'une variable continue est toujours un peu arbitraire, on reprend ici celle de [15] qui correspond pour fixer les idées à $T_{\text{opt}} \approx 40$°C chez les mésophiles, $T_{\text{opt}} \approx 60$°C chez les thermophiles et $T_{\text{opt}} \approx 80$°C chez les hyperthermophiles.

9. C'est quand même plus compact que `Y ~ Ala + Arg + Asn + Asp + Cys + Gln + Glu + Gly + His + Ile + Leu + Lys + Met + Phe + Pro + Ser + Thr + Tyr + Val`.



Le modèle n'est pas très bon (malgré un $r^2 = 0.697$) puisqu'il a tendance à sous-estimer pour les thermophiles et hyperthermophiles. On peut faire mieux ($r^2 = 0.912$ voir la figure 1 page 7) en ajoutant les interactions :

$$Y = \beta_0 + \mathbf{X}\beta + \mathbf{X}^\top\mathbf{X}\gamma + \epsilon \tag{6}$$

Mais le problème ici est que le nombre de paramètres croit de façon quadratique avec le nombre p de variables explicatives, il faut alors utiliser des techniques de sélection de variables, mais ce n'est pas l'objet de cette fiche. Disons simplement que la figure 1 page 7 donne une idée de ce que l'on peut obtenir avec des approches traditionnelles de régression linaire.

1.3 Application pratique à LUCA

L'acronyme LUCA pour *Last Universal Common Ancestor* est un peu pléonastique, mais référence oblige à *Lucy*, on ne prendra pas ombre de cette licence poétique. Avec des méthodes de reconstitution phylogénétiques, on peut tenter d'inférer la composition du protéome du dernier ancêtre commun à toutes les formes de vie sur terre [6]. D'où une application amusante si on possède un modèle prédisant T_{opt} à partir des fréquences des acides aminés : essayer de prédire la température à laquelle vivait LUCA. En pratique, les résultats sont plutôt décevants, avec le modèle de la figure 1 page 7 on arrive à des prédictions du type $T_{opt} = 99.4 \pm 19.5$ °C, c'est-à-dire d'une extrême imprécision.

2 Arbres de décision

Les CART [5] (*Classification And Regression Tree*) sont des arbres de régression quand la variable Y est quantitative (on cherche à prédire T_{opt}) et des arbres de classification quand la variable Y est qualitative (on cherche à prédire la classe de thermophilie). Pour qu'il n'y ait pas d'ambiguïté, on notera `Yr` quand Y est utilisée pour faire de la régression et `Yc` quand Y est utilisée pour

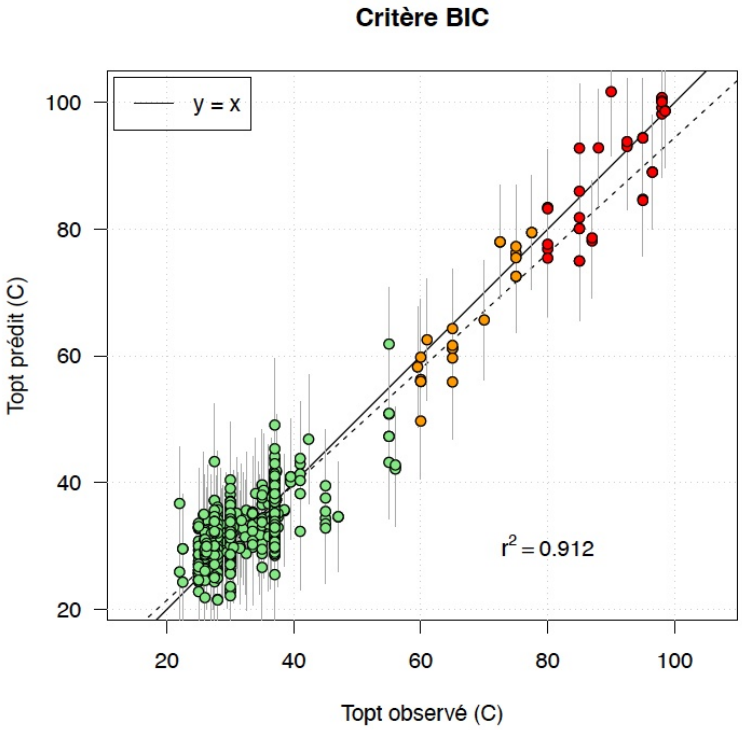


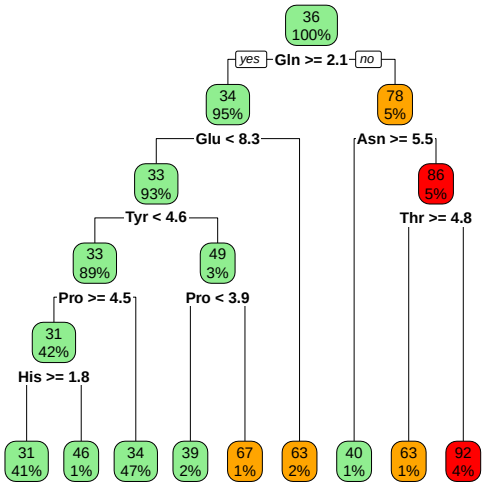
FIGURE 1 – Le modèle de régression linéaire avec toutes les interactions et sélection de variables selon le critère BIC [25]. Détails à `https://pbil.univ-lyon1.fr/R/pdf/tdr411.pdf`

faire de la classification. Les variables explicatives X, les fréquences des acides aminés, seront notées dans tous les cas `X`.

2.1 Arbres de régression

On utilise la fonction `rpart()` du paquet éponyme [27] avec les valeurs par défaut des paramètres pour le calculer. On utilise la fonction `rpart.plot()` du paquet éponyme [20] pour le représenter.

```
library(rpart) ; library(rpart.plot) ; Yr <- Y
Yr.rpart <- rpart(Yr ~ ., X) ; yval <- Yr.rpart$frame$yval
colphile <- rep("orange", nrow(Yr.rpart$frame))
colphile[yval < 59] <- "palegreen2"
colphile[yval >= 80] <- "red"
rpart.plot(Yr.rpart, box.col = colphile)
```



Les couleurs des boites reprennent la convention de couleurs déjà utilisée pour les classes de thermophilie. Les pourcentages donnent la proportion d'individus qui se trouvent à chaque nœud ou feuille de l'arbre : on part de 100 % à la racine en haut puis les individus sont séparés progressivement jusqu'aux feuilles en bas. La valeur au dessus du pourcentage est la moyenne de T_{opt} pour les individus du nœud ou de la feuille. On peut noter que les feuilles sont plus ou moins peuplées, celles à 31 °C et 34 °C représentent à elles seules 88 % des individus [10]. Les critères de classification sont indiqués sous les boites des nœuds.

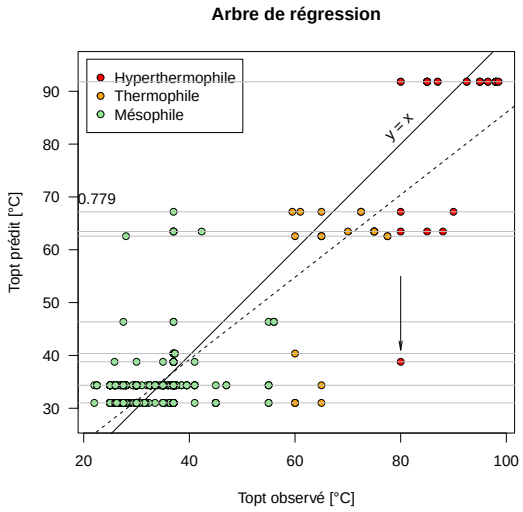
Le gros intérêt des arbres de régression et qu'ils sont très lisibles ce qui facilite grandement l'interprétation. Par exemple, on remarque ici que `Gln` et `Asn` interviennent très haut dans l'arbre de décision. Or, `Gln` et `Asn` sont les deux seuls acides aminés à avoir une liaison peptidique dans leur chaîne latérale. Cette liaison est connue pour être sensible à l'hydrolyse conduisant aux réactions `Gln` → `Glu` et `Asn` → `Asp`, soit à la production d'acides aminés chargés pouvant gravement affecter la fonction des protéines, ne serait-ce qu'en altérant leurs points

10. Il y a un gros biais d'échantillonnage en faveur des mésophiles dans le jeu de données parce que les pathogènes, très étudiés, font parti de cette classe

iso-électriques. Comme cette réaction est plus rapide à haute température, on peut concevoir qu'il y ait une pression de sélection pour diminuer la fréquence en `Gln` et `Asn` chez les espèces thermophiles. Bref, les arbres de régression se prêtent bien à une lecture interprétative.

On confronte maintenant, comme pour le modèle linéaire précédent, les valeurs observées aux valeurs prédites par l'arbre de régression.

```
myplot(predict(Yr.rpart), main = "Arbre de régression", pxy = c(80, 83),
       pl = "topleft", pr2 = c(20, 70))
with(subset(Yr.rpart$frame, var == "<leaf>"), abline(h = yval, col = grey(0.75)))
arrows(80, 55, 80, 41, angle = 10, le = 0.1)
```



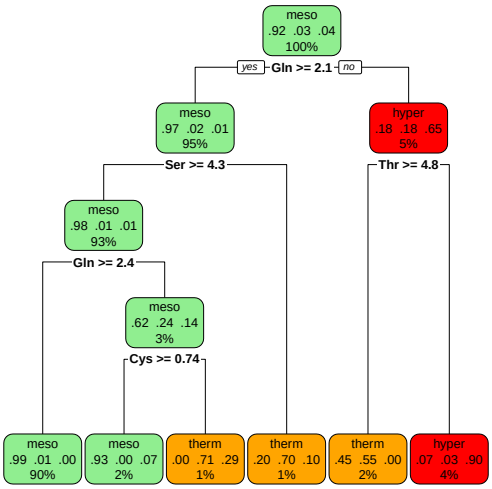
La démarche est très différente de celle du modèle linéaire. Dans le modèle linéaire, on suit une approche *globale* où toutes les données sont prises en compte. Dans l'arbre de régression, on a une démarche *locale* : les individus sont regroupés en classes (les feuilles de l'arbre) et pour chaque classe, on prédit la même valeur qui n'est rien d'autre que la moyenne. Ceci est mis en évidence ici par les neuf lignes grises horizontales correspondant aux neuf feuilles de l'arbre de régression et dont l'ordonnée est égale à la moyenne des T_{opt} des bactéries de la feuille. Globalement si on compare au modèle linéaire ce n'est pas mauvais ($r^2 = 0.779$) contre ($r^2 = 0.697$), mais on voit qu'il y a des points mal classés, avec en particulier un hyperthermophile regroupé avec des mésophiles (flèche). Pour améliorer les choses, il faudrait augmenter le nombre de feuilles, mais on risque alors de tomber dans un problème de sur-apprentissage : avec un nombre de feuilles égal au nombre de valeurs distinctes, on ferait des prédictions parfaites, mais on aurait pas vraiment synthétisé les données.

Une limitation intrinsèque des arbres de régression est que l'on ne pourra pas les utiliser pour faire de l'extrapolation. Par construction les valeurs prédites ne peuvent pas sortir de la gamme des valeurs observées, donc des valeurs de T_{opt} comprises entre 22.0 °C et 98.5 °C. Cela peut être un problème ici puisqu'avec le modèle linéaire (figure 1 page 7) on prédisait une valeur de 99.4 °C pour LUCA, ce qui est une extrapolation par rapport à la gamme des valeurs disponibles dans le jeu de données.

2.2 Arbres de classification

On ne cherche plus ici à prédire le T_{opt} des bactéries mais simplement leur classe de thermophilie (*i.e.* mésophile, thermophile et hyperthermophile). On a intérêt à expliciter qu'il s'agit d'une variable qualitative *ordonnée* pour faciliter la lecture de l'arbre de classification.

```
Yc <- as.factor(mycol)
levels(Yc) <- c("therm", "meso", "hyper")
Yc <- factor(Yc, levels = c("meso", "therm", "hyper"), ordered = TRUE)
Yc.rpart <- rpart(Yc ~ ., X)
rpart.plot(Yc.rpart, box.col = c("palegreen2", "orange", "red")[Yc.rpart$frame$yval])
```



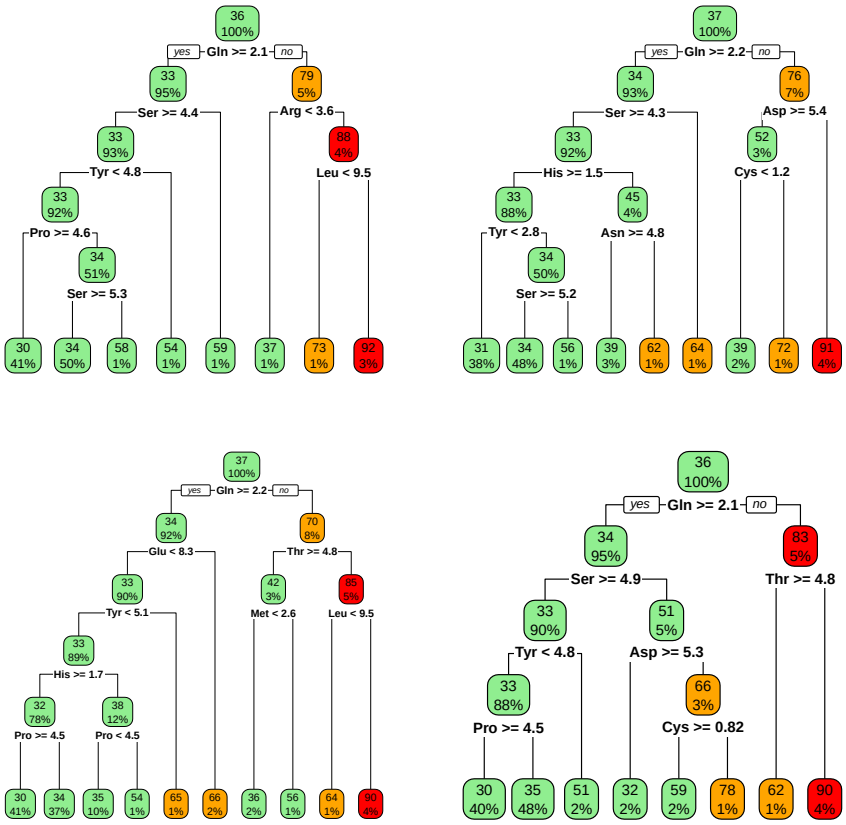
On retrouve le poids de `Gln` qui intervient sur deux nœuds de classification. Le pourcentage représente comme pour l'arbre de régression le nombre de bactéries présentes à chaque nœud ou feuille. Les trois valeurs au centre des boites sont les fractions de chaque classe. La couleur des boites est celle de la classe majoritaire.

2.3 Instabilité des arbres de décision

Un problème bien connu des arbres de décision est leur extrême instabilité, dans le sens où une faible perturbation du jeu de données peut modifier profondément les résultats. Nous pouvons facilement illustrer cet aspect en tirant avec remise un échantillon *bootstrap* des données de départ. Comme on peut le constater, les résultats sont très variables.

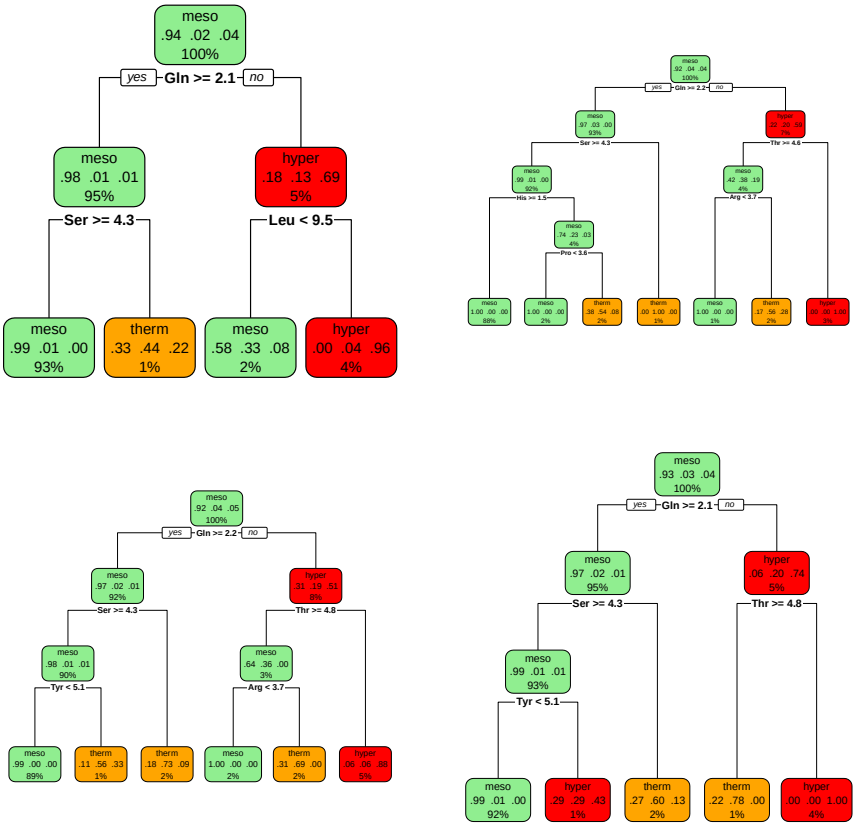
```
par(mfrow = c(2, 2), oma = c(0, 0, 1, 0))
for(i in seq_len(4)){
  set.seed(i)
  isample <- sample(1:nrow(X), nrow(X), replace = TRUE)
  X2 <- X[isample, ] ; Yr2 <- Yr[isample]
  Yr2.rpart <- rpart(Yr2 ~ ., X2) ; yval <- Yr2.rpart$frame$yval
  colphile <- rep("orange", nrow(Yr2.rpart$frame))
  colphile[yval < 59] <- "palegreen2"
  colphile[yval >= 80] <- "red"
  rpart.plot(Yr2.rpart, box.col = colphile)
}
title(main = "Instabilité des arbres de régression", outer = TRUE)
```

Instabilité des arbres de régression




```
par(mfrow = c(2, 2), oma = c(0, 0, 1, 0))
for(i in seq_len(4)){
  set.seed(i)
  isample <- sample(1:nrow(X), nrow(X), replace = TRUE)
  X2 <- X[isample, ] ; Yc2 <- Yc[isample]
  Yc2.rpart <- rpart(Yc2 ~ ., X2) ; yval <- Yc2.rpart$frame$yval
  rpart.plot(Yc2.rpart, box.col = c("palegreen2", "orange", "red")[yval])
}
title(main = "Instabilité des arbres de classification", outer = TRUE)
```

Instabilité des arbres de classification



La glutamine `Gln` apparaît bien toujours à la racine des arbres de régression mais `Asn` ne figure plus qu'une fois. De plus `Asn` n'apparaît jamais dans les arbres de classification. Tout ceci met à mal l'interprétation couci-couça que nous avions avancée. Mais si au lieu de 4 échantillons *bootstrap* nous en faisions beaucoup plus pour voir si quelques îlots de stabilité n'émergent pas du lot ? C'est un peu l'idée des forêts aléatoires, mais en encore plus fou que ça.

3 Forêts aléatoires

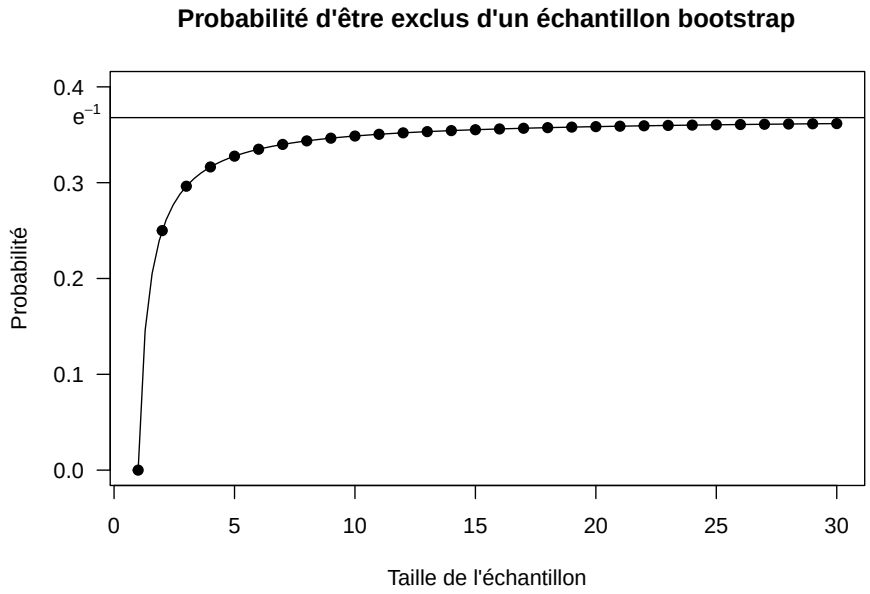
C'est pour pallier l'inconvénient de l'instabilité des arbres de décision que les forêts aléatoires ont été introduites [4]. Au lieu d'utiliser un seul arbre de décision, on va utiliser une collection d'arbres, d'où le terme de « forêt. » La première difficulté vient de ce que l'on utilise des arbres maximaux, c'est-à-dire avec autant de feuilles qu'il y a d'éléments distincts dans Y, soit 66 dans notre cas. Si un arbre à 9 feuilles comme nous avions précédemment se lit facilement, un arbre à 66 feuilles est beaucoup plus touffu et donc plus difficile à appréhender. Mais le pire c'est que nous utilisons non pas 1 mais typiquement 500 arbres de décision pour calculer une réponse moyenne. C'est impossible à visualiser, c'est une véritable boite noire.

Sans entrer dans les détails, je renvoie à la très didactique présentation [9] pour plus d'information, il y a deux niveaux d'aléa utilisés. On fait du *bootstrat* classique en ré-échantillonnant avec remise dans Y, cela revient à pondérer aléatoirement plus ou moins les observations, chaque arbre de décision aura sa propre vision, partielle, du jeu de données. Mais, et c'est là que c'est encore plus fou, les arbres de décision auront également une vision partiale des variables explicatives en tirant au hasard, sans remise, un sous-ensemble des colonnes de X. Et c'est en se fiant au jugement moyen de cette collection de décideurs aux jugements partiels et partiaux que l'on espère améliorer les choses, bigre !

Un aspect très intéressant des forêts aléatoires est qu'elles possèdent en quelque sorte un contrôle de qualité interne [11]. Quand on tire un échantillon *bootstrap* avec remise, à chaque tirage une observation Y_i a une probabilité $\frac{n-1}{n}$ de ne pas être retenue. On en déduit que pour de grand échantillons il y aura en moyenne environ 40 % des observations qui seront exclues d'un échantillon *bootstrap* parce que :

$$\lim_{n\to+\infty}\left(\frac{n-1}{n}\right)^n = \lim_{n\to+\infty} e^{n\ln\left(1-\frac{1}{n}\right)} = \lim_{n\to+\infty} e^{\frac{\ln\left(1-\frac{1}{n}\right)}{\frac{1}{n}}} = \lim_{n\to+\infty} e^{\frac{\left(\frac{1}{1-\frac{1}{n}}\right)\frac{1}{n^2}}{\frac{-1}{n^2}}} = e^{-1}$$

En pratique, la convergence est très rapide, et pour des échantillons de taille raisonnable [12], disons $n \geq 30$, on est à une épaisseur de trait de l'asymptote :



Plutôt que de jeter à la poubelle les points qui n'ont pas été retenus dans l'échantillon *bootstrap*, les forêts aléatoires les recyclent pour apprécier les performances de l'arbre de décision sur ces données qu'il n'a jamais vues. On apprécie ainsi sa capacité de généralisation.

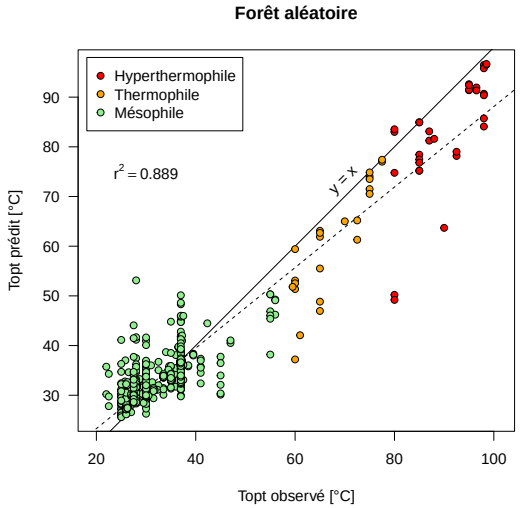
11. Dans le jargon des forêts aléatoires, on parle d'erreur OOB pour *Out Of Bagging*.
12. Et ma collègue de commenter en lisant ce passage : « c'est papi qui parle ! » Chère Anne, ton impertinence malicieuse va beaucoup nous manquer…

Un autre aspect tout aussi intéressant des forêts aléatoires est qu'elle permettent de classer les variables explicatives par ordre d'importance décroissante. L'idée est d'utiliser un échantillon *bootstrap* perturbé en mélangeant la j^{e} colonne de X pour apprécier l'importance de la j^{e} variable. Si cette permutation n'affecte que peu les performances de l'arbre de décision, on conviendra que c'est une variable de faible importance.

3.1 Régression

Nous utilisons ici la fonction `randomForest()` du paquet éponyme [14]. Pour la reproductibilité des résultats, on fixe la graine du générateur de nombres pseudo-aléatoires avec `set.seed(1)`.

```
library(randomForest)
set.seed(1)
Yr.RF <- randomForest(X, Yr, importance = TRUE)
myplot(Yr.RF$predicted, "Forêt aléatoire", px = c(70, 73), pl = "topleft", pr2 = c(30, 75))
```



Cela peut sembler à peine croyable, mais en se fiant à la *vox populi* d'une meute de décideurs borgnes (vision partielle des données) ayant une main attachée dans le dos (accès à une fraction des variables explicatives) on a des résultats tout à fait comparables avec ceux obtenus avec le modèle linéaire avec interactions (figure 1 page 7). Donc, oui, notre boite noire fonctionne bien. D'une façon générale, les performances empiriques des forêts aléatoires sont exceptionnelles, même si on ne comprend pas encore très bien pourquoi [9].

3.2 Classification

```
set.seed(1)
Yc.RF <- randomForest(X, Yc, importance = TRUE)
Yc.RF
```

```
Call:
 randomForest(x = X, y = Yc, importance = TRUE)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 4
        OOB estimate of  error rate: 1.64%
```

```
Confusion matrix:
      meso therm hyper class.error
meso   675     0     0   0.0000000
therm    8    15     0   0.3478261
hyper    3     1    28   0.1250000
```

La matrice de confusion montre que la classification est très bonne. Les 675 bactéries prédites comme mésophiles le sont toutes. Il reste 8 mésophiles dans celles prédites comme thermophiles et 3 dans celles prédites comme hyperthermophiles.

4 Sélection de variables avec des forêts aléatoires

Nous allons utiliser ici la méthode [10] la fonction `VSURF()` du paquet éponyme [11]. Pour sélectionner les variables, `VSURF()` procède en trois étapes. Dans un premier temps, les variables vont être classées par importance décroissante [13] en faisant la moyenne de l'importance des variables sur `nfor.thres` [14] forêts aléatoires. Ce ne sont donc plus des forêts aléatoires mais carrément des massifs forestier aléatoires ! Dans un deuxième et troisième temps, les variables vont être sélectionnées en distinguant si l'objectif est de les interpréter ou bien de faire des prédictions.

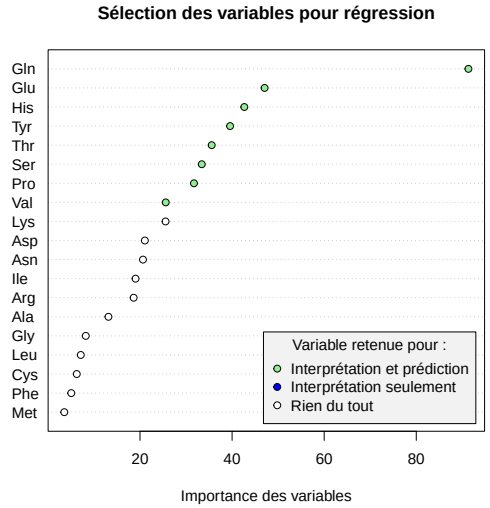
4.1 Régression

Les temps de calculs sont tout à fait raisonnables, sur mon ordinateur portable, cela prend 3 minutes et demi, descend à 1 minute si j'active l'option `parallel = TRUE` et même à 23 secondes en faisant appel au paquet `ranger` [30].

```
library(VSURF)
set.seed(1)
Yr.VSURF <- VSURF(X, Yr,
  nfor.thres = 50, nfor.interp = 25, nfor.pred = 25,
  parallel = TRUE, clusterType = "ranger", RFimplem = "ranger")
save(Yr.VSURF, file = "FAS/YrVSURF.Rda")
```

```
load(url(paste0(chmin, "YrVSURF.Rda")))
with(Yr.VSURF, {
  mycol <- rep("white", length(imp.mean.dec))
  mycol[varselect.interp] <- "blue"
  mycol[varselect.pred] <- "palegreen2"
  dotchart(rev(imp.mean.dec),
    labels = names(X)[rev(imp.mean.dec.ind)], pch = 21,
    main = "Sélection des variables pour régression",
    bg = mycol[rev(imp.mean.dec.ind)],
    xlab = "Importance des variables")
})
legend("bottomright", inset = 0.02,
  legend = c("Interprétation et prédiction",
             "Interprétation seulement", "Rien du tout"),
  pch = 21, pt.bg = c("palegreen2", "blue", "white"), bg = grey(0.95),
  title = "Variable retenue pour :")
```

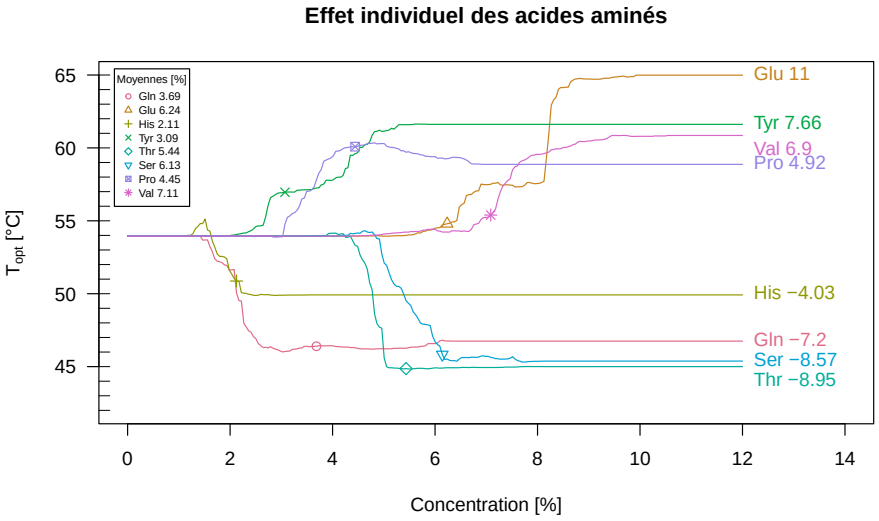
13. Et éventuellement éliminées quand il y a beaucoup plus de variables explicatives que de valeurs observées, mais ce n'est pas le cas ici avec 19 variables explicatives et 730 observations.
14. 20 par défaut dans la version 1.2.0 du paquet `VSURF` mais la valeur typique de 50 est donnée dans [9].



La glutamine `Gln` arrive largement en tête. Ici les variables retenues pour l'interprétation et la prédiction sont les mêmes. Il y a donc 8 variables qui ont été sélectionnées ici.

Pour comprendre dans quel sens jouent les variables explicatives retenues, j'ai regardé quels étaient les T_{opt} prédits pour des homopolymères de concentration croissante en un acide aminé. Cela n'a bien entendu aucun sens physique, c'est juste pour illustrer l'effet individuel des acides aminés retenus.

```
library(ranger)
Xinterp <- X[ , Yr.VSURF$varselect.interp]
set.seed(1)
Yr.interp <- ranger(Yr ~ ., Xinterp, num.tree = 2500)
nullprot <- rep(0, ncol(Xinterp))
maxaa <- 12 ; aaseq <- seq(0, maxaa, le = 255)
plot.new() ; plot.window(xlim = c(0, maxaa + 2), ylim = c(42, 65))
box() ; axis(1) ; axis(2, las = 1)
cols <- hcl.colors(ncol(Xinterp), "Dark 3")
pchs <- 1:ncol(Xinterp)
for(j in seq_len(ncol(Xinterp))){
  homoprot <- matrix(nullprot, nrow = length(aaseq), ncol = ncol(Xinterp))
  homoprot[ , j] <- aaseq ; colnames(homoprot) <- colnames(Xinterp)
  pred <- predict(Yr.interp, homoprot)$predictions
  lines(aaseq, pred, col = cols[j])
  imean <- which.min((aaseq - mean(Xinterp[ , j]))^2)
  points(aaseq[imean], pred[imean], pch = pchs[j], col = cols[j])
  txt <- colnames(Xinterp[j])
  delta <- ifelse(txt %in% c("Val", "Thr"), -1, 0)
  txt <- paste(txt, signif(pred[length(pred)] - pred[1], 3))
  text(maxaa, delta + pred[length(pred)], txt, col = cols[j], pos = 4, xpd = NA)
}
legend("topleft", inset = 0.02,
  legend = paste(colnames(Xinterp), signif(colMeans(Xinterp), 3)),
  pch = pchs, col = cols, title = "Moyennes [%]", cex = 0.6)
title(main = "Effet individuel des acides aminés",
      ylab = expression(paste(T[opt], " [°C]")),
      xlab = "Concentration [%]")
rug(40:70, side = 2)
```

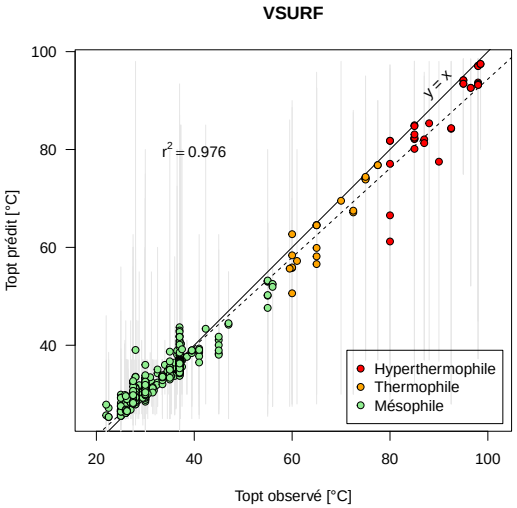


Examinons le cas de `Gln`. Nous avons une réponse de forme approximativement sigmoïdale décroissante. Pour les faibles concentrations, moins de 1 %, nous avons un T_{opt} de 54 °C, et pour les fortes concentrations, plus de 6 %, un T_{opt} de 47 °C. La transition entre les deux se fait au voisinage de l'abscisse du point d'inflexion, environ 2 %, soit moins que la concentration moyenne (3.7 %) en `Gln` dans notre jeu de données. On s'attend donc à ce qu'il y ait un évitement de `Gln` chez les thermophiles, ce qui est bien le cas (*cf.* section 5.3 page 20).

Ce graphique permet d'avoir une intuition du fonctionnement de la forêt aléatoire : chaque arbre de régression a une réponse en escalier mais avec un seuil variable qui gambade entre 1 et 6 % selon les arbres. En faisant la moyenne de toutes ces réponses en escalier, on obtient une courbe de réponse lissée, à l'air globalement sigmoïdal. Notez que la courbe de réponse peut être plus subtile en n'étant pas monotone pour certains acides aminés, par exemple pour `Glu`.

Pour faire des prédictions, il nous suffit de faire une forêt aléatoire avec les variables sélectionnées. Je passe par la fonction `ranger()` parce qu'elle implémente une méthode [19] pour estimer la précision des prédictions.

```
library(ranger)
Xpred <- X[ , Yr.VSURF$varselect.pred]
set.seed(1)
Yr.ranger <- ranger(Yr ~ ., Xpred, quantreg = TRUE)
CI <- predict(Yr.ranger, Xpred, type = "quantiles", quantiles = c(0.025, 0.975))$predictions
myplot(predict(Yr.ranger, Xpred)$predictions,
  main = "VSURF", CI = CI, pxy = c(90, 93), pr2 = c(40, 80))
```

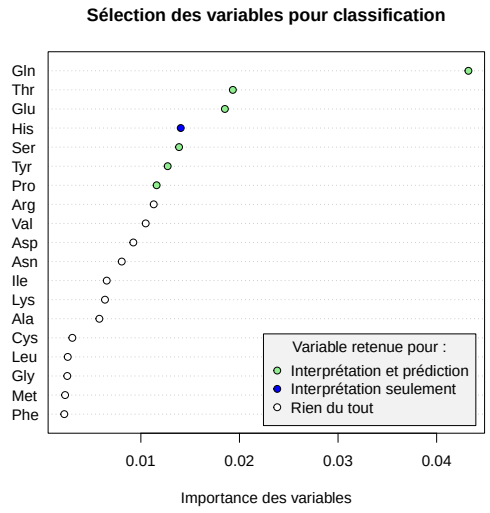


Le résultat est assez impressionnant ($r^2 = 0.976$, ce n'est pas courant) si on le compare avec celui de la régression linéaire classique (figure 1 page 7) : on améliore considérablement la qualité des prédictions, en particulier chez les mésophiles. Il y a quand même deux hyperthermophiles assez mal prédits, et on a beaucoup perdu sur la précision des prédictions.

4.2 Classification

```
set.seed(1)
Yc.VSURF <- VSURF(X, Yc,
  nfor.thres = 50, nfor.interp = 25, nfor.pred = 25,
  parallel = TRUE, clusterType = "ranger", RFimplem = "ranger")
save(Yc.VSURF, file = "FAS/YcVSURF.Rda")
```

```
load(url(paste0(chmin, "YcVSURF.Rda")))
with(Yc.VSURF, {
  mycol <- rep("white", length(imp.mean.dec))
  mycol[varselect.interp] <- "blue"
  mycol[varselect.pred] <- "palegreen2"
  dotchart(rev(imp.mean.dec),
    labels = names(X)[rev(imp.mean.dec.ind)], pch = 21,
    main = "Sélection des variables pour classification",
    bg = mycol[rev(imp.mean.dec.ind)],
    xlab = "Importance des variables")
})
legend("bottomright", inset = 0.02,
  legend = c("Interprétation et prédiction",
             "Interprétation seulement", "Rien du tout"),
  pch = 21, pt.bg = c("palegreen2", "blue", "white"), bg = grey(0.95),
  title = "Variable retenue pour :")
```



La glutamine arrive toujours en tête. Il y a 7 variables retenues pour l'interprétation mais plus que 6 pour la prédiction. On retrouve les mêmes variables que pour la régression, sauf la valine qui n'a pas été retenue ici.

```
library(ranger)
Xpred <- X[ , Yc.VSURF$varselect.pred]
set.seed(1)
Yc.ranger <- ranger(Yc ~ ., Xpred)
Yc.ranger$confusion.matrix
```

```
      predicted
true   meso therm hyper
  meso  675     0     0
  therm   5    17     1
  hyper   2     2    28
```

Par rapport à la forêt aléatoire, on a amélioré légèrement les choses pour les thermophiles avec 17 bien prédits sur 23 contre 15 auparavant (section 3.2 page 14).

5 Application à LUCA

5.1 Régression

```
load(url(paste0(chmin, "brooks.Rda")))
LUCA <- as.data.frame(t(brooks[-18, "LUA", drop = FALSE]))*100
stopifnot(all(names(LUCA) == names(X)))
predict(Yr.ranger, LUCA[Yr.VSURF$varselect.pred])$predictions
```

```
[1] 74.4682
```

```
predict(Yr.ranger, LUCA[Yr.VSURF$varselect.pred], type = "quantiles",
  quantiles = c(0.025, 0.975))$predictions
```

```
     quantile= 0.025 quantile= 0.975
[1,]              28              98
```

Les résultats sont assez différents de ceux obtenus avec le modèle linéaire, avec $T_{\text{opt}} \approx 74.5$ °C, LUCA serait plus un thermophile qu'un hyperthermophile, mais vu que l'on se ballade entre 28°C et 98°C, on ne peut pas en tirer grand chose avec certitude.

5.2 Classification

```
predict(Yc.ranger, LUCA[Yc.VSURF$varselect.pred])$predictions
```

```
[1] hyper
Levels: meso therm hyper
```

```
set.seed(1)
Yc.ranger <- ranger(Yc ~ ., Xpred, probability = TRUE)
predict(Yc.ranger, LUCA[Yc.VSURF$varselect.pred])$predictions
```

```
          meso     therm  hyper
[1,] 0.1354905 0.1670095 0.6975
```

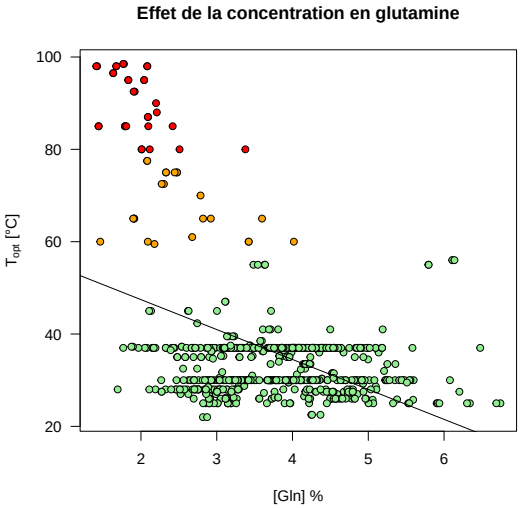
Cette fois LUCA est prédit comme un hyperthermophile, mais là encore sans grande certitude puisqu'il y a quand même 30 % de chances qu'il ne le soit pas, et même la mésophilie n'est pas complètement exclue.

5.3 Conclusion

En terme de prédiction, l'utilisation de forêts aléatoires pour sélectionner des variables conduit à des résultats assez bluffants si on compare le graphique de la section 4.1 page 15 avec celui de la régression linéaire (figure 1 page 7). Mais il y a un prix à payer : on perd toute capacité à extrapoler (ce qui n'est pas forcément un mal) et la précision des prédictions n'est pas au rendez-vous.

En terme d'interprétation, l'utilisation de forêts aléatoires pour sélectionner des variables met en vedette la glutamine. Retournons aux données de départ pour voir ce qu'il en est.

```
iGln <- which(colnames(X) == "Gln")
plot(X[ , iGln], Y, pch = 21, bg = mycol, las = 1,
  xlab = "[Gln] %", ylab = expression(paste(T[opt], " [°C]")),
  main = "Effet de la concentration en glutamine")
abline(lm(Y~X[ , iGln]))
```



On voit qu'il y a indubitablement un lien entre la concentration en glutamine et T_{opt} : elle est faible chez les hyperthermophiles (2 %) et les thermophiles (2.5 %), mais couvre une large gamme chez les mésophiles. Ce genre de relation « triangulaire » est très mal pris en compte par le modèle linéaire alors qu'avec l'approche locale des forêt aléatoires cela ne pose pas de problème. L'utilisation de forêts aléatoires à des fins de sélection de variables dans un but interprétatif présente donc un intérêt certain.

Références

[1] Anonymous. SYTRAL MOBILITÉS - Le financement des transports collectifs urbains et le service rendu à l'usager (métropole de LYON) Exercices 2015 et suivants. Technical report, Chambre régionale des comptes AUVERGNE-RHÔNE-ALPES, 2025.

[2] H. Balzter, B. Cole, C. Thiel, and C. Schmullius. Mapping CORINE land cover from Sentinel-1A SAR and SRTM digital elevation model data using random forests. *Remote Sensing*, 7(11) :14876–14898, 2015.

[3] M.A. Barber. The rate of multiplication of *Bacillus coli* at different temperatures. *Journal of Infectious diseases*, 5 :379–400, 1908.

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

[5] L. Breiman, J. Friedman, R.A Olshen, and C.J. Stone. *Classification and regression trees*. Chapman & Hall, New York, USA, 1984.

[6] D.J. Brooks, J.R. Fresco, and M Singh. A novel method for estimating ancestral amino acid composition and its application to proteins of the Last Universal Ancestor. *Bioinformatics*, 20 :2251–2257, 2004.

[7] R.E. Buchanan. Life phases in a bacterial culture. *Journal of Infectious Diseases*, 23 :109–125, 1918.

[8] A.C.R. Dean and P.L. Rogers. The cell size and macromolecular composition of *Aerobacter aerogenes* in various sytems of continuous culture. *Biochimica Biophysica Acta*, 148 :267–279, 1967.

[9] R. Genuer and J.-M. Poggi. Arbres CART et Forêts aléatoires - Importance et sélection de variables. In M Maumy-Bertrand, G. Saporta, and C. Thomas-Agnan, editors, *Apprentissage statistique et données massives*. Edition TECHNIP, 5 avenue de la République, 75011 Paris, France, 2018.

[10] R. Genuer, J.-M. Poggi, and C. Tuleau-Malot. Variable selection using random forests. *Pattern recognition letters*, 31(14) :2225–2236, 2010.

[11] R. Genuer, J.-M. Poggi, and C. Tuleau-Malot. *VSURF : Variable Selection Using Random Forests*, 2022. R package version 1.2.0.

[12] C.J. Griffith and T.H. Melville. Growth of oral streptococci in a chemostat. *Archives of Oral Biology*, 19 :87–90, 1974.

[13] A.L. Koch. Turbidity measurement of bacterial cultures in some available commercial instruments. *Analytical Biohemistry*, 38 :252–259, 1970.

[14] A. Liaw and M. Wiener. Classification and regression by randomforest. *R News*, 2(3) :18–22, 2002.

[15] J.R. Lobry and D. Chessel. Internal correspondence analysis of codon and amino-acid usage in thermophilic bacteria. *Journal of Applied Genetics*, 44 :235–261, 2003.

[16] J.R. Lobry and A. Necşulea. Synonymous codon usage and its potential link with optimal growth temperature in prokaryotes. *Gene*, 385 :128–136, 2006.

[17] R. Luedeking and E.L. Piret. A kinetic study of the lactic acid fermentation. *Journal of Biochemical and Microbiological Technology and Engineering*, 1 :393–412, 1959.

[18] M Maumy-Bertrand, G. Saporta, and C. Thomas-Agnan. Avant-propos. In M Maumy-Bertrand, G. Saporta, and C. Thomas-Agnan, editors, *Apprentissage statistique et données massives*. Edition TECHNIP, 5 avenue de la République, 75011 Paris, France, 2018.

[19] N. Meinshausen and G. Ridgeway. Quantile regression forests. *Journal of machine learning research*, 7(6), 2006.

[20] S. Milborrow. *rpart.plot : Plot rpart Models : An Enhanced Version of plot.rpart*, 2024. R package version 3.1.2.

[21] J. Monod. *Recherches sur la croissance des cultures bactériennes*. PhD thesis, Paris, 1941.

[22] J. Monod. *Recherches sur la croissance des cultures bactériennes*. Herman, Paris, France, 1942.

[23] L. Rosso, J.R. Lobry, and J.-P. Flandrois. An unexpected correlation between cardinal temperatures of microbial growth highlighted by a new model. *Journal of Theoretical Biology*, 162(4) :447–463, 1993.

[24] M Schaechter, O. Maaløe, and N.O. Kleldgaard. Dependency on medium and temperature of cell size and chemical composition during balanced growth of *Salmonella typhimurium*. *Journal of General Microbiology*, 19 :592–606, 1958.

[25] G. Schwarz. Estimating the dimension of a model. *Annals of Statistcs*, 6 :461–464, 1978.

[26] J. Spaun. Problems in standardization of turbidity determinations of bacterial suspensions. *Bulletin of the World Health Organisation*, 26 :219–255, 1962.

[27] T. Therneau and B. Atkinson. *rpart : Recursive Partitioning and Regression Trees*, 2023. R package version 4.1.23.

[28] R. Thom. *Prédire n'est pas expliquer*. Flammarion, Paris, France, 1993.

[29] G. Toennies and D.L. Gallant. The relation between photometric turbidity and bacterial concentration. *Growth*, 13 :7–20, 1949.

[30] M.N. Wright and A. Ziegler. ranger : A fast implementation of random forests for high dimensional data in C++ and R. *Journal of Statistical Software*, 77(1) :1–17, 2017.