Rapport STA202

Cyril Nefzaoui Blanchard , Bertrand Paturel

February 2020

1 Introduction

Pour ce projet nous avons choisi d'étudier des relevés de température pour le parc Montsouris à Paris. Nous avons donc récupéré un jeu de données qui contient les champs suivant :

- l'identifiant de la station de mesure de la température (ici c'est toujours celle du Parc de Montsouris)
- le nom de la station (Ici c'est toujours le parc de Montsouris dans le 14ème)
- la date (distibuée par journée, le champs date va du 2009-06-01 au 2017-09-30, soit plus de 8 ans de données)
- le volume de précipitations de pluie
- les relevés de precipitation de neige
- la temperature moyenne TAVG (en degrés Fahrenheit)
- la température maximale TMAX (en degrés Fahrenheit)
- la temperature minimale TMIN (en degrés Fahrenheit)

Dans le jeu de données, l'unité associée à la température n'était pas spécifiée. A priori, la température est soit en degrés Celsius $°C$, soit en degrés Kelvin $°K$, soit en degrés Fareinheit $°F$ .D'après les ordres de grandeurs de température de notre série, on a établi que la température est en $°F$.

Nous avons choisi de travailler sur la série temporelle TAVG représentant la température moyenne lors de la journée car celle ci est plus facilement prévisible, alors que les températures extrêmes ne suivent pas forcément une tendance ou saisonalité autant appréciable. D'autre part les valeurs aberrantes sur les températures extrêmes sont bien plus fréquentes puisqu'elles correspondent aux niveaux de saturation des capteurs.

2 Analyse des données

2.1 Description de la source

Nous gardons donc d'un point de vue pratique uniquement la date et la température moyenne dans notre jeu de données, quitte à revenir plus tard pour comparer nos résultats avec les allures des autres champs commes les températures extremales. Nous avons donc donc un Data Frame avec deux colonnes, la Date et les températures moyennes pour chaque jour.

2.2 Prétraitement

Avant de selectionner nos données pour l'entrainement et la prédiction nous allons effectuer un prétraitement sur celle ci pour enlever toutes les valeurs aberrantes ou de saturation des capteurs.

Tout d'abord nous regardons si toutes les données correspondent au Parc MONTSOURIS et deuxièmenet si il y a plusieurs capteurs dans le Parc. On remarque donc qu'il n'y a qu'un seul capteur de température dans le Parc MONTSOURIS et qu'il n'y a que des relevés du Parc.

```
setwd("C:/Users/patur/OneDrive/Bureau/ENSTA/ENSTA Cours/STA/STA202/projet_STA20
Data=read.csv(file="montsouris.csv", sep=",", dec='.')
plot(Data$DATE,Data$TAVG)

#Il y a t il des relevÃ©s qui ne correspondent pas au Parc MONTSOURIS
#Il y a t il plusieurs points de relevÃ©s dans le Parc ?
which(Data$STATION!="FR000007150")
which(Data$NAME!="PARIS 14E PARC MONTSOURIS, FR")

#Analyse des valeurs aberrantes
which(is.na(Data$TMIN))
Data$TMIN[15]
Data$TAVG[15]
Data$TMAX[15]
```

2.3 Mise en forme des données

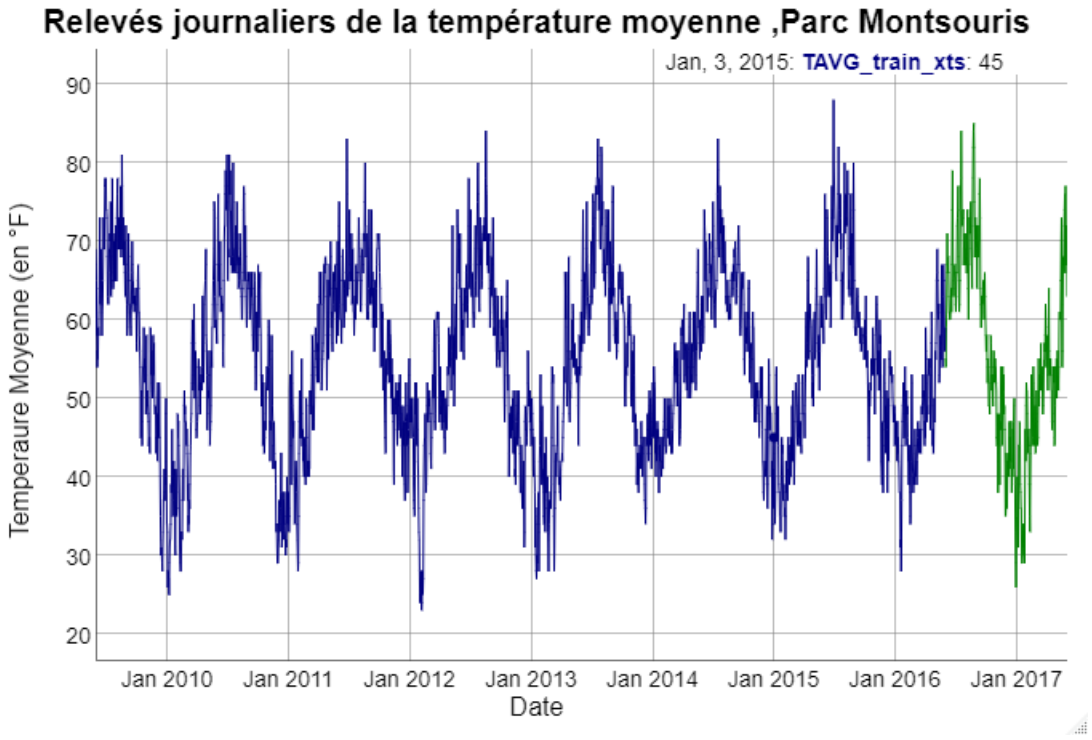
Nous allons diviser notre jeu donnée en deux parties pour cette étude. Un premier jeu de données, que nous appelerons $TAVG - train$ pour analyser l'évolution des températures ainsi qu'un deuxième jeu de données, nommé $TAVG - test$ avec la dernière année que l'on cherchera à prédire. Le jeu de données train est s'étale sur 7 années consécutives complètes, du 2009-06-01 au 2016-06-01.Le jeu de de données test est une année complète consécutive qui succède directement au jeu de données d'entrainement du 2016-06-02 au 2017-06-03. Nous avons choisi cette mise en forme des données afin de faciliter la mise en évidence d'une éventuelle saisonnalité annuelle (1 période correspond à une année).On considère donc 7 périodes pour l'entrainement et 1 période pour le test.

on choisit de considérer 7 années complètes consécutives pour l'entrainement, puis 1 année consécutive complète le train va donc du 2009-06-01 au 2016-06-01 (2545 jours) et le test va du 2016-06-02 au 2017-06-03 (365 jours)

On va creer artificiellement un test de 1 an et on ignore donc les derniers jours des données (119 jours ignorés)

2.4 Visualisation des données

On a représenté en bleu les données d'entrainement que l'on suppose connues et en vert les données à prédire.



3 Modelisation des données

3.1 Tendance et saisonnalité

Etant donné le graphique ci dessus l'on a clairement un modèle additif avec une saisonnalité annuelle. Ainsi le modèle de notre série que l'on cherche a déterminer est $y(t) = f(t) + s(t) + X(t)$

avec :

-f la tendance : f doit être une fonction déterministe du temps.

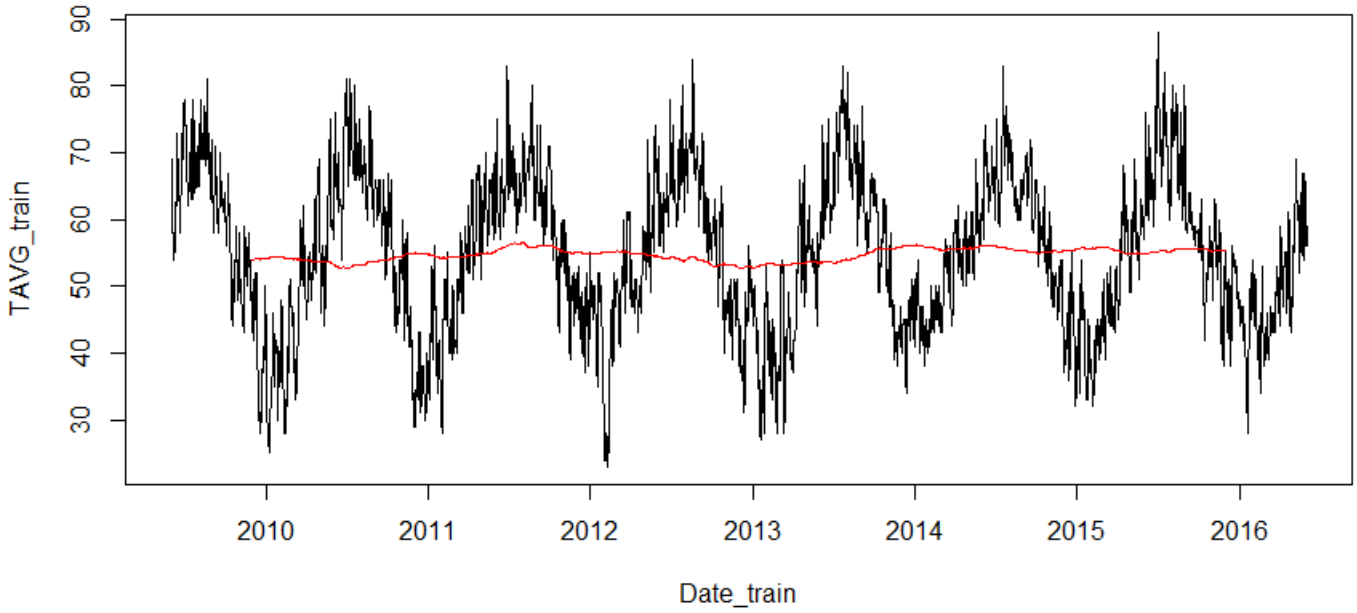
-s la saisonnalité : s doit être une fonction périodique

-X_t un processus aléatoire, qui doit au moins être faiblement stationnaire.

Nos données sont des relevés de température moyenne, journaliers, étalés sur une période de 7 années complètes pour la partie d'apprentissage, et 1 année complète pour la partie test.

3.1.1 Tendance

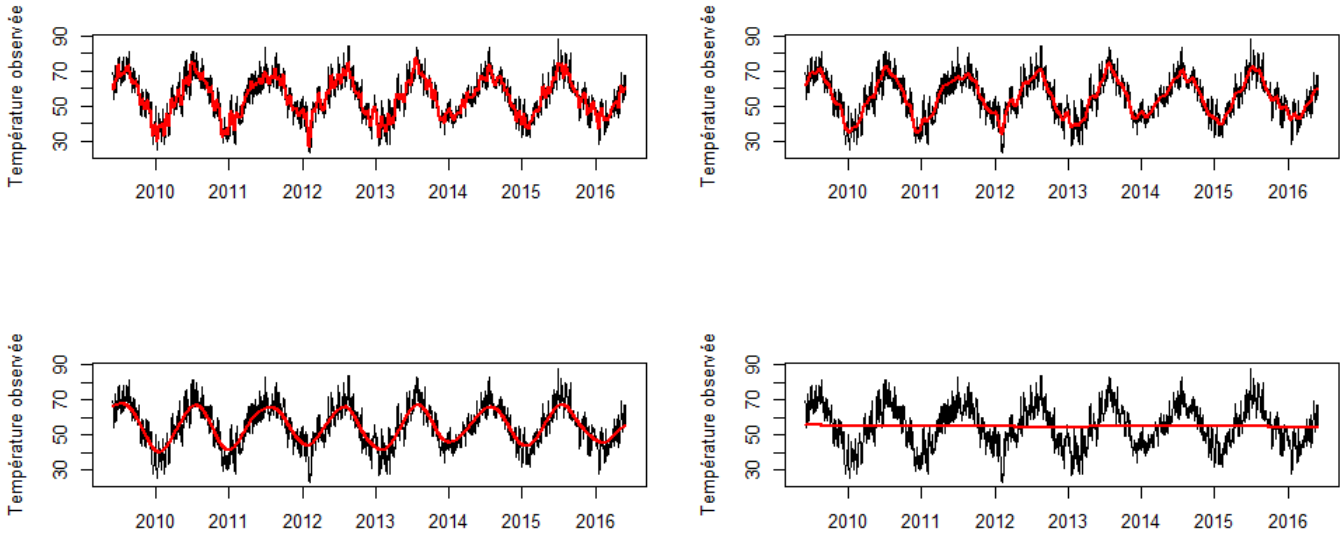
Dans un premier temps, nous cherchons à estimer la tendance. On utilise un filtrage par moyenne mobile. Ce filtrage par moyenne mobile permet de désaisonnaliser la série et ainsi de faire apparaitre la tendance. En effet, en choisissant comme largeur de fenêtre la période de notre série (estimée annuelle à 365, cf partie Saisonalité), la composante périodique est filtrée, puisqu'elle est par définition d'intégrale nulle sur une période.



La tendance est constante, ce qui est cohérent avec notre modélisation. En effet, la température ne devrait pas présenter de tendance croissante ou décroissante étant donné que c'est une grandeur physique naturellement périodique.

Nous choisissons alors d'utiliser une méthode de régression à noyau. Cette méthode nous a paru plus avantageuse car on peut directement contrôler la régularité de l'estimateur de la tendance. Ainsi, nous avons implémenté un estimateur à noyau gaussien, en faisant varier la largeur de la fenêtre de lissage :(10,25,100,1000).

On met alors en évidence qu'il n'y a pas de tendance non constante dans notre série : en effet, pour des fenetres larges, la série parait périodique :les seules évolutions sont dues à la saisonnalité. Nous voulons finalement ignorer les effets de la saisonnalité, on choisit alors une fenêtre très large (1000).La série obtenue après filtrage est constante,ce qui met en évidence que la tendance est une fonction constante.



3.1.2 Saisonnalité

On a remarqué que notre jeu de données présente une évolution cyclique,ce qui normal puisque la température doit a priori avoir une saisonnalité annuelle.On va donc mettre en évidence une saisonnalité de période 365. Pour cela, on va faire une décomposition en série de Fourier. Dans notre cadre, il s'agit en fait de faire une régression linéaire sur une base trigonométrique de cardinal fixé. On choisit de construire une base de dimension 10, puis on fait la régression sur cette base .

On obtient deux résultats intéressants (fenetre=50)

```
> reg<-lm(TAVG_train~fourier[,1:2])
> summary(reg)

Call:
lm(formula = TAVG_train ~ fourier[, 1:2])

Residuals:
     Min       1Q   Median       3Q      Max
-18.4314  -4.1378  -0.0197   4.1854  20.4620

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       54.7628     0.1187   461.3   <2e-16 ***
fourier[, 1:2]1    9.3868     0.1682    55.8   <2e-16 ***
fourier[, 1:2]2  10.4066     0.1676    62.1   <2e-16 ***
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.989 on 2542 degrees of freedom
Multiple R-squared:  0.7329,    Adjusted R-squared:  0.7327
F-statistic:  3487 on 2 and 2542 DF,  p-value: < 2.2e-16

>
```

Premièrement, les deux seuls coefficients significatifs sont les fondamentaux (il n'y a pas d'harmoniques). On obtient un coefficient R assez elevé mais qu'on peut encore améliorer (R=0.75).

On refait une regression linéaire sur le signal filtré (regression sur noyau gaussien avec l=50) et là on obtient un bien meilleur résultat car le signal n'est plus bruité.

```
Call:
lm(formula = noyau$y ~ fourier[, 1:2])

Residuals:
    Min      1Q  Median      3Q     Max
-5.3052 -1.0880  0.2194  1.2918  3.8279

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)      54.74604    0.03430  1596.3   <2e-16 ***
fourier[, 1:2]1  7.64427     0.04859   157.3   <2e-16 ***
fourier[, 1:2]2  8.59080     0.04841   177.5   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.73 on 2542 degrees of freedom
Multiple R-squared:  0.9568,    Adjusted R-squared:  0.9567
F-statistic: 2.814e+04 on 2 and 2542 DF,  p-value: < 2.2e-16
```

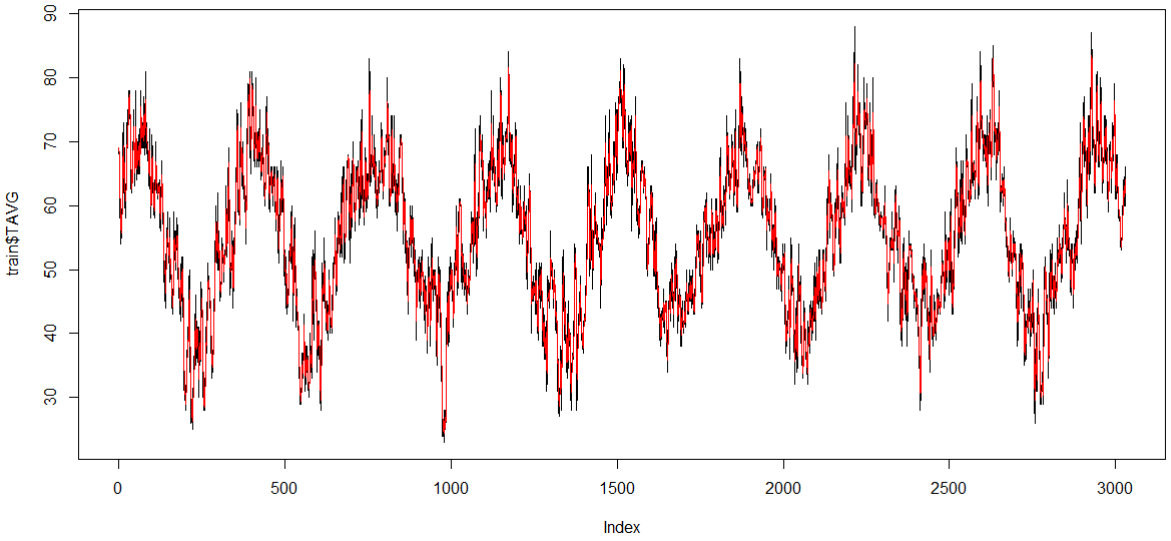
4 Prévision sur notre échantillon test

4.1 Méthodes de lissage exponentiel

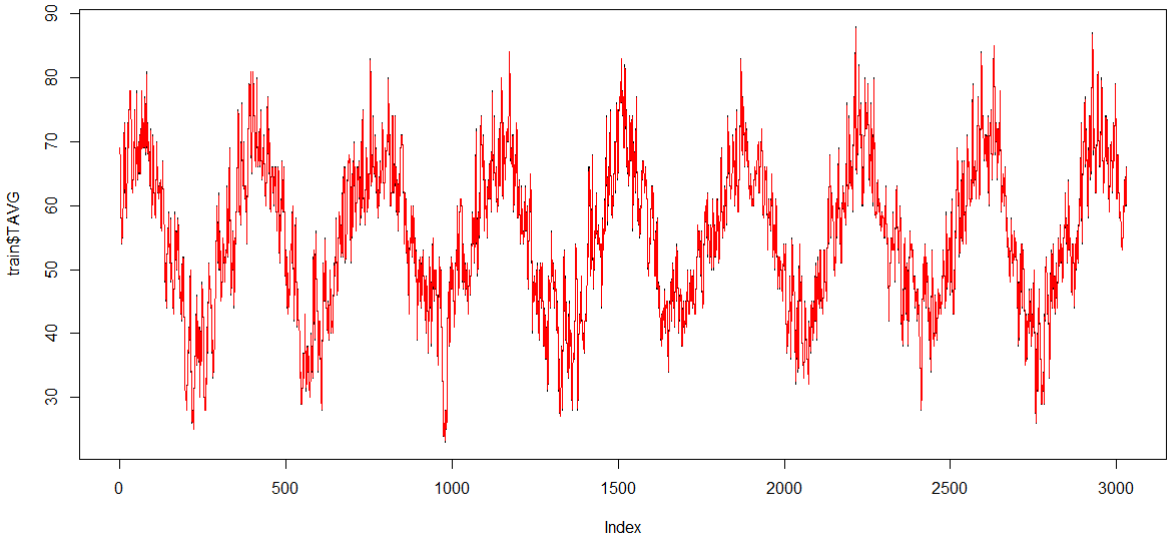
Dans un premier temps, nous allons utiliser la méthode du lissage exponentiel pour lisser nos données.

4.2 Lissage exponentiel simple smooth

On effectue premièrement un lissage exponentiel simple. On choisit comme paramètre α la valeur numérique 0.5, et on obtient alors le lissage suivant en rouge :

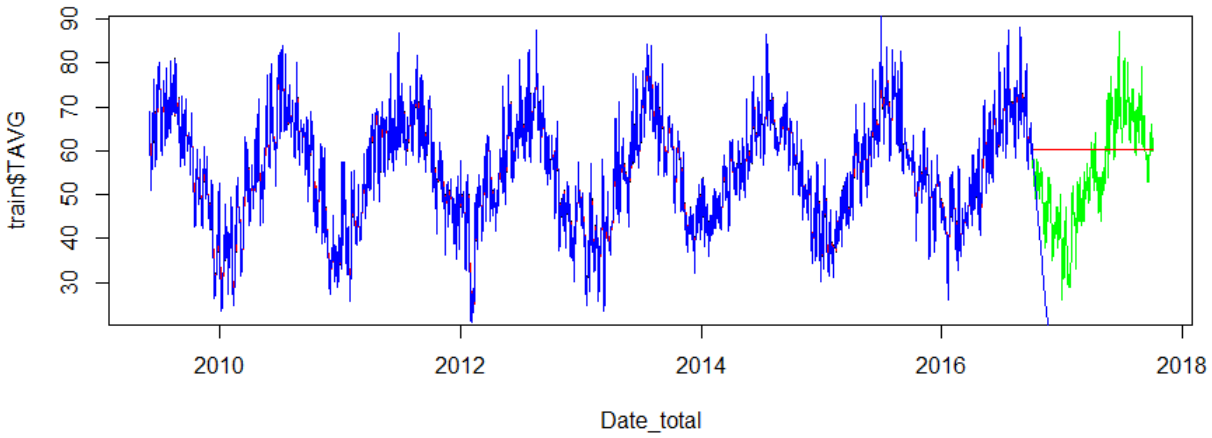


On obtient $RMSE = 2.14$ Pour améliorer notre lissage, on peut tester plusieurs paramètres α, puis choisir celui qui minimise le RMSE. Pour cet α optimal, on obtient une courbe lissée qui reproduit fidèlement les données. On détermine que le α optimal est 0.95



Ici, $RMSE = 0.19$ donc le lissage avec $\alpha = 0.95$ approxime mieux la série d'origine.

On peut également utiliser des méthodes de lissage exponentiel pour prévoir le futur de la série à horizon donné. Par construction, le lissage exponentiel simple va prédire l'echantillon test avec une constante, ce qui est une prévision très grossière.On peut améliorer cette prévision en utilisant un lissage exponentiel double : on obtient alors une droite, ce qui reste peu satisfaisant. Enfin, on peut utiliser le lissage triple (Holt-Winters) qui permet de prendre en compte la saisonnalité et donc d'approximer de manière plus fidèle le futur de la série.



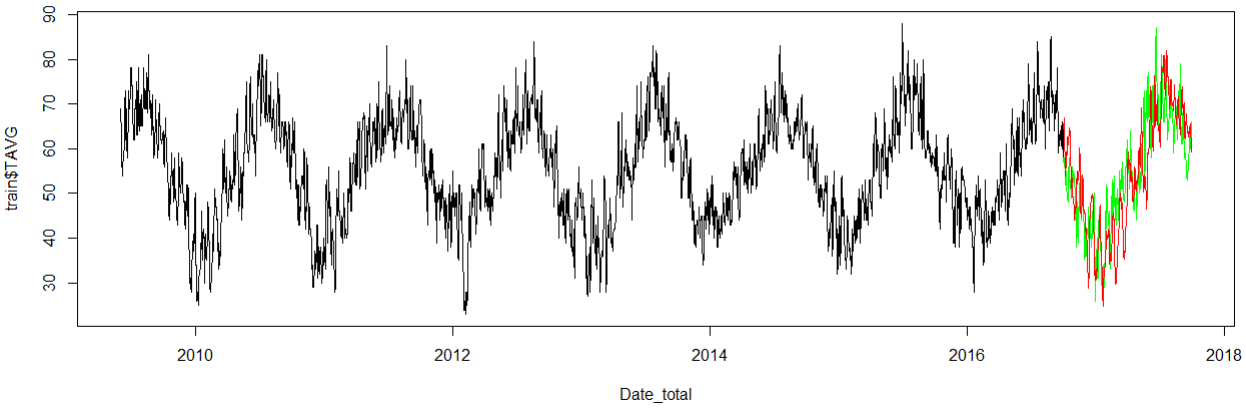
Nous avons représenté ici :

- en rouge la prévision par la constante du lissage simple.
- en bleu la prévision par la droite du lissage exponentiel double.
- en vert la prédiction de Holt Winters. On obtient, en considérant l'écart entre les valeurs fixées du test et les valeurs obtenues par prédiction

$$RMSE_{simple} = 12.7 \text{ et } RMSE_{double} = 570$$
$$RMSE_{Holt-Winters} = 12.4$$

On remarque que puisqu'on évalue sur une sur une période complète, en moyenne la température vaut son esperance sur la période donc c'est normal de trouver un RMSE très faible lorsque qu'on fait une prévision simple (prévision par une constante). Par contre , le RMSE donné par le lissage double est très élevé donc la prévision par lissage double est mauvaise au sens du RMSE. Enfin , le RMSE fourni par Holt-Winters est le plus faible.Il présente cependant l'inconvénient de prédire en conservant les irrégularités de la série d'entrainement.

On compare finalement dans ce dernier graphique la modélisation de Holt Winters en vert et les valeur réelles en rouge.



4.3 Modélisation de la partie aléatoire de la série par un processus ARMA(1,1)

4.3.1 Etude de la stationnarité de la série

On rappelle que le modèle de notre série est : $y(t) = f(t) + s(t) + X_t$ où :

-f la tendance : f doit être une fonction déterministe du temps. On a déterminé que c'est en fait une constante $f(t) = \mu$

-s la saisonnalité : s doit être une fonction périodique

-X_t un processus aléatoire qui représente le bruit qui doit au moins être faiblement stationnaire.

A ce stade de la modélisation, on a : $y(t) = \mu + \alpha \sin(2\pi t/365) + \beta \cos(2\pi t/365) + X_t$

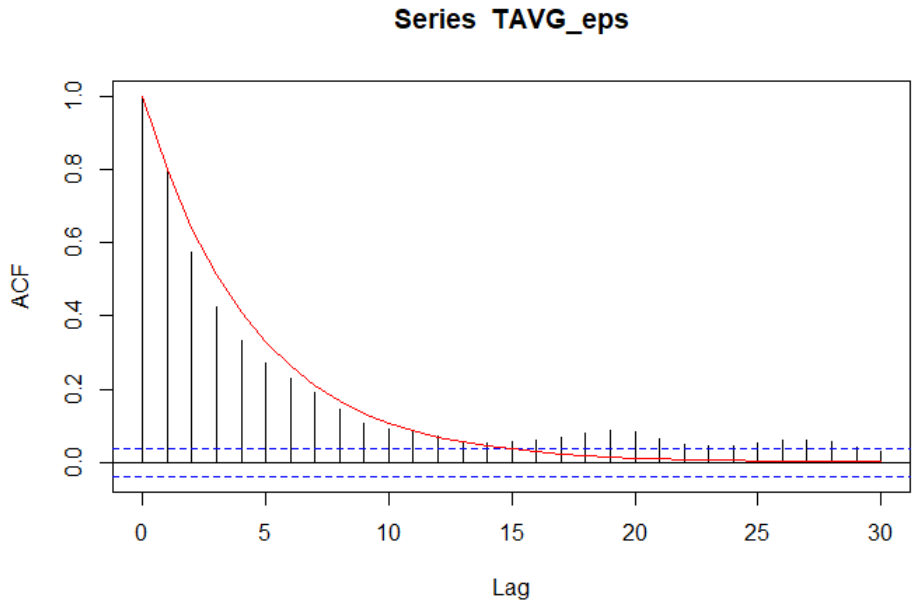
Désormais, nous allons étudier le caractère stationnaire de la série.

- Vu que la tendance de la série temporelle $y(t)$ qui représente la température moyenne à une tendance constante (et donc indépendante du temps), cette série peut donc être stationnaire.

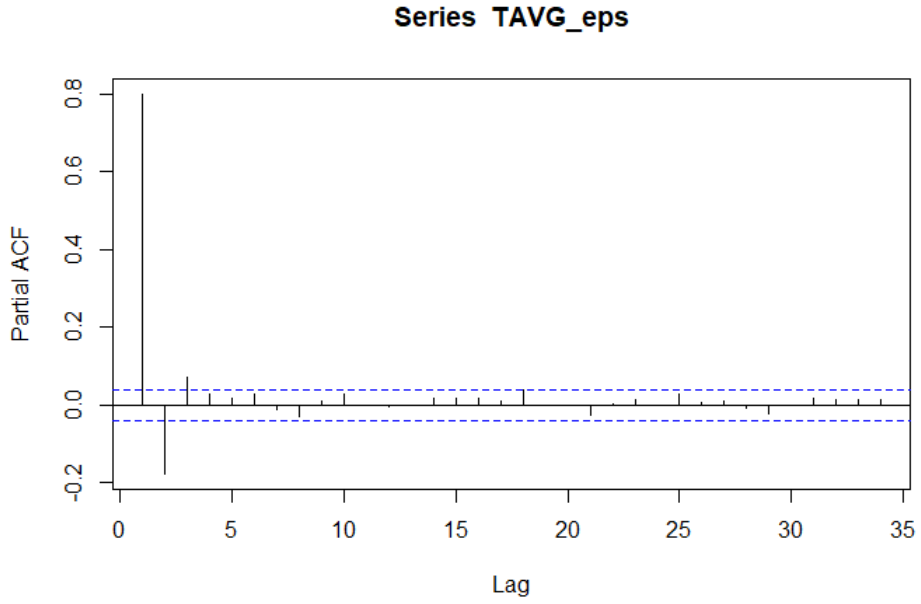
Par contre,étant donné qu'il y a une saisonnalité, par définition, notre série ne peut pas être stationnaire.Cependant, nous pouvons la désaisonnaliser et étudier la stationarité de la série $Y_t - s_t - \mu = X_t$.

4.3.2 Etude de la stationnarité du processus X_t

On considère désormais le processus alétoire $X_t = Y(t) - s(t) - \mu$ associé à notre signal. On réalise un diagragramme d'autocorrélation partielle (ACF) pour ce processus X_t. On remarque que la suite des coefficients de l'ACF décroit de manière exponentielle, ce qui indique, par caractérisation,que le processus ϵ peut être modélisé par un ARIMA(p,d,q).



On exploite ensuite le diagramme PACF, ce qui nous indique que le processus ARIMA correspond en fait à processus ARMA(1,1).



Nous allons poursuivre la modélisation de la partie stationnaire par un processus ARMA.D'après notre étude du PACF, le processus peut être modélisé par un ARMA,notamment un ARMA (1,1), ie la partie AR est un AR(1) et la partie MA est un MA(1).

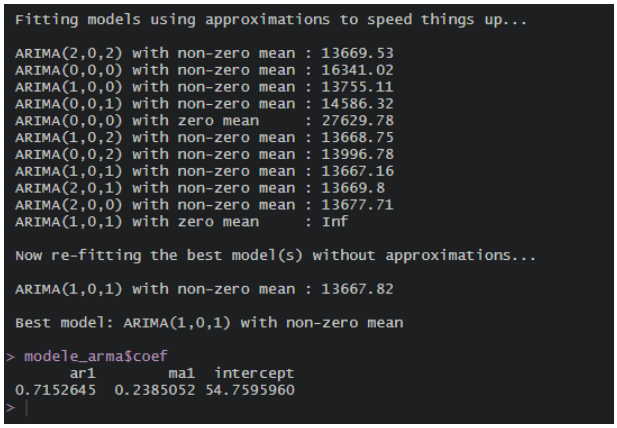
Pour confirmer ce choix de modélisation, on va utiliser la fonction auto.arima implémentée en R.Notons bien que le modèle ARIMA(p,d,q) est plus générique que les modèles ARMA(p,q) : ainsi les processus ARMA(p,q) sont des processus ARIMA dégénéres de la forme ARIMA (p,0,q).

On a déterminé précedemment que notre série ne présentait pas de tendance.On peut confirmer ou infirmer cela à l'aide de la modélisation ARIMA :en effet, si le meilleur modèle trouvé est un vrai ARIMA(p,d,q) avec d différend de 0 , alors l'ordre de différenciation d est non nul, donc la série présente une tendance polynomiale d'ordre d.

Au contraire, si auto.arima fournit un modèle ARMA, alors cela est cohérent avec le fait que la série ne présente pas de tendance puisque que l'ordre de différentiation est 0.

On peut également établir, grâce à un test de stationnarité (test KPSS), que le processus X_t, modélisant la partie aléatoire de la série ,est stationnaire.

En effet, dans la documentation de auto.arima(), il est indiqué que dans un premier temps, c'est la stationnarité de la série qui est étudiée.Plusieurs tests de type KPSS sont effectués.Dans un test KPSS,l'hypothèse nulle est que la série est stationnaire.Si l'hypothèse nulle est rejetée, alors d'après le test KPSS la série n'est pas stationnaire, donc l'algorithme différencie notre série pour obtenir une série stationnaire.On peut donc en déduire que si aucun ordre de différenciation n'a été effectué,alors l'hypothèse nulle dans le test KPSS n'a jamais été rejetée, donc par conséquent la série est stationnaire.



auto.arima indique que le meilleur modèle est un ARMA(1,1), selon le critère de l'AIC. On interprète :
$ar1 = \varphi = 0.715$

$ma1 = \theta = 0.715$ On indique qu'on a appliqué auto.arima au processus $X_t + \mu$ donc il est cohérent de trouver un intercept qui vaut $\mu = 54.8$.Dans la suite,on considère bien le processus X_t qui lui est centré.

Nous avons donc déterminé un processus ARMA , centré (X_t) qui s'écrit sous la forme :
(quitte à changer φ en -φ)

$\forall t X_{t+1} - \varphi X_t = \epsilon_t + \theta \epsilon_{t-1}$

Dans cette décomposition, ϵ_t est un processus d'innovations.

Comme dans le cours, on introduit deux filtres Φ et Θ, qui utilisent l'opérateur de retard L
(où L est l'opérateur défini par : $LX_t = X_{t-1}$)
On pose donc : $\Phi(L) = 1 - \varphi L$
$\Theta(L) = 1 + \theta L$ de sorte que :
$\Phi(L)X_t = 1 - \varphi X_{t-1}$

$\Theta(L)\epsilon_t = 1 + \theta \epsilon_{t-1}$
Ainsi, on peut décrire de façon plus compacte le processus ARMA(1,1) :

$\Phi(L)X_t = \Theta(L)\epsilon_t, \forall\, t \in Z$

Le processus ARMA(1,1) que l'on veut utiliser dans le cadre de notre modèle doit posséder de bonnes propriétés si l'on veut qu'il soit réellement utile :

1) il doit être causal
2) il doit être inversible

Vérifions la propriété de causalité :
Le processus X_t solution de l'équation définissant le modèle ARMA(1,1) vérifie :

$X_t = \frac{1}{\Phi(L)}\Theta(L)_t$
Puisqu'on a déterminé à l'aide de auto.arima que $|\varphi| = 0.715 < 1$, on peut utiliser un développement en série entière :

$\frac{1}{\Phi(L)}\theta(L) = (1 + \varphi L + \varphi^2 L^2 + \varphi^3 L^3 + ...)(1 + \theta L) = 1 + \varphi L + \varphi^2 L^2 + \varphi^3 L^3 + ... + \theta L + \varphi\theta L^2 + \varphi^2\theta L^3 + \varphi^3\theta L^4 + ...$

$\frac{1}{\Phi(L)}\theta(L) = 1+(\varphi+\theta)L+(\varphi^2+\varphi\theta)L^2+(\varphi^3+\varphi^2\theta)L^3+... = 1+(\varphi+\theta)L+(\varphi+\theta)^2+(\varphi+\theta)\varphi^2L^3+....$
$\frac{1}{\Phi(L)}\theta(L) = \sum_{j=0}^{+\infty}\psi_j L^l$

où la suite des ψ_j est définie ainsi : $\psi_0 = 1$ et : $\psi_j = (\phi + \theta)\psi^{j-1}$,pour $j > 0$
on a donc la décomposition suivante pour le processus X_t :

$\mathrm{X}_t = \epsilon_t + (\varphi + \theta)\sum_{j=1}^{+\infty}\varphi^{j-1}\epsilon_{t+j}$

Finalement, on peut décomposer la série pour se ramener au cadre du théorème de Wold :

$X_t = \sum_{j=1}^{+\infty}\psi_j\epsilon_{t-j}$.

En fait, le processus que l'on veut modéliser par un ARMA est centré et stationnaire au second ordre, donc d'après le théorème de Wold on peut déjà afirmer qu' il peut se décomposer sous la forme vue précedemment.Grâce à nos calculs, on a explicité la suite des ψ_j.Par unicité du développement en série entière, la suite des ψ_j qui apparait dans le théorème de Wold est identique à celle que nous avons calculée.

Arrivé à ce stade, on a démontré la propriété de causalité, c'est-à-dire qu'on peut explicitement calculer X_t en fonction des réalisations passées du processus ϵ_t.
De facon symétrique, la propriété d'inversibilité permet de déterminer $_t$en fonction des réalisations passées du processus $_t$. On rappelle l'équation définissant le modèle ARMA(1,1)
$\Phi(L)X_t = \Theta(L)_t, \forall\, t \in Z$
donc :
$\epsilon_t = \frac{1}{\Theta(L)}\Phi(L)X_t = \frac{1}{1+\theta L}(1 - \Phi B)X_t$
Or, on a déterminé à l'aide de auto.arima() que : $|\theta| = 0.238 < 1$. Considérons la fonction $f(x) = \frac{1}{1+\theta x}$ = Alors f est développable en série entière pour notre valeur de θ On a le développement en série entière :
$f(x) = \frac{1}{1+\theta x} = \sum_{j=0}^{\infty}(-\theta)^j x^j$
$\frac{1}{1+\theta L} = \sum_{j=0}^{\infty}(-\theta)^j L^j$
Donc, en composant par l'opérateur L, on a :
$\frac{1}{1+\theta L} = \sum_{j=0}^{\infty}(-\theta)^j L^j$
Ainsi : $\epsilon_t = \sum_{j=0}^{\infty}(-\theta)^j L^j(1 - \Phi L)X_t$

$\epsilon_t = X_t - (\varphi + \theta)\sum_{j=1}^{\infty}(-\theta)^{j-1}X_{t-j}$

On remarque ici que la propriété d'inversibilité est vérifiée si $|\theta| < 1$, ce qui est le cas dans notre ARMA $|\theta| = 0.238$ On conclue donc que notre ARMA est inversible.

On conclue donc ici en disant que le processus ARMA(1,1) est un processus stationnaire qui admet une décomposition par le théorème de Wold.On a explicité cette décomposition et utilisé l'unicité du développement en série entière.Dès lors, on peut prédire les futures réalisations du processus X_t en fonction uniquement du processus d'innovation .C'est la propriété de causalité. De facon symétrique , on a montré que notre ARMA vérifiait la propriété d'inversibilité.

Les deux propriétés qui sont utiles pour notre modélisation , la causalité et l'inversibilité, sont vérifiées pour des valeurs de ϕ et de θ qui sont dans une région carrée :

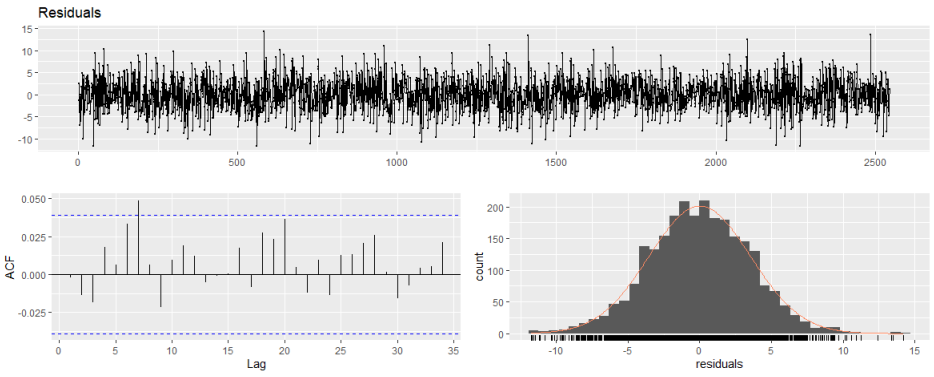
$-1 < \varphi < 1$ et $-1 < \theta < 1$

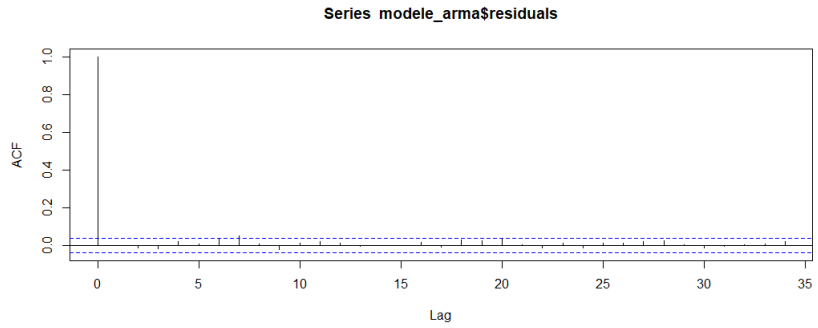
4.4 Analyse des résidus du modèle ARMA

```
> ####################
> Box.test(modele_arma$residuals, lag =5, type = "Box-Pierce", fitdf = 2)

        Box-Pierce test

data:  modele_arma$residuals
X-squared = 2.272, df = 3, p-value = 0.5179
```





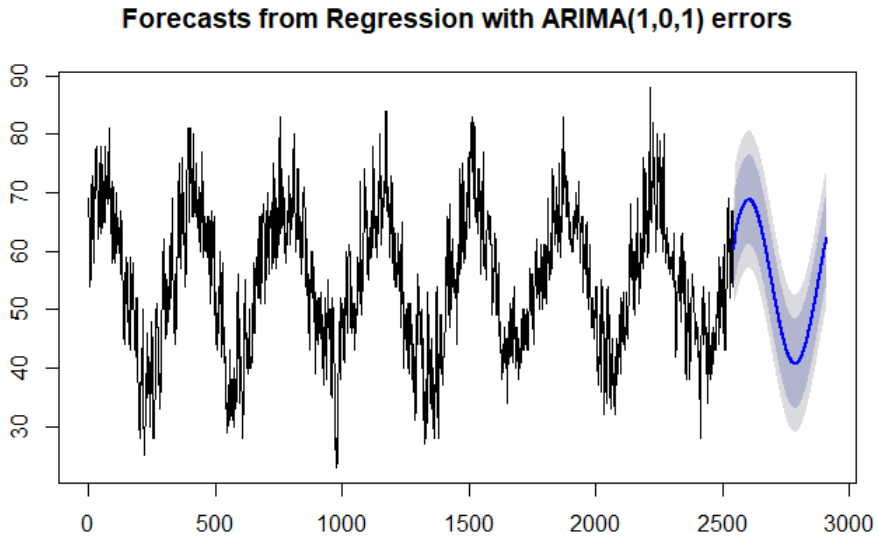
On doit étudier de manière rigoureureuse la validité de notre modélisation du processus par un ARMA(1,1).Nous allons donc étudier les résidus du modèle ARMA.Ces résidus ne doivent contenir aucune information.Les résidus doivent de plus être décorréles.On le met en évidence grâce à l'ACF qui ne comporte aucune autocorrélation significative pour h>0.Cela prouve que les résidus sont décorrélés. On peut également vérifier si les résidus sont gaussiens.On vérifie que les résidus ont l'allure d'un bruit gaussien.

Sur l'ACF, on remarque que les autocorrélations partielles sont négligeables ce qui confirme le caractère non corrélé des résidus.On observe également que l'histogramme des réalisations du bruit gaussien peut être approché par une densité gaussienne. Enfin, pour montrer que les résidus sont bien décorrélés, on peut utiliser un test de Box-Pierce. Dans le cadre de ce test, l'hypothèse nulle est :il n'y a pas d'autocorrélation des résidus de l'ordre 1 à l (ici l=5).

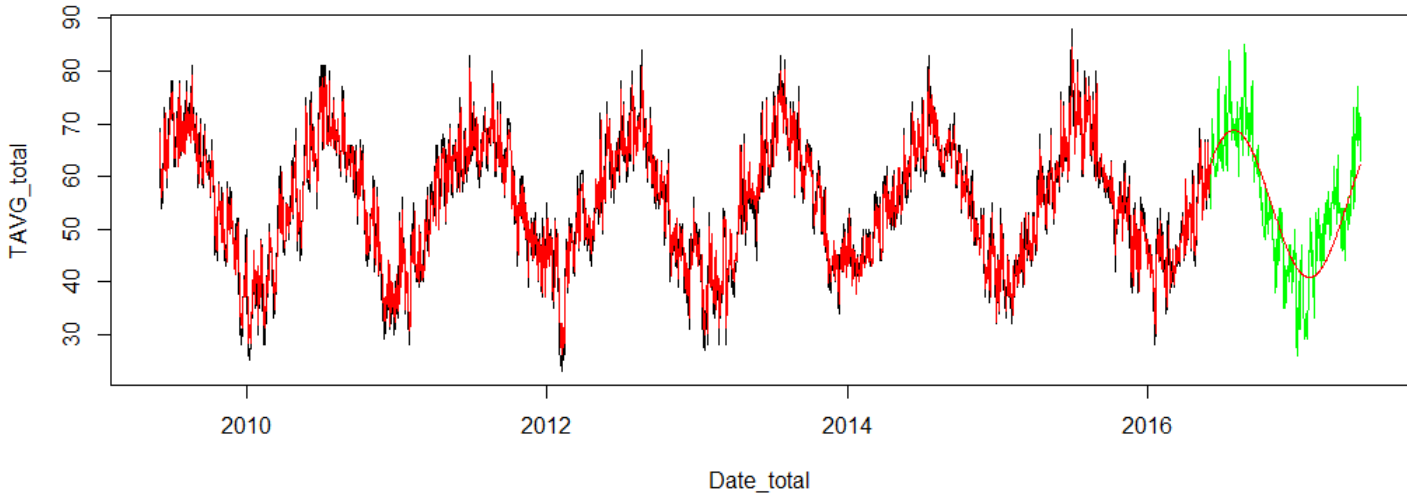
On obtient une p-value de 0.5179 , non significative, donc on ne rejette pas l'hypothèse nulle, donc par le test de Box-Pierce les résidus sont bien décorrélés (de l'ordre 1 à 5).

4.5 Prévision avec le modèle ARMA(1,1)

On peut désormais faire de la prévision à l'aide de notre modèle ARMA(1,1).En effet, on a montré que ce processus modélisait fidèlement nos données.De plus, notre modèle ARMA(1,1) possède les propriétés de causalité et d'invesibilité, ce qui permet de prévoir ses futures réalisations.La prévision fournie par forecast est accompagnée par des intervalles de confiance à 80/100 et 95/100, en nuances de gris. Puisque les résidus suivent une loi gaussienne, il est possible de construire des intervalles de confiance.

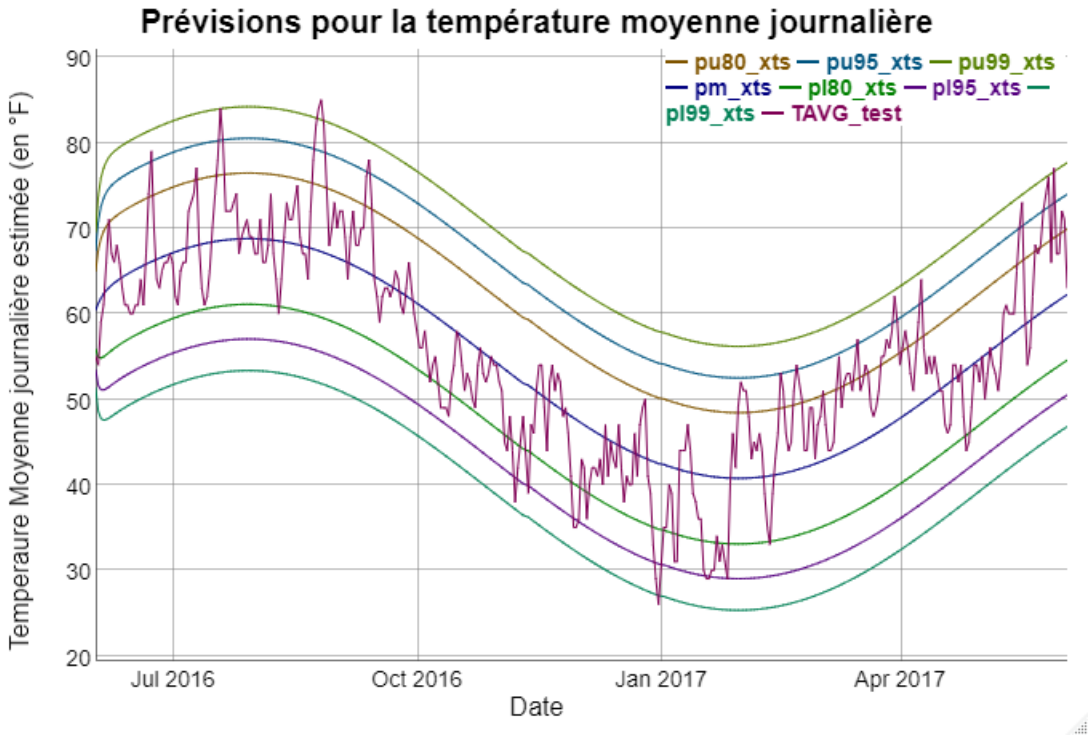


Cette prévision a été assez ardue à implémenter car il faut prendre en compte l'effet de la saisonnalité.(On rappelle que le modèle ARMA concerne uniquement la partie désaisonalisée de la série).On a choisi de considérer les termes de la décomposition en série de Fourier en tant que régresseurs externes lors de la prévision.Ainsi,la prévision du modèle ARMA tient compte de la saisonnalité et la prévision est fidèle à la réalité.

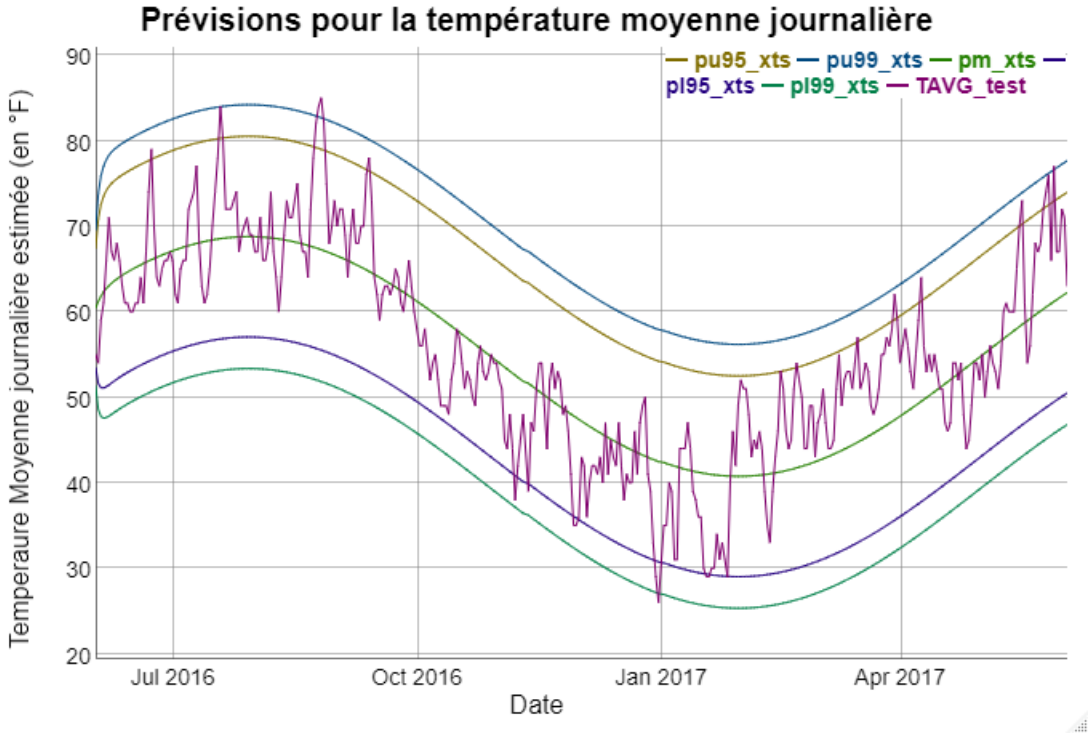


Sur ce graphe, en noir sont représentés les températures de l'échantillon d'apprentissage, en vert les températures de l'échantillon de test.En rouge est représenté le modèle ARMA,sur la période d'apprentissage, puis la prévision moyenne.

En fait,on peut construire explicitement plusieurs intervalles de confiance car nos résidus sont gaussiens.On a donc choisi de créer 3 intervalles de confiance, de niveau 0.80 0.95 et 0.99. La température réelle est bien éstimée car la quasi-totalité des températures est dans la région de confiance à 0.99. Considérer un intervalle de confiance à 0.80 est peu intéressant dans notre cas car trop de valeurs extrêmes en sont exclues.



Finalement,on conserve les intervalles de prédiction à 0.95 et 0.99 qui sont plus pertinents.Cependant, il faut remarquer que le RMSE augmente lorsque la largeur de l'intervalle de confiance augmente Ainsi : $RMSE_{pm} = 6.4$; $RMSE_{pu80} = 9.5$; $RMSE_{pu95} = 12.8$; $RMSE_{pu99} = 16.1$ Il faut donc fixer un compromis entre largeur de l'intervalle de confiance et valeur du RMSE.



5 Conclusion

Au cours de ce projet, nous avons tout d'abord mis en forme les données pour étudier la température moyenne journalière au Parc Montsouris sous la forme d'une série temporelle.Après une analyse descriptive, nous avons étudié la décomposition en tendance, saisonnalité et partie aléatoire. A ce stade de l'étude, on a postulé que :

$Y_t = f_t + S_t + X_t$
On a montré que la tendance était simplement une constante :$f_t = \mu = 54.8$

$Y_t = \mu + S_t + X_t$
Nous avons ensuite déterminé la saisonnalité en réalisant une décomposition en série de Fourier de la série, ce qui a été implémenté sous la forme d'une régression sur une base trigonométrique, de période (365).L'analyse de la régression a montré que seuls deux coefficients étaient significatifs

Désormais : $Y_t = \mu + acos(\omega t) + bsin(\omega t) + X_t$
On a donc mis au jour la partie déterministe (tendance et saisonalité) de notre série.Il reste à étudier le processus aléatoire X_t. On a montré que ce processus était stationnaire, et que c'était un processus de type ARMA(1,1).Ce processus possède les propriété de causalité et d'inversibilité.En faisant le lien avec le théorème de Wold , on a prouvé que ce processus était prévisible à l'aide des réalisation du processus d'innovation. L'analyse des résidus a montré que les résidus correspondaient à un bruit gaussien, donc ne contenant pas d'information utiles pour étudier la série.

Dès lors, on peut appliquer les techniques de prévision à l'aide du package forecast.Les résultats de prévisions sont concluants.

En utilisant le RMSE comme métrique, on peut même affirmer que les prévisions fournies par le modèle ARMA sont meilleures que celles fournies par le lissage exponentiel.

En effet, $RMSE_{ARMA,pm-xts} < RMSE_{Holt-Winters}$ ($RMSE_{Holt-Winters} = 12.5$ alors que $RMSE_{ARMA} = 6.4$)

Nous avons mené notre étude en améliorant à chaque étape notre RMSE pour proposer une prévision plus proche des mesures réelles.

Le modèle ARMA permet même de fournir une représentation analytique de la partie aléatoire de la série, grâce au théorème de Wold,alors que la méthode de Holt-Winters ne permet pas,à notre connaissance, de fournir une telle représentation.Pour poursuivre notre étude, on pourrait considérer les relevés de température extrêmes TMIN et TMAX et vérifier si notre modélisation est encore valide.