Fiche TD avec le logiciel R : Bataille

Bataille : modèle phénologique contre IA

Pr Jean R. Lobry

À ma gauche, une approche phénologique du XVIIIe siècle, ce bon vieux modèle de de Réaumur, toujours bon pied bon œil malgré son grand âge : né en 1735, bientôt 300 ans. À ma droite, le tout nouveau tout beau challenger IA avec ses forêts aléatoires du XXIe siècle : né en 2001, bientôt 30 ans. Ding ! Ding ! Ding ! Que le duel commence ! Aïe aïe aïe ! Victoire par KO du modèle phénologique dès le premier round...

Table des matières

1 Le ring (la lice)

1.1 Les données en sortie

Les données sont extraites de la thèse de Léa Keurinck [6]. Elles sont issues du traitement des données du feu RNSA[1] sur les quantités journalières de pollen de chêne présentes dans l'air pour le site de Bourg-en-Bresse pendant les années de 2008 à 2017. On cherche à prédire le début de la vague pollinique, calculée comme la date à laquelle la distribution de pollen annuelle atteint son quantile 10 % et notée `Day_of_Year` dans la table `Reauphe` :

```
chmin <- "https://esb.univ-lyon1.fr/donnees/Bataille/"
load(url(paste0(chmin, "Reauphe.Rda")))
Reauphe
```

```
   Year Day_of_Year
1  2008         114
2  2009         103
3  2010         111
4  2011          99
5  2012          97
6  2013         114
7  2014          98
8  2015         107
9  2016         106
10 2017          98
```

Les dates sont exprimées ici par le rang du jour dans l'année, ce qui est commode pour les traitements numériques, mais par forcément très intuitif pour un lecteur humain. On peut facilement traduire le rang du jour de l'année en quelque chose de plus lisible avec le paquet `lubridate` [5].

```
library(lubridate)
Reauphe$Date <- as.Date(sprintf("%d-01-01", Reauphe$Year))
yday(Reauphe$Date) <- Reauphe$Day_of_Year
Reauphe$JourMois <- with(Reauphe, paste(mday(Date), month(Date, label = TRUE)))
Reauphe[order(Reauphe$Day_of_Year), c("Year", "Day_of_Year", "JourMois")]
```

```
   Year Day_of_Year JourMois
5  2012          97    6 Apr
7  2014          98    8 Apr
10 2017          98    8 Apr
4  2011          99    9 Apr
2  2009         103   13 Apr
9  2016         106   15 Apr
8  2015         107   17 Apr
3  2010         111   21 Apr
1  2008         114   23 Apr
6  2013         114   24 Apr
```

Pour les 10 années de suivi à Bourg-en-Bresse, le début de la vague pollinique de chêne est donc régulièrement tombée au mois d'avril[2]. La phénologie est précisément l'étude de l'apparition de ce type d'évènements périodiques : on cherche à prédire le début de la vague pollinique.

1.2 Le juge de paix (l'arbitre)

Comme juge de paix, nous allons utiliser comme référence le modèle le plus simple que l'on puisse imaginer, à savoir celui qui résume les données par une simple valeur numérique ξ^*. Comment choisir la valeur de ξ^* idéale ? Nous

[1] Le réseau national de surveillance aérobiologique a été placé en liquidation judiciaire le 2025-03-26.

[2] Notons au passage que nous avons deux valeurs de `Day_of_Year` dupliquées : 98 le 8 avril 2014 et 2017 et 114 le 24 avril 2013 et le 23 avril 2008 (année bissextile).

allons utiliser le critère classique des moindres carrés. On considère un échantillon $\mathbf{x}$ de n éléments de terme générique x_i, soit $\mathbf{x} = (x_1, x_2, \ldots, x_i, \ldots x_n)$. On veut résumer cet échantillon par une seule valeur numérique ξ^* qui minimise la somme des carrés des écarts entre les valeurs observées x_i et cette valeur ξ^*. Définissons la fonction $f(\xi)$ qui renvoie la somme des carrés des écarts entre les x_i et une valeur arbitraire ξ :

$$\begin{array}{rl} f: & \mathbb{R} \to \mathbb{R} \\ & \xi \mapsto f(\xi) = \sum_{i=1}^{n}(x_i - \xi)^2 \end{array}$$

On cherche donc la valeur particulière ξ^* qui minimise cette fonction. Étudions la rapidement. Elle est continue et dérivable sur son ensemble de définition comme somme et produit de fonctions qui le sont. Les limites aux bornes de l'ensemble de définition sont :

$$\lim_{\xi \to \pm\infty} f(\xi) = +\infty$$

C'est rassurant : puisque $f(\xi)$ tend vers $+\infty$ des deux cotés, c'est bien qu'il doit exister (au moins) un minimum au milieu. Calculons la dérivée première, pas de difficulté, on applique $(u^2)' = 2uu'$, sauf qu'il faut bien penser que l'on dérive par rapport à ξ et non par rapport à x.

$$f'(\xi) = \frac{d}{d\xi} f(\xi) = \sum_{i=1}^{n} -2(x_i - \xi) = -2(\sum_{i=1}^{n} x_i - \sum_{i=1}^{n} \xi) = -2(\sum_{i=1}^{n} x_i - n\xi)$$

Cette dérivée s'annule pour un ξ^* qui n'est rien d'autre que la moyenne de l'échantillon $\bar{x}$.

$$\xi^* = \frac{1}{n}\sum_{i=1}^{n} x_i = \bar{x}$$

Il y a donc une tangente horizontale dans la courbe représentative de $f(\xi)$ pour $\xi = \bar{x}$. C'est le minimum puisque $f'(\xi)$ est négative avant $\bar{x}$ et positive après. Vérifions que nous ne nous sommes pas enduits d'erreur avec un point d'inflexion traître en calculant la dérivée seconde :

$$f''(\xi) = \frac{d}{d\xi} f'(\xi) = 2n > 0$$

La dérivée seconde est toujours strictement positive, la concavité de la courbe est donc toujours tournée vers le haut, il n'y a pas de point d'inflexion[3], mais bien le minimum que nous cherchions. C'est un point remarquable, en fait le seul que nous ayons, calculons la valeur de la fonction pour $\bar{x}$:

$$f(\xi^*) = \sum_{i=1}^{n}(x_i - \bar{x})^2 = n s_x^2$$

On voit apparaître ici la variance de l'échantillon, s_x^2, ce qui est normal puisqu'elle est définie comme la moyenne des carrés des écarts à la moyenne de l'échantillon $\bar{x}$.

[3] Il eût fallu que le signe de la dérivée seconde changeasse de signe en traversant la racine de la dérivée première, mais tel n'est pas le cas, n'est-ce-pas ?

Une région de confiance pour la valeur des paramètres avec un risque de première espèce α est donnée [1] par l'ensemble des valeurs des paramètres telles que la somme des carrés des résidus n'excède pas un seuil donné,

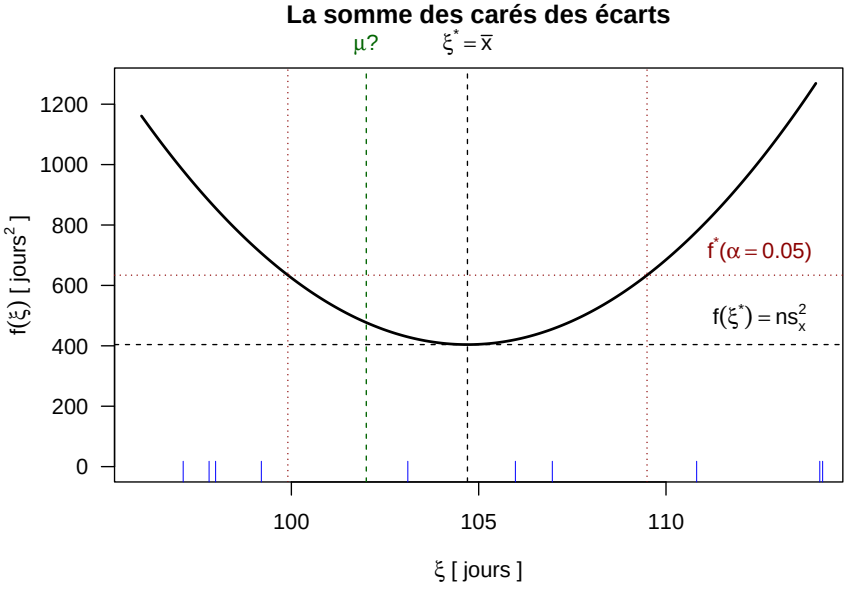
$$\theta \quad : \quad f(\theta) \leq f(\theta^*)(1 + \frac{p}{n-p} F^{\alpha}_{p;n-p})$$

où p est le nombre de paramètres du modèle, n le nombre de points disponibles dans le jeu de données, et θ^* le vecteur des valeurs des paramètres tel que la somme des carrés des écarts, $f(\theta^*)$, soit minimale. Dans notre cas avec $n = 10$ et $p = 1$, l'expression se simplifie en :

$$\xi \quad : \quad f(\xi) \leq f(\xi^*)(1 + \frac{1}{9} F^{\alpha}_{1;9}) \tag{1}$$

Représentons graphiquement ces informations.

```
x <- Reauphe$Day_of_Year ; n <- length(x)
f <- function(xi) sum((x - xi)^2)
xiseq <- seq(96, 114, le = 255)
y <- sapply(xiseq, f)
par(mar = c(5, 5, 4, 2) + 0.1)
plot(xiseq, y, type = "l", las = 1, ylim = c(0, max(y)),
     main = "La somme des carés des écarts", lwd = 2,
     xlab = bquote(paste(xi, " [ jours ]")),
     ylab = bquote(paste(f(xi), " [ jour", s^2, " ]")))
rug(jitter(x), col = "blue", ticksize = 0.05)
abline(v = mean(x), lty = 2)
text(mean(x), par("usr")[4], bquote(xi^"*" == bar(x)), xpd = NA, pos = 3)
var.n <- function(x) var(x)*(length(x) - 1)/length(x)
abline(h = n*var.n(x), lty = 2)
text(112.5, n*var.n(x), bquote(f(xi^"*") == n*s[x]^2), pos = 3)
seuil <- n*var.n(x)*( 1 + (qf(p = 1 - 0.05, df1 = 1, df2 = 9))/(9) )
abline(h = seuil, col = "darkred", lty = 3)
text(112.5, seuil, bquote(paste(f^"*", "(", alpha == 0.05, ")")), pos = 3, col = "darkred")
abline(v = t.test(x)$conf.int, col = "darkred", lty = 3)
abline(v = 102, col = "darkgreen", lty = 2)
text(102, par("usr")[4], bquote(paste(mu, "?")), xpd = NA, pos = 3, col = "darkgreen")
```



La somme des carrés des écarts a donc l'aspect d'une courbe parabolique qui est minimale pour la moyenne de l'échantillon $\bar{x}$, pas étonnant que la moyenne soit si souvent employée pour résumer un jeu de données. On peut comprendre ici pourquoi la variance de l'échantillon s_x^2 est un estimateur *biaisé* de la variance de la population. On ne connaît pas la moyenne de la population, μ, dont est issu notre échantillon, mais il y a bien peu de chances qu'elle soit exactement égale à celle de l'échantillon, on pourrait imaginer qu'elle soit par exemple à l'endroit figuré en vert sur le graphique, et là la somme des carrés des écarts est plus grande que $f(\bar{x})$. C'est parce que $\bar{x}$ minimise la somme des carrés des écarts des écarts que s_x^2 sous-estime la valeur de la population σ_x^2. C'est pour corriger ce biais que l'on utilise en pratique comme estimateur $\hat{\sigma}_x^2 = \frac{n}{n-1} s_x^2$.

Pour déterminer un intervalle de confiance pour la moyenne de la population il nous faut couper notre courbe à un certain seuil f^* défini par l'équation 1 et représenté par la ligne horizontale en rouge sur le graphique. Pour trouver les coordonnées en abscisse il faut trouver les racines de $f(\xi) - f^*$, ce que l'on peut faire avec la fonction de base `uniroot()` :

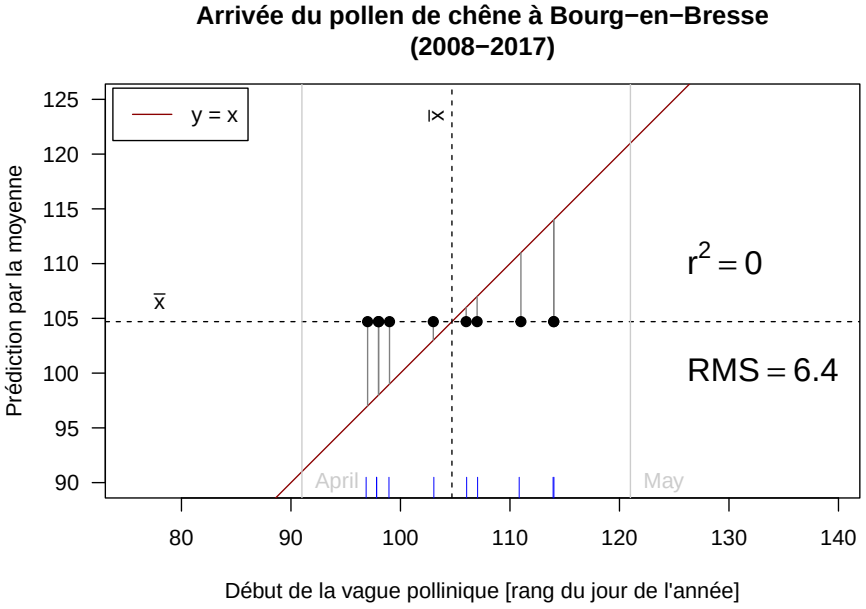
```
fseuil <- function(x) f(x) - seuil
uniroot(fseuil, c(99, 101))$root
[1] 99.9066
uniroot(fseuil, c(109, 110))$root
[1] 109.4934
```

Il est donc vraisemblable ($\alpha = 0.05$) que la date moyenne du début de la vague pollinique de chêne soit comprise entre le 100e et le 110e jour de l'année, soit à la mi-avril plus ou moins 5 jours, et, en arrondissant, à la mi-avril plus ou moins une semaine. Ce n'est pas d'une précision diabolique, mais avec seulement 10 observations il ne faut pas non plus s'attendre à des miracles. Une autre façon plus directe d'obtenir cet intervalle de confiance est de passer par un test de Student [4] implémenté dans la fonction de base `t.test()` :

```
c(t.test(x)$conf.int)
```
```
[1]  99.90657 109.49343
```

NOUS avons donc emprunté un chemin bien détourné pour retrouver les résultats d'un test classique. Ce n'a d'intérêt autre que didactique : nous pourrons suivre la même approche pour déterminer les régions de confiance de modèles à plusieurs paramètres. Représentons maintenant notre juge de paix : c'est un simple nuage de points dans un repère orthonormé où l'on confronte les données observées, x_i, en abscisse aux données prédites, y_i par un modèle, ici le modèle « moyenne ». On y ajoute quelques fioritures que nous allons expliquer.

```
myplot <- function(y,
                   main = "Arrivée du pollen de chêne à Bourg-en-Bresse\n(2008-2017)",
                   ylab = "Prédiction par la moyenne"){
  with(Reauphe, {
    x <- Day_of_Year
    lims <- c(90, 125)
    plot(x, y, pch = ".", las = 1, xlim = lims, ylim = lims,
         asp = 1, bg = rgb(0, 0, 0, 0.5),
         xlab = "Début de la vague pollinique [rang du jour de l'année]",
         ylab = ylab, main = main)
    abline(c(0, 1), col = "darkred")
    abline(h = mean(x), lty = 2)
    text(78, mean(x), bquote(bar(x)), pos = 3)
    abline(v = mean(x), lty = 2)
    text(mean(x), 124, bquote(bar(x)), srt = 90, pos = 2)
    r2 <- signif(cor(x, y)^2, 2) ; if(is.na(r2)) r2 <- 0
    text(125, 110, bquote(r^2 == .(r2)), cex = 1.5, pos = 4)
    RMS <- signif(sqrt(mean((x - y)^2)), 2)
    text(125, 100, bquote(RMS == .(RMS)), cex = 1.5, pos = 4)
    segments(x, x, x, y, col = grey(0.5))
    mois <- as.Date(sprintf("2017-%0.2d-01", 1:12))
    abline(v = yday(mois), col = grey(0.8))
    text(yday(mois), rep(90, 12), month(mois, label = TRUE, abbr = FALSE), pos = 4, col = grey(0.8))
    legend("topleft", inset = 0.01, legend = "y = x", lty = 1, col = "darkred")
    points(x, y, pch = 19)
    rug(jitter(x), col = "blue", ticksize = 0.05)
  })
}
y <- with(Reauphe, rep(mean(Day_of_Year), length(Day_of_Year)))
myplot(y)
```



Les traits bleus sur l'axe des abscisses qui figurent les valeurs observées, x_i, sont ici légèrement bruités avec la fonction `jitter()` pour repérer les valeurs dupliquées. Les points aux coordonnées (x_i, y_i) sont alignés sur une droite horizontale d'ordonnée $\bar{x}$ puisque nous avons résumé les données par la moyenne de l'échantillon. Les segments verticaux entre les points et la première bissectrice en rouge sont les résidus, $y_i - x_i$, entre les prédictions et les observations, c'est la somme des carrés d'iceux que nous avons minimisée. Pour un modèle parfait tous, les résidus seraient nuls et les points seraient alignés sur la première bissectrice en rouge.

On donne également la valeur de deux statistiques classiques. La première, r^2, est le carré du coefficient de corrélation linéaire qui exprime la fraction de la variabilité résiduelle (par rapport au modèle « moyenne ») pris en compte par un modèle. Elle est nulle ici puisque la moyenne ne nous dit absolument rien sur la variabilité inter-annuelle de la date du début de la vague pollinique. Elle serait égale à 1 pour un modèle parfait (100 % de la variabilité expliquée). La seconde, RMS pour *root mean square*, est la racine carrée de la moyenne des carrés des résidus. C'est un indicateur de la dispersion des résidus qui nous donne une idée de la précision que l'on peu attendre des prédictions du modèle. Dans le cas du modèle « moyenne » ce n'est rien d'autre que l'écart-type, s_x, de l'échantillon : on a une précision de l'ordre d'une semaine.

Notre modèle « moyenne » bien que très peu sophistiqué donne des indications utiles : le début de la vague pollinique n'arrive pas n'importe quel mois de l'année. On pourrait ainsi conseiller à l'habitant de Bourg-en-Bresse se sachant allergique au pollen de chêne d'éviter d'aller se promener en forêt à partir de la mi-avril, même si l'on sait que ce type de recommandation n'est pas toujours suivi d'effets [8, p. 140]. La question maintenant est de savoir si l'on peut être plus précis que ce modèle « moyenne » pour rendre compte de la variabilité inter-annuelle.

1.3 Les données en entrée

Les chênes étant poïkilothermes, ils ne contrôlent pas leur température interne, leur niveau d'activité métabolique dépend fortement de la température ambiante. Cette activité est nulle en dessous de 0°C et, en première approximation, croît linéairement avec les températures externes positives. Il est donc naturel d'introduire la température comme variable explicative. On utilise ici un extrait de la base SAFRAN [9], voir la fiche de TD[4] « Récupération de l'historique des données météo quotidiennes d'un site en France métropolitaine » pour l'importation dans R.

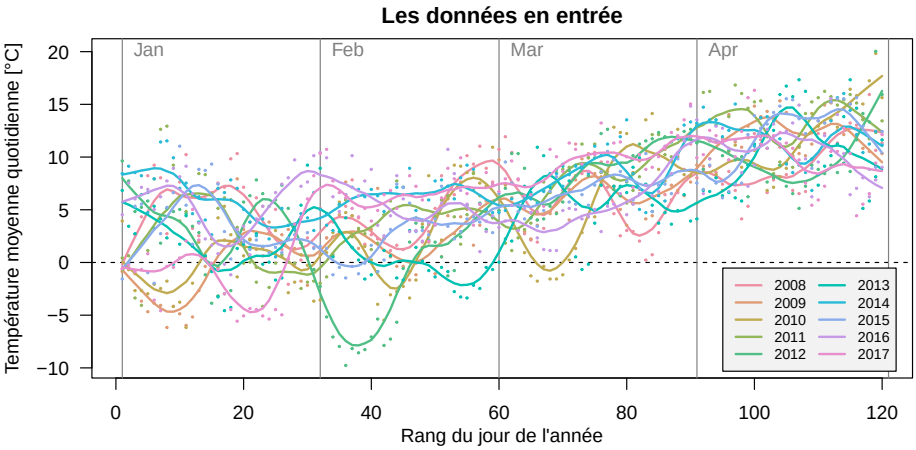
```
load(url(paste0(chmin, "Reaumet.Rda")))
head(Reaumet)
```

```
        Date Year doy  Tmean
1 2008-01-01 2008   1 -0.668
2 2008-01-02 2008   2 -0.668
3 2008-01-03 2008   3  2.632
4 2008-01-04 2008   4  4.332
5 2008-01-05 2008   5  8.232
6 2008-01-06 2008   6  9.232
```

La colonne `Tmean` contient les données en entrée : la température moyenne en degrés Celcius pour la maille SAFRAN la plus proche du site RNSA de Bourg-en-Bresse. On n'a conservé ici que les 10 années de 2008 à 2017 et les 120 premiers jours de chaque année. Il y a donc 1200 variables en entrée. Une représentation graphique possible est la suivante :

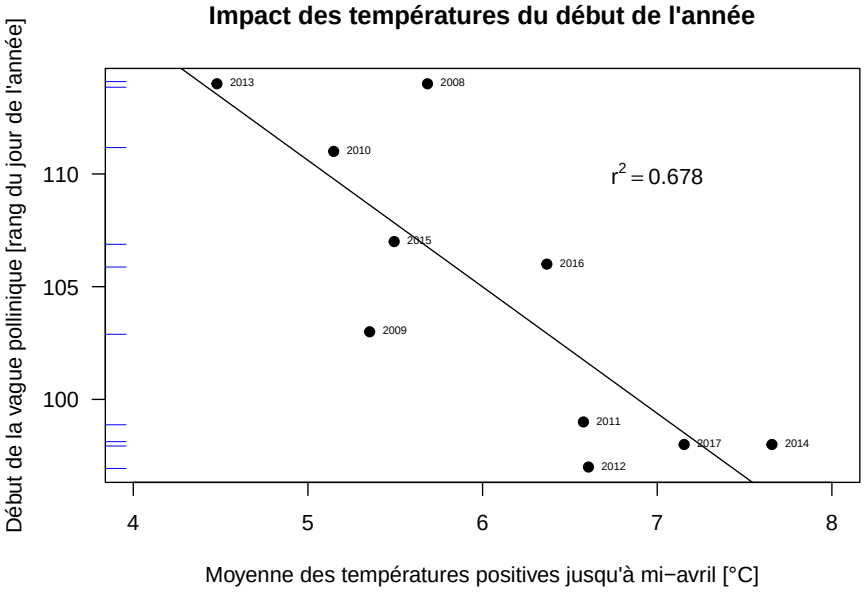
```
par(mar = c(3, 4, 2, 0) + 0.1)
plot.new() ; plot.window(xlim = range(Reaumet$doy), ylim = range(Reaumet$Tmean))
box() ; axis(1) ; axis(2, las = 1) ; abline(h = 0, lty = 2)
mois <- as.Date(sprintf("2017-%0.2d-01", 1:12))
abline(v = yday(mois), col = grey(0.5))
text(yday(mois[1:4]), rep(20, 4), month(mois[1:4], label = TRUE), pos = 4,
  col = grey(0.5), xpd = NA)
title(main = "Les données en entrée", ylab = "Température moyenne quotidienne [°C]")
title(xlab = "Rang du jour de l'année", line = 2)
Years <- unique(Reaumet$Year) ; ny <- length(Years) ; mycols <- hcl.colors(ny, "Set 2")
for(i in seq_len(ny)){
  with(subset(Reaumet, Year == Years[i]), {
    points(doy, Tmean, col = mycols[i], pch = 19, cex = 0.25)
    mys <- loess.smooth(doy, Tmean, span = 0.1, evaluation = length(doy))
    lines(mys[[1]], mys[[2]], col = mycols[i], lwd = 2)
  })
}
legend("bottomright", inset = 0.02, ncol = 2, legend = Years, lty = 1,
  col = mycols, cex = 0.75, lwd = 2, bg = grey(0.95))
```

[4] https://esb.univ-lyon1.fr/pdf/histoMeteo.pdf



On visualise bien l'augmentation des températures moyennes quand on avance dans la saison. La variabilité inter-annuelle est également bien visible, par exemple le début du mois de février a été particulièrement rigoureux en 2012. Pour voir si notre hypothèse de travail est réaliste on confronte la moyenne des températures positives jusqu'à la mi-avril à la date du début de la vague pollinique :

```
tmoy <- with(subset(Reaumet, doy <= 105), tapply(Tmean, Year, \(x) mean(abs(x))))
y <- Reauphe$Day_of_Year
plot(tmoy, y, pch = 19, las = 1, xlim = c(4, 8),
  xlab = "Moyenne des températures positives jusqu'à mi-avril [°C]",
  ylab = "Début de la vague pollinique [rang du jour de l'année]",
  main = "Impact des températures du début de l'année")
text(tmoy, y, Reauphe$Year, pos = 4, cex = 0.5)
rug(jitter(y), col = "blue", ticksize = 0.05, side = 2)
abline(lm(y~tmoy))
r2 <- signif(cor(y, tmoy)^2, 3)
text(7, 110, bquote(r^2 == .(r2)))
```



Il y a indubitablement un signal exploitable. Par exemple, les années les plus chaudes, plus de 6.5 °C, (2011, 2012, 2014 et 2017) sont également celles où la vague pollinique est la plus précoce (avant le 98e jour, le 8 avril). De même, les années les plus tardives (2008, 2010 et 2013), après le 110e jour (le 20 avril) sont également parmi les plus froides (moins de 6 °C). C'est donc cette information sur les températures que nous allons exploiter pour tenter d'améliorer le « modèle moyenne ».

2 Le match (le duel)

2.1 Le modèle phénologique de de Réaumur (1735)

On trouvera dans le premier chapitre de la thèse de Yann Vitasse [10] un état de l'art (2009) en français sur la phénologie des arbres et les modèles prédictifs utilisés. Dans la littérature scientifique les modèles du type de de Réaumur sont habituellement désignés par les acronymes GDD (*Growing Degree Days*) ou SW (*Spring Warming*).

Le modèle de de Réaumur qui date de 1735 [3] cherche à prédire, pour une année donnée, la date d'un stade phénologique, par exemple le début de la vague pollinique, à partir des données de température journalière. Les données en entrée, T_d, sont les températures moyennes journalières exprimées en degrés Celcius, mais on force les températures négatives à 0 (d'où le terme $max(T_d, 0)$ dans l'équation ci-après). Il consiste à faire la somme au cours du temps des degrés positifs jusqu'à ce qu'un seuil critique soit atteint. Il ne comporte que deux paramètres : la date t_0 à partir de laquelle on commence à sommer et le seuil critique $F^* \geq 0$ qu'il faut dépasser. La date prédite, t_d, tout comme t_0 sont exprimées classiquement pour ce type de modèle par le rang du jour dans l'année, donc entre 1 et 365 ou 366 selon que l'année est bissextile ou non.

$$t_d = min(t^*) \quad | \quad \sum_{d=t_0}^{t^*} max(T_d, 0) > F^*$$

La fonction `dddRl()` définie ci-dessous calcule la date prédite par le modèle de de Réaumur pour une série de valeur de température donnée et un jeu de paramètre donné. Pour éviter d'avoir une fonction objective « en escalier » on fait une interpolation avec une fonction de lissage des données météorologiques de façon à pouvoir travailler avec des t_0 et t_d pas forcément entiers.

```
# Entrées :
# d : vecteur des jours consécutifs de l'année
# Td : vecteur des températures moyennes correspondant
# param[1] : t0 : début de l'intégration
# param[2] : Seuil F* à dépasser
dddRl <- function(d, Td, param, warn = FALSE){
  d <- as.integer(d)
  # Check parameter values
  if(length(d) != length(Td)) stop("Different length for d and Td")
  if(!all(diff(d) == 1L)) stop("Consecutive values expected for d")
  if(length(param) != 2) stop("2 parameters are expected")

  t0 <- param[1]
  # On force t0 à rester dans les jours observés
  if(t0 < d[1]) t0 <- d[1]
  if(t0 > d[length(d)]) t0 <- d[length(d)]

  Fstar <- param[2]
```

```
  # On ne conserve que les degrés positifs
  Td[Td < 0] <- 0.0

  # On calcule les degrés-jours
  dj <- cumsum(Td)

  if(dj[length(dj)] < Fstar){
    if(warn) warning("F* unreachable")
    return(d[length(d)])
  }
  lo <- loess(dj~d, span = 0.1) # Lissage serré
  f <- function(x) predict(lo, x) - Fstar - predict(lo, t0)
  tryres <- try(uniroot(f, c(d[1], d[length(d)]))$root, silent = TRUE)
  if(inherits(tryres, "try-error")) return(d[length(d)])
  return(tryres)
}
```

La fonction `SSRRl()` renvoie la somme des carrés des écarts et calcule les valeurs prédites dans la colonne `théo` de la table `Reauphe` :

```
Reauphe$théo <- NA # colonne des valeurs prédites
SSRRl <- function(param, dmin = 1, dmax = 120){
  dspan <- dmin:dmax
  for(i in seq_len(nrow(Reauphe))){
    the_year <- Reauphe[i, "Year"]
    # indice du premier janvier de l'année :
    imétéo <- with(Reaumet, which(Year == the_year & doy == 1))
    Reauphe[i, "théo"] <<- with(Reaumet,
      dddRl(dspan, Tmean[(imétéo + dmin - 1):(imétéo + dmax - 1)], param))
  }
  return(with(Reauphe, sum((Day_of_Year - théo)^2)))
}
```

On utilise ici la fonction de base R `nlm()` pour trouver la valeur des paramètres qui minimise la somme des carrés des écarts. Voir la fiche de TD[5] « régression non linéaire » pour plus de détails sur les sorties de cette fonction.

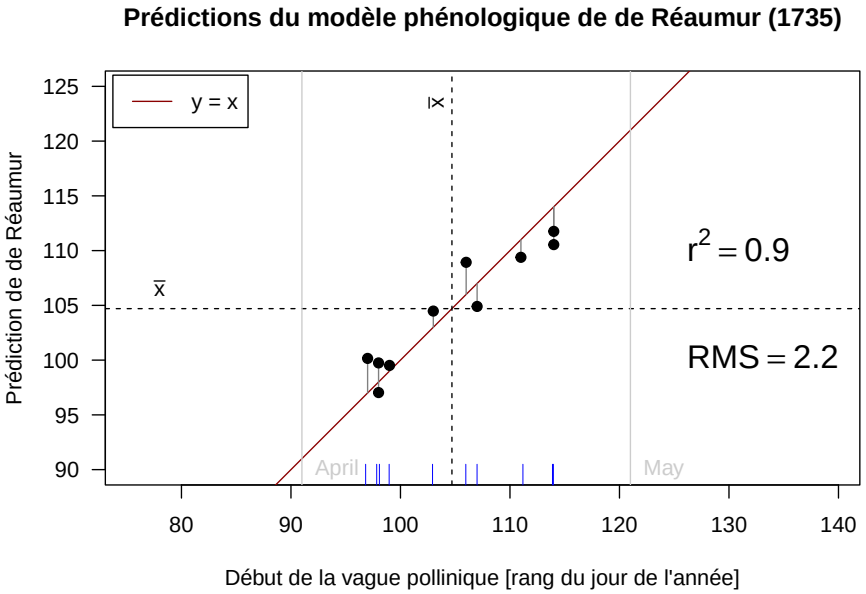
```
(resnlm <- nlm(SSRRl, c(50, 400)))
```

```
$minimum
[1] 48.95742
$estimate
[1]  57.87206 386.39706

$gradient
[1] -2.294725e-07 -4.106247e-08

$code
[1] 1

$iterations
[1] 14
```

On obtient une somme des carrés des écarts minimum de 48.95 pour $t_0 = 57.9$ (soit approximativement début mars) et $F^* = 386.4$ °C. On a une précision de l'ordre de 2.2 jours ici pour la prédiction du début de la vague pollinique avec le modèle de de Réaumur. Graphiquement :

```
invisible(SSRRl(resnlm$estimate)) # pour calculer les valeurs prédites
y <- Reauphe$théo
myplot(y, main = "Prédictions du modèle phénologique de de Réaumur (1735)",
       ylab = "Prédiction de de Réaumur")
```

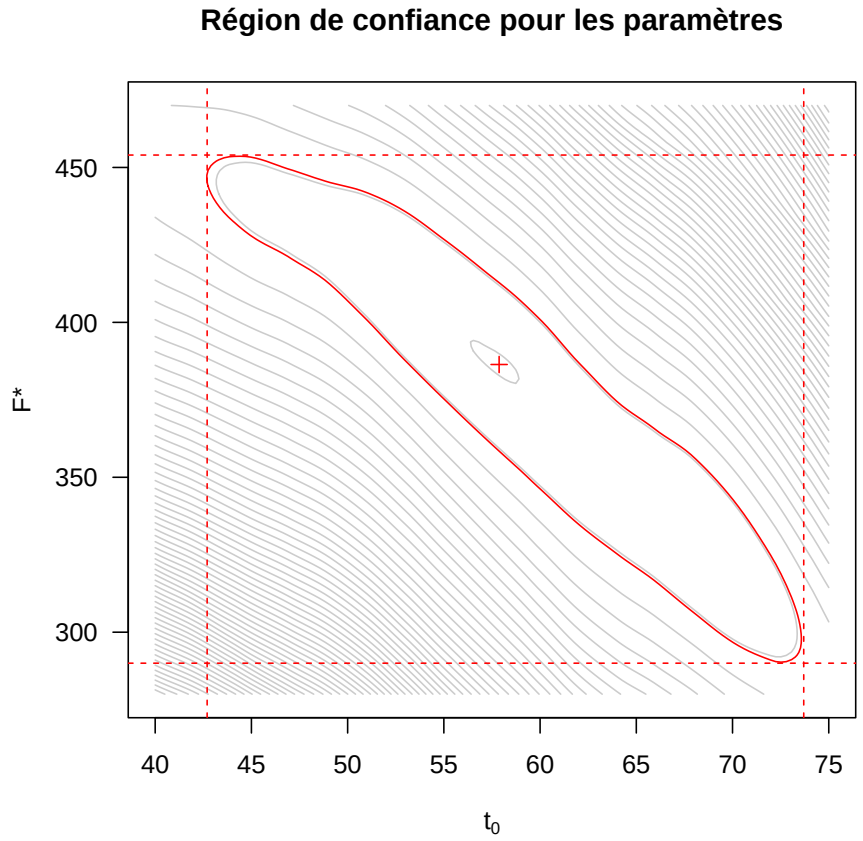
[5] tdr46 : `https://esb.univ-lyon1.fr/pdf/tdr46.pdf`



La fonction R de base `contour()` permet de visualiser simplement avec des courbes de niveau l'allure de la somme des carrés des écarts. La croix rouge représente l'estimation de la valeur des paramètres. La courbe en rouge donne la région de confiance pour les paramètres. La projection de cette région sur les axes donne les intervalles de confiance marginaux pour la valeur des paramètres.

```
scemin <- resnlm$minimum
n <- nrow(Reauphe)
p <- 2 ; alpha <- 0.05
seuil <- scemin*( 1 + (p*qf(p = 1 - alpha, df1 = p, df2 = n - p))/(n - p) )
npts <- 128
p1seq <- seq(40, 75, le = npts)
p2seq <- seq(280, 470, le = npts)
matSSR <- matrix(NA, nrow = length(p1seq), ncol = length(p2seq))
for(i in seq_len(nrow(matSSR))){
  for(j in seq_len(ncol(matSSR))){
    matSSR[i, j] <- SSRR1(c(p1seq[i], p2seq[j]))
  }
}
save(p1seq, p2seq, matSSR, resnlm, seuil, file = "Bataille/matSSR.Rda")
```

```
load(url(paste0(chmin, "matSSR.Rda")))
contour(p1seq, p2seq, matSSR, nlevels = 50, drawlabels = FALSE, col = grey(0.8),
        xlab = expression(t[0]), ylab = "F*", las = 1,
        main = "Région de confiance pour les paramètres")
contour(p1seq, p2seq, matSSR, levels = seuil, drawlabels = FALSE,
        add = TRUE, col = "red")
points(resnlm$estimate[1], resnlm$estimate[2], pch = 3, col = "red")
abline(v = c(42.7, 73.7), col  = "red", lty = 2)
abline(h = c(290, 454), col  = "red", lty = 2)
```



L'intervalle de confiance pour t_0 va de 42.7 (12 février) à 73.7 (15 mars), on a donc une incertitude de l'ordre du mois sur la date du début de l'intégration. L'intervalle de confiance pour F^* va de 290 à 454 °C. L'aspect de la région de confiance est la conséquence de la corrélation structurelle négative entre les paramètres, pour plus d'explications voir la fiche de TD[6] « Ajustement du modèle de de Réaumur (1735) aux données publiées (1735-1740) et manuscrites (1734-1756) de de Réaumur. »

2.2 Approche IA des forêts aléatoires (2001)

Pour une introduction tout en douceur et en français aux forêts aléatoires on consultera la fiche de TD[7] « Forêts aléatoires sombres » qui illustre pourquoi elles ont été introduites [2] pour pallier l'inconvénient de l'instabilité des arbres de décision. Nous utilisons ici la fonction `randomForest()` du paquet éponyme [7]. Pour la reproductibilité des résultats, on fixe la graine du générateur de nombres pseudo-aléatoires avec `set.seed(1)`.

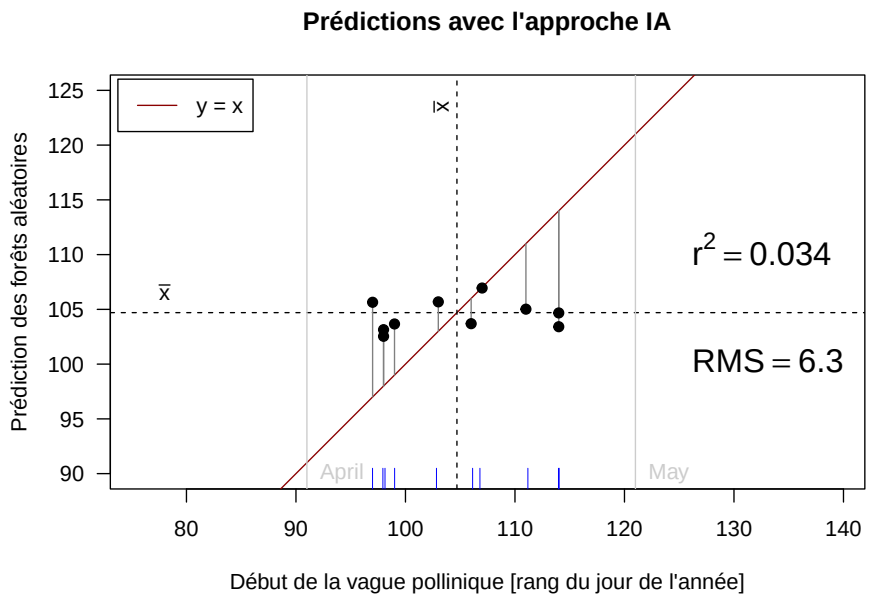
```
library(randomForest)
Yr <- Reauphe$Day_of_Year ; ncol <- length(unique(Reaumet$doy))
X <- matrix(NA, nrow = ny, ncol = ncol)
```

[6] https://esb.univ-lyon1.fr/pdf/tdr4R.pdf
[7] https://esb.univ-lyon1.fr/pdf/FAS.pdf

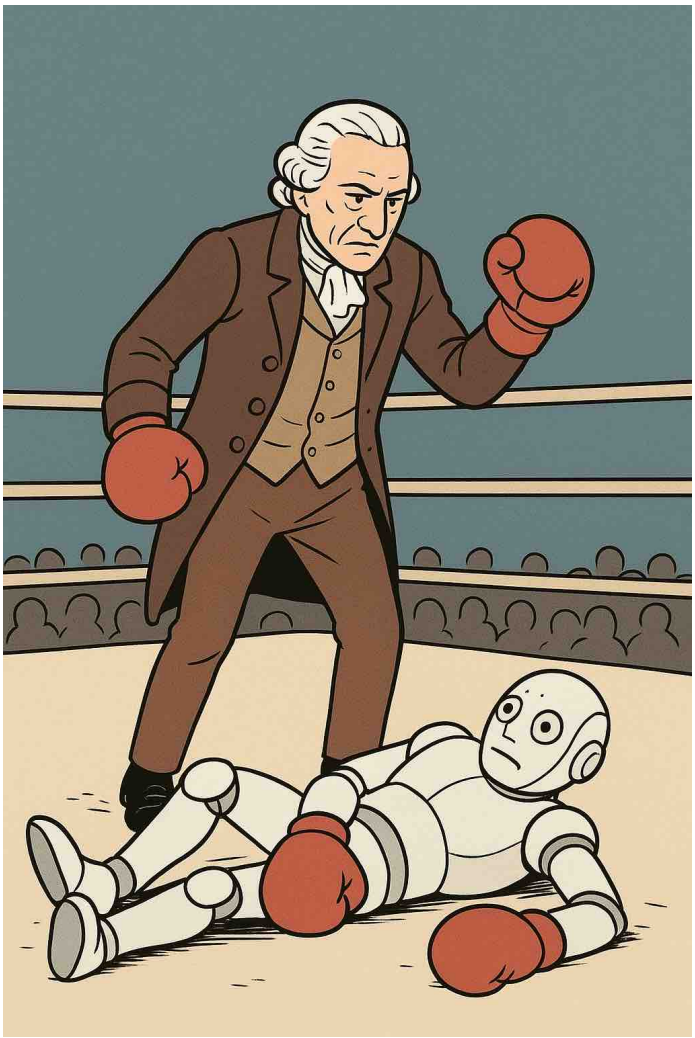
```
for(i in seq_len(ny)) X[i, ] <- subset(Reaumet, Year == Years[i])[ , "Tmean"]
rownames(X) <- Years ; colnames(X) <- paste0("d", 1:ncol)
X[1:5, 1:10]
```

```
        d1     d2     d3     d4     d5     d6     d7     d8     d9    d10
2008 -0.668 -0.668  2.632  4.332  8.232  9.232  8.032  2.732  6.632  7.632
2009 -0.868 -0.668 -1.068 -3.668 -4.668 -3.868 -4.068 -6.168 -4.168 -4.868
2010  3.932 -0.668 -2.668 -2.668 -2.968 -2.068 -2.468 -1.768 -4.168 -3.768
2011  0.432  0.632 -3.368 -4.068  1.132  7.932 12.632 12.932  9.232  3.932
2012  9.632  7.732  4.832  5.232  5.832  3.232  1.832  5.132  4.832  5.032
```

```
set.seed(1)
Yr.RF <- randomForest(X, Yr)
myplot(Yr.RF$predicted,
  main = "Prédictions avec l'approche IA",
  ylab = "Prédiction des forêts aléatoires")
```



2.3 Victoire par KO



C'est d'un *uppercut* magistral ($r^2 = 0.90$) que le modèle de de Réaumur envoie son opposant IA au tapis, où il reste complètement *groggy* ($r^2 = 0.03$). Insistons sur le fait que le match n'était pas truqué puisque que nous avons utilisé *exactement* les mêmes données en entrée et en sortie. Ce sont de plus des données issues d'une problématique réelle et non des données artificielles fabriquées pour les besoins de la cause.

3 Annexe : on refait le match

Ce n'est pas parce que nous avons exhibé un exemple édifiant où une approche IA se vautre lamentablement face à un modèle phénologique à deux paramètres du XXVIII^e siècle que nous serions en être satisfaits. Encore faut-il comprendre pourquoi. On refait le match. Que nul n'entre en cette annexe s'il n'est statisticien.

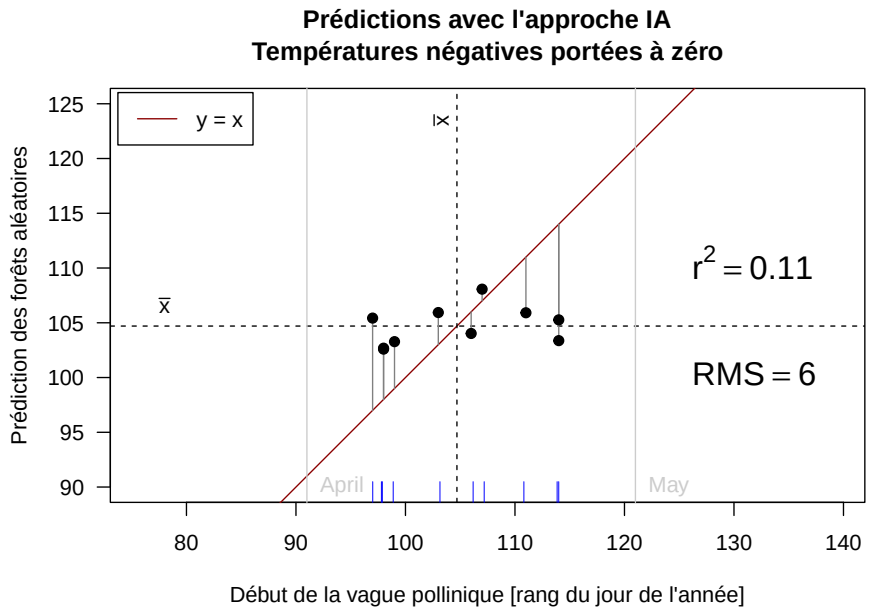
3.1 Les températures négatives

Dans le modèle de de Réaumur nous introduisons implicitement une connaissance biologique en portant les températures négatives à zéro : l'activité métabolique est nulle pour les températures négatives, quelles que soit leurs intensités. Ne serait-il pas plus juste de faire également cette opération pour les forêts aléatoires, pour comparer les approches sur un pied d'égalité ? Voyons ce qu'il en est :

```
Xpos <- matrix(NA, nrow = ny, ncol = ncol)
for(i in seq_len(ny)){
  x <- subset(Reaumet, Year == Years[i])[ , "Tmean"]
  x[x < 0] <- 0
  Xpos[i, ] <- x
}
rownames(Xpos) <- Years ; colnames(Xpos) <- paste0("d", 1:ncol)
Xpos[1:5, 1:10]
```

```
       d1    d2    d3    d4    d5    d6     d7     d8    d9   d10
2008 0.000 0.000 2.632 4.332 8.232 9.232  8.032  2.732 6.632 7.632
2009 0.000 0.000 0.000 0.000 0.000 0.000  0.000  0.000 0.000 0.000
2010 3.932 0.000 0.000 0.000 0.000 0.000  0.000  0.000 0.000 0.000
2011 0.432 0.632 0.000 0.000 1.132 7.932 12.632 12.932 9.232 3.932
2012 9.632 7.732 4.832 5.232 5.832 3.232  1.832  5.132 4.832 5.032
```

```
set.seed(1)
Yr.RFpos <- randomForest(Xpos, Yr)
myplot(Yr.RFpos$predicted,
  main = "Prédictions avec l'approche IA\nTempératures négatives portées à zéro",
  ylab = "Prédiction des forêts aléatoires")
```



On a amélioré les performances du modèle mais que de façon extrêmement marginale. Ce n'est pas ici que l'on trouvera l'explication des piètres performances des forêts aléatoires.

3.2 Les séries temporelles

Les données utilisées en entrée sont des séries temporelles : il y a une relation d'ordre totale entre les températures quotidiennes qui définit leur relation chronologique au cours de l'année. Cette structure est complètement ignorée par les forêts aléatoires, comme on peut s'en convaincre en mélangeant les données :

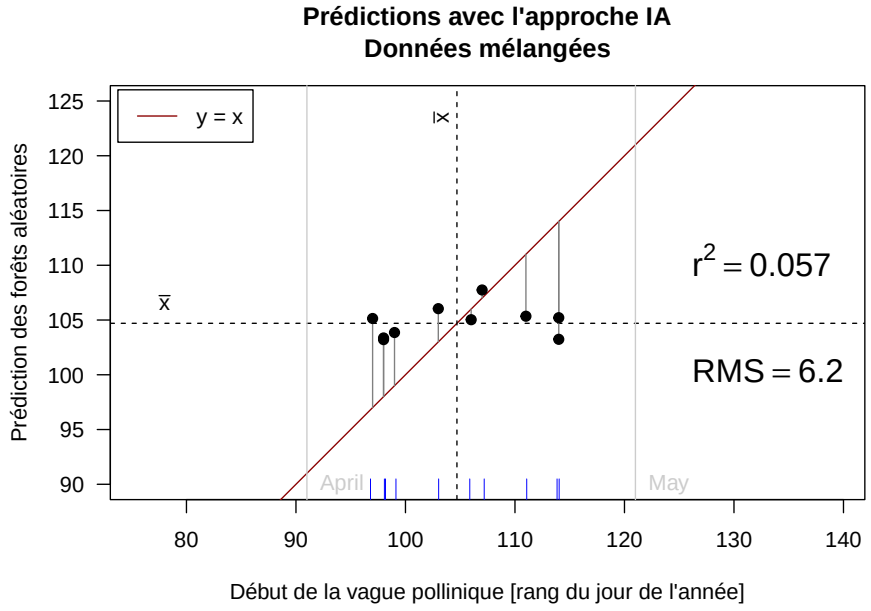
```
head(X[ , 1:5])
```

```
          d1     d2     d3     d4     d5
2008 -0.668 -0.668  2.632  4.332  8.232
2009 -0.868 -0.668 -1.068 -3.668 -4.668
2010  3.932 -0.668 -2.668 -2.668 -2.968
2011  0.432  0.632 -3.368 -4.068  1.132
2012  9.632  7.732  4.832  5.232  5.832
2013  6.832  3.432  2.732  6.432  6.632
```

```
set.seed(1)
Xr <- X[ , sample(1:ncol(X))]
head(Xr[ , 1:5])
```

```
       d68    d39     d1    d34    d87
2008  4.832  1.732 -0.668  3.832  5.932
2009  4.632  0.332 -0.868  2.432  7.332
2010 -1.568  1.532  3.932  2.832  9.632
2011  6.632  2.732  0.432  0.732 10.032
2012  5.332 -8.268  9.632 -7.068 11.432
2013  7.132 -1.568  6.832  0.532  3.632
```

```
Yr.RFr <- randomForest(Xr, Yr)
myplot(Yr.RFr$predicted,
  main = "Prédictions avec l'approche IA\nDonnées mélangées",
  ylab = "Prédiction des forêts aléatoires")
```



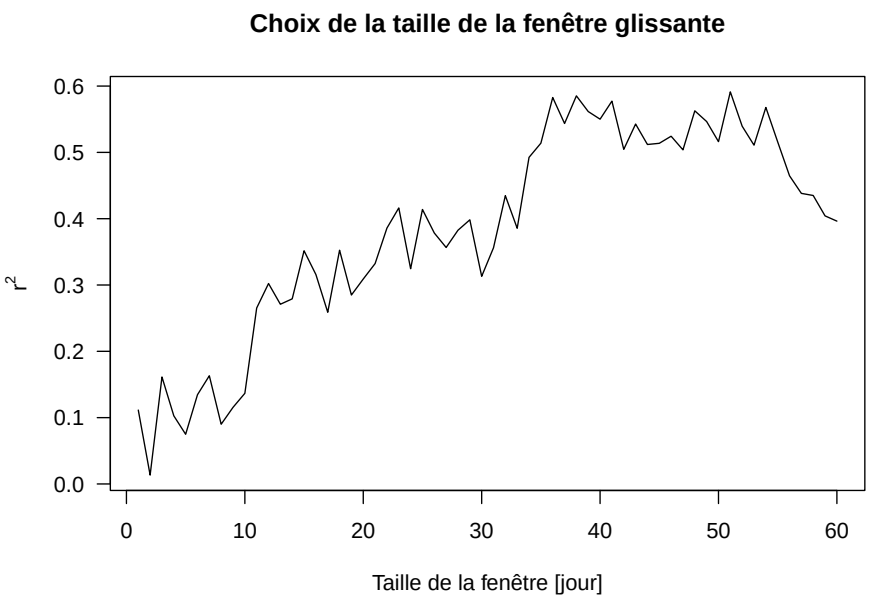
3.3 Fenêtres glissantes

Une façon possible de tenir compte de la structure temporelle des données et de travailler non pas sur les valeurs ponctuelles quotidiennes mais sur des valeurs lissées, en prenant la moyenne sur une fenêtre glissante. On fait une petite analyse en amont pour trouver la taille optimale de la fenêtre.

```
fgliss <- function(x, span = 51){
  n <- length(x) ; res <- numeric(n)
  for(i in seq_len(n)) res[i] <- mean(x[i:(i+span-1)], na.rm = TRUE)
  return(res)
}
n <- 60 ; r2 <- numeric(n)
for(i in seq_len(n)){
  Xg <- t(apply(Xpos, 1, fgliss, span = i))
  set.seed(1)
  Yr.RFg <- randomForest(Xg, Yr)
  r2[i] <- cor(Yr, Yr.RFg$predicted)^2
}
which.max(r2)
```

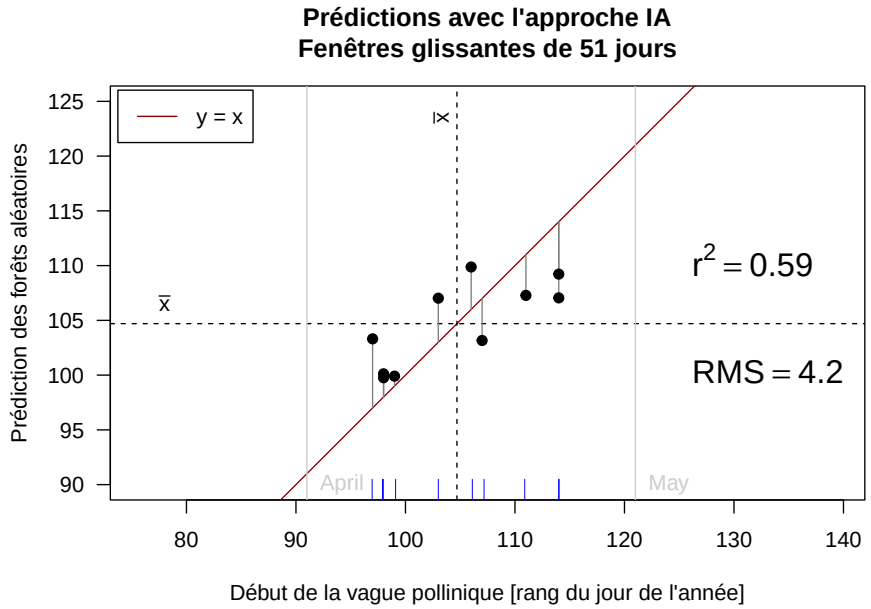
```
[1] 51
```

```
plot(r2, type = "l", las = 1,
  xlab = "Taille de la fenêtre [jour]",
  ylab = bquote(r^2),
  main = "Choix de la taille de la fenêtre glissante")
```



C'est donc avec une fenêtre glissante de 51 jours, soit environ 7 semaines, que nous obtenons les meilleurs résultats. Visualisons les prédictions du modèle.

```
Xg <- t(apply(Xpos, 1, fgliss))
colnames(Xg) <- paste0("d", 1:ncol(Xg))
set.seed(1)
Yr.RFg <- randomForest(Xg, Yr)
myplot(Yr.RFg$predicted,
  main = "Prédictions avec l'approche IA\nFenêtres glissantes de 51 jours",
  ylab = "Prédiction des forêts aléatoires")
```



C'est beaucoup mieux, mais avec un $r^2 = 0.59$ on est encore loin des performances ($r^2 = 0.9$) du modèle de de Réaumur. On peut néanmoins conclure que c'est bien la non prise en compte de la structure temporelle des données par les forêts aléatoires qui est responsable de la mauvaise performance d'icelles.

Références

[1] E.M.L. Beale. Confidence regions in non-linear estimation. *Journal of the Royal Statistical Society*, 22B :41–88, 1960.

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

[3] R.-A.F. de Réaumur. Observations du thermomètre faites à Paris pendant l'année 1735 comparées avec celles qui ont été faites sous la ligne, à l'Île de France, à Alger et en quelques-unes de nos Îles de l'Amérique. *Mémoires de l'académie royale des sciences de Paris*, 1738 :545–576, 1735.

[4] Student [Gosset, W.S.]. The probable error of a mean. *Biometrika*, 6 :1–25, 1908.

[5] G. Grolemund and H. Wickham. Dates and times made easy with lubridate. *Journal of Statistical Software*, 40(3) :1–25, 2011.

[6] L. Keurinck. *Changement climatique et reproduction des plantes pérennes : le rôle clé de la phénologie florale*. PhD thesis, Université Claude Bernard - Lyon 1, 2024.

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

[8] M. Thibaudon and J.-P. Besancenot. Forêts et allergies. *Revue forestière française*, 70 :137–146, 2018.

[9] J.-P. Vidal, E. Martin, L. Franchistéguy, M. Baillon, and J.-M. Soubeyroux. A 50-year high-resolution atmospheric reanalysis over France with the safran system. *International Journal of Climatology*, 30(11) :1627–1644, 2010.

[10] Y. Vitasse. *Déterminismes environnemental et génétique de la phénologie des arbres de climat tempéré. Suivi des dates de débourrement et de sénescence le long d'un gradient altitudinal et en tests de provenances*. PhD thesis, Université de Bordeaux 1, 2009.