

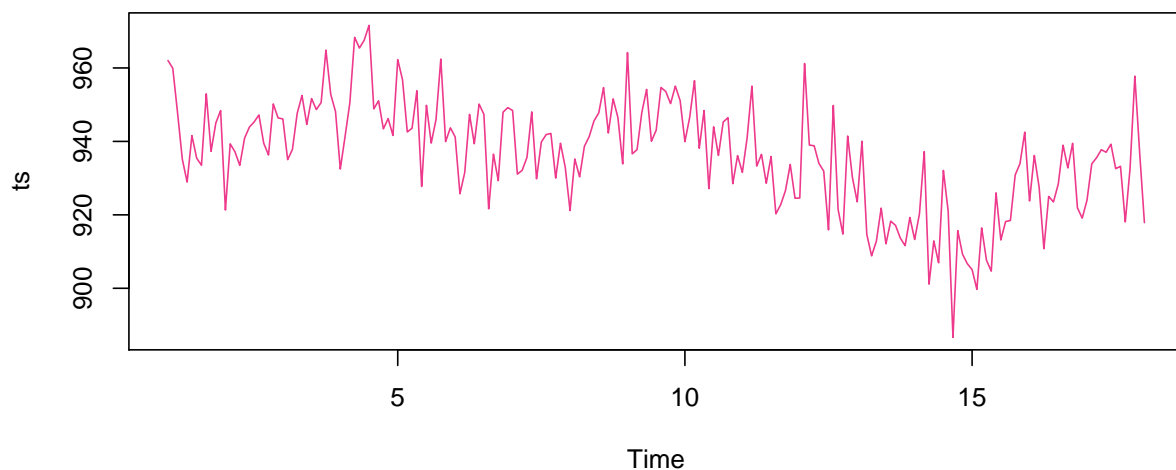
# Analyse d'une série temporelle

Lucie Guillaumin

## Création de la série temporelle

```
donnees = scan('serie_20.dat')  
  
ts = ts(donnees, frequency = 12)  
ts.plot(ts, main = 'Affichage de notre série temporelle ts', col = 'violetred2')
```

**Affichage de notre série temporelle ts**

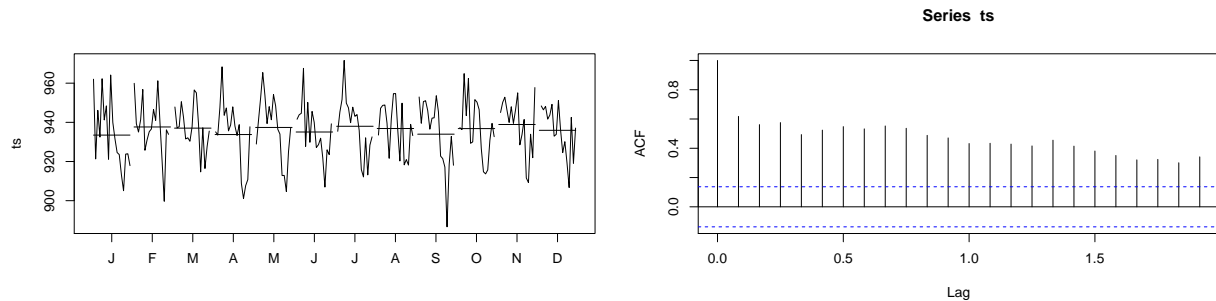


Nous allons étudier une série temporelle que nous avons nommé **ts** : elle est composée de 205 observations et sa période est de 12.

De plus, à première vue, la série temporelle ne semble pas stationnaire : elle présente une tendance ainsi qu'une saisonnalité, on remarque aussi qu'elle oscille entre 880 et 990.

## Analyse qualitative de notre série

```
par(mfrow = c(1,2))
monthplot(ts)
acf(ts)
```



On remarque ici à l'aide de la fonction `monthplot` l'existence d'une saisonnalité.

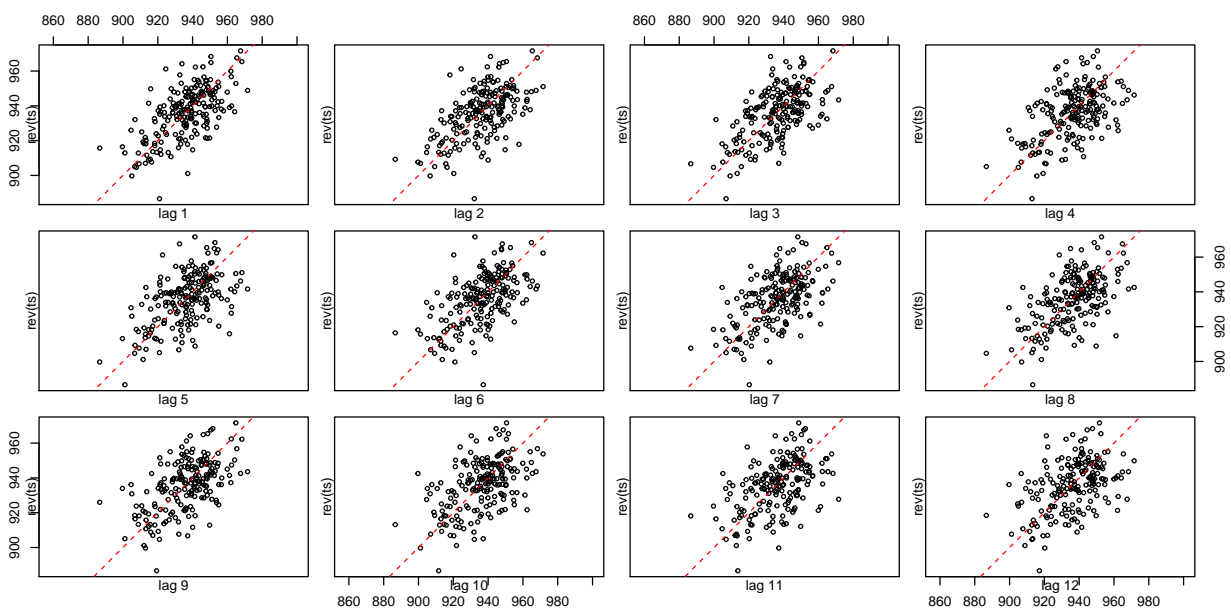
De plus, l'auto-corrélogramme de notre série temporelle nous confirme que notre série n'est pas stationnaire : ce qui est logique car elle possède une saisonnalité ainsi qu'une tendance.

Nous allons maintenant afficher le `lag.plot` de notre série temporelle qui est un diagramme de dispersion des points ayant pour abscisse la série retardée de  $k$  ( $=12$  : notre période) instants et pour ordonnée la série non retardée.

De plus, nous allons utiliser la fonction `rev` pour retourner la série (la fonction `lag` ne donne pas la série retardée mais celle avancée).

On cherche donc à comprendre la dépendance de la série par rapport à son passé.

```
lag.plot(rev(ts),12, layout = c(3,4), diag.col = 'red')
```



D'après le graphique, on remarque que les auto-corrélations sont globalement forte avec tous les retards.

# Méthode d'élimination jointe de la tendance et de la saisonnalité à X

## Décomposition de la tendance

```
P = 12
q = P/2

long=length(ts)
tendance=rep(NA, long)

for(t in (1+q):(long-q)){tendance[t]= 1/P*(0.5*ts[t-q]+sum(ts[(t-q+1):(t+q-1)])+0.5*ts[t+q])}

tendance=ts(tendance, frequency = 12)

sanstend = ts - tendance
```

## Décomposition de la saisonnalité

```
sais = rep(NA, 12)
N = floor(long/P)

for(s in 1:P){sais[s]=mean(sanstend[s+((1:N)-1)*P], na.rm=TRUE)}

saisfin=sais-mean(sais)

saisfin=c(rep(saisfin,N), sais[1])
saisonnalite=ts(saisfin, frequency=12)

sans_sais = ts - saisonnalite
```

## Décomposition du bruit

```
residus = ts - (tendance + saisonnalite)
```

Nous allons maintenant estimer la tendance et la saisonnalité.

## Estimation de la tendance

```
tend2=rep(NA, long)

for(t in (1+q):(long-q)){
  tend2[t]= 1/P * ((1/2)*sans_sais[t-q]+sum(sans_sais[(t-q+1):(t+q-1)])+(1/2)*sans_sais[t+q])}

estim_tend = ts(tend2, frequency = 12) #estimation de la tendance

sanstend2 = ts - tend2
```

## Estimation de la saisonnalité

```
sais2=rep(NA,12)

for(s in 1:P){sais2[s]=mean(sanstend2[s+((1:N)-1)*P],na.rm=TRUE)}

saisfin2=sais2-mean(sais2)

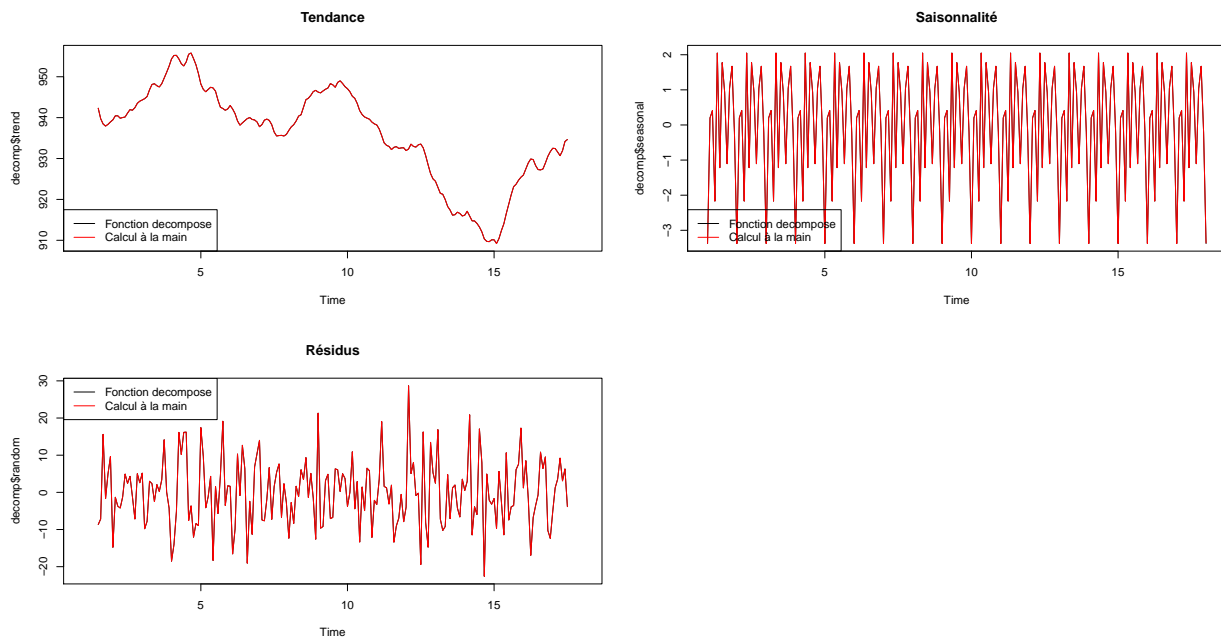
saisfin2=c(rep(saisfin2,N))
estim_saisonnalite=ts(saisfin2, frequency = 12) #estimation de la saisonnalité
```

## Comparaison des résultats avec la fonction decompose :

Nous allons utiliser la fonction `decompose` afin de vérifier que nos calculs sont bon :

```
decomp = decompose(ts)

par(mfrow=c(2,2));
plot(decomp$trend, main = 'Tendance'); lines(tendance, type = 'l', col = 'red')
legend('bottomleft',col = c('black','red' ), legend = c('Fonction decompose','Calcul à la main'), lty =
plot(decomp$seasonal, main = 'Saisonnalité'); lines(saisonnalite, type = 'l', col = 'red')
legend('bottomleft',col = c('black','red' ), legend = c('Fonction decompose','Calcul à la main'), lty =
plot(decomp$random, main = 'Résidus'); lines(residus, type = 'l', col = 'red')
legend('topleft',col = c('black','red'), legend = c('Fonction decompose','Calcul à la main'), lty = 1.2
```



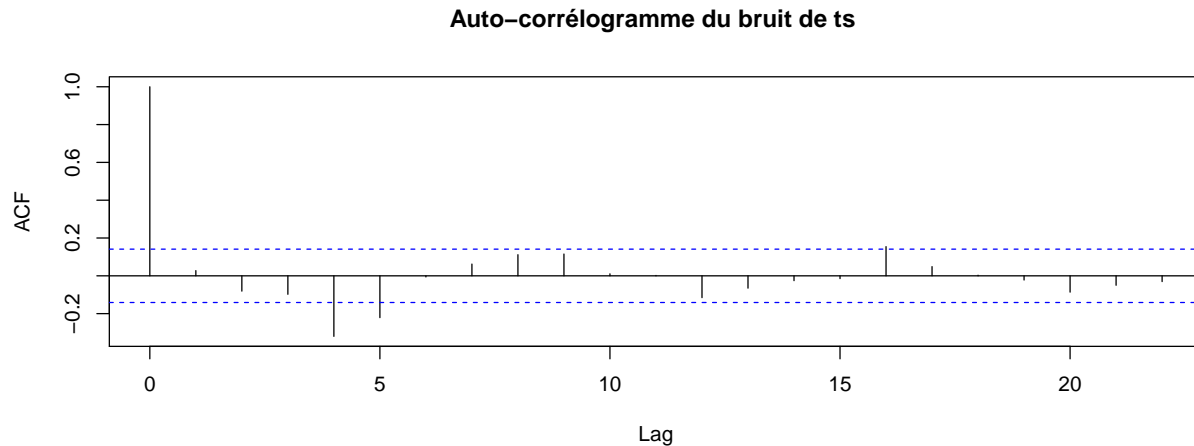
On remarque que nous avons fait un bon travail :

En effet, on ne remarque pas de grosses différences; les tracés de la tendance, de la saisonnalité et des résidus se confondent : notre décomposition 'à la main' est donc bonne.

## Bruit stationnaire? Proposition et estimation de modèle ARMA pour le résidu.

On cherche maintenant à savoir si le bruit de notre série temporelle est stationnaire. Pour cela, nous allons tracer l'auto-corrélogramme de notre bruit :

```
residus = residus[-which(is.na(residus))] #on enlève les valeurs manquantes
acf(residus, main = 'Auto-corrélogramme du bruit de ts ')
```



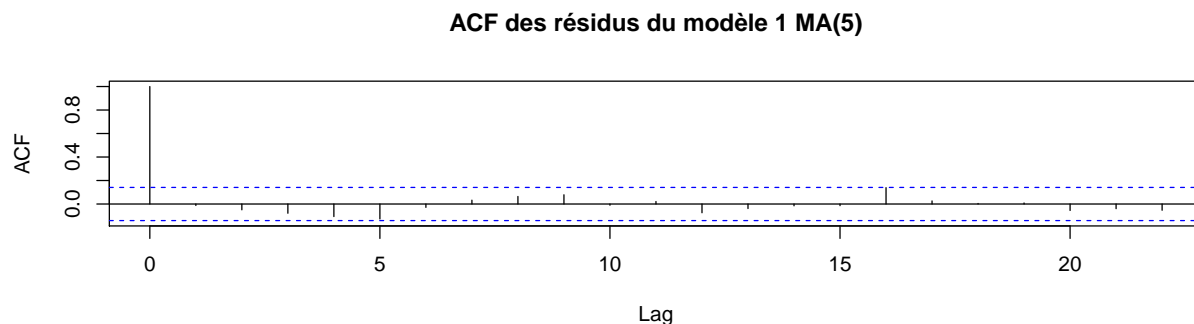
D'après l'auto-corrélogramme, on remarque une décroissance rapide des coefficients : on est dans le cas d'un processus stationnaire.

De plus, on remarque que  $\rho(4)$  et  $\rho(5)$  dépassent la borne de l'intervalle ainsi que  $\rho(16)$  qui le dépasse très légèrement, on peut donc considérer pour  $h=16$  que  $\rho(h) = 0$  (le pic n'est pas significatif). Pour le reste des coefficients, on peut considérer qu'il n'y a pas d'autocorrélation dans les résidus en effet les autres valeurs sont comprises dans l'intervalle de confiance à 95% tracé en bleu sur notre corrélogramme.

On sait que lorsque l'on regarde l'ACF, on peut proposer un modèle  $MA(q)$  lorsque  $\rho(h) = 0$  pour  $q+1$ . Il serait donc exagéré de proposer un modèle  $MA(15)$  ce qui ferait beaucoup trop de coefficients à estimer.

On souhaite donc proposer un modèle  $MA(5)$ , en effet pour  $h=6$   $\rho(h) = 0$ , donc ici  $q+1 = 6$  on prend ainsi  $q = 5$ . Nous avons donc 6 coefficients à estimer :

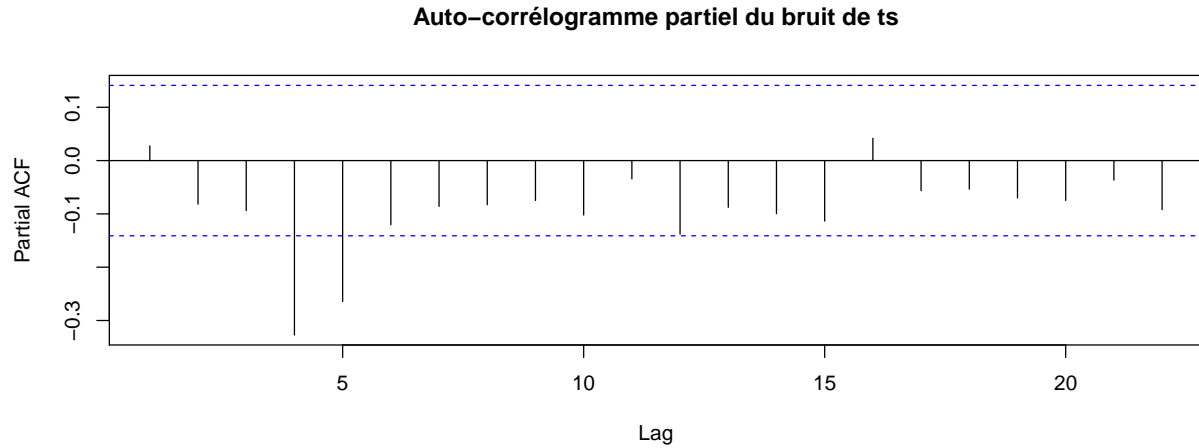
```
modele1 = arima(residus, order = c(0,0,5))
acf(modele1$residuals, main = 'ACF des résidus du modèle 1 MA(5)')
```



$\rho(0) = 1$  et toutes les autres valeurs de  $\rho_n$  sont dans l'intervalle de confiance, ils sont donc considérés comme nuls, ainsi on accepte la blancheur des résidus.

Interprétons maintenant l'auto-corrélogramme partiel de notre résidu.

```
pacf(residus, main = 'Auto-corrélogramme partiel du bruit de ts')
```

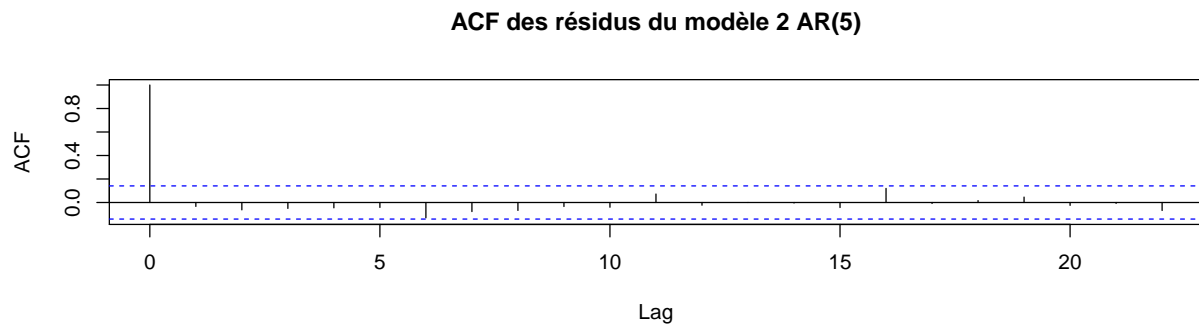


D'après l'auto-corrélogramme partiel, on remarque que nous avons deux pics à  $\tau(4)$  et  $\tau(5)$  qui sortent de l'intervalle de confiance à 95%. Concernant les autres valeurs de  $h$ , tous les pics sont inclus dans l'intervalle donc considéré comme nulles.

On sait que lorsque l'on regarde la PACF, on peut proposer un modèle  $AR(p)$  lorsque  $\tau(h) = 0$  pour  $p+1$ . Ici, après  $h=5$  les valeurs de  $\tau$  sont nulles, donc pour  $h=6$   $\tau(h) = 0$ .

On souhaite donc proposer un modèle  $AR(5)$  (ici  $p+1 = 6$  donc  $p=5$ )  
 Estimons les 6 paramètres de ce modèle :

```
modele2 = arima(residus, order = c(5,0,0))
acf(modele2$residuals, main = 'ACF des résidus du modèle 2 AR(5)')
```

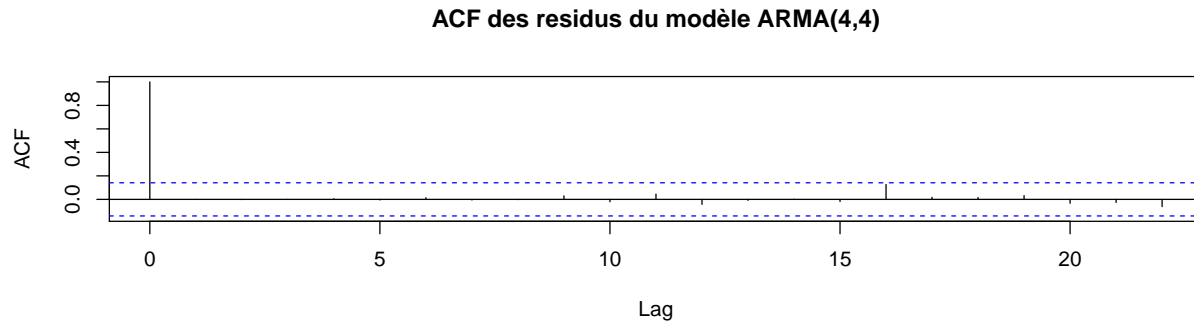


$\rho(0) = 1$  et toutes les autres valeurs de  $\rho_n$  sont dans l'intervalle de confiance, ils sont donc considérés comme nuls, ainsi on accepte la blancheur des résidus.

A l'aide de la méthode du compliqué au plus simple nous allons proposer un modèle ARMA(p,q) qui nous semble le plus optimal :

On part d'un modèle ARMA(4,4), on pense qu'on ne le retiendra pas : trop de coefficients à estimer il y a sûrement un modèle ARMA avec moins de coefficients à estimer.

```
modele4.4 = arima(residus, order = c(4,0,4))
acf(modele4.4$residuals, main = 'ACF des residus du modèle ARMA(4,4)')
```



$\rho(0) = 1$  et toutes les autres valeurs de  $\rho_n$  sont dans l'intervalle de confiance, ils sont donc considérés comme nuls, ainsi on accepte la blancheur des résidus.

On regarde ensuite la significativité de chacun de nos coefficients pour voir si on peut proposer un modèle avec moins de coefficients.

```
confint(modele4.4)
```

##		2.5 %	97.5 %
##	ar1	0.15128559	1.34916273
##	ar2	-1.28681198	0.45262729
##	ar3	-0.73435472	0.89623536
##	ar4	-0.53771156	0.19735100
##	ma1	-1.60459336	-0.40093495
##	ma2	-0.66241935	1.33490605
##	ma3	-1.11802334	0.75715173
##	ma4	-0.60096712	0.29488945
##	intercept	-0.03790279	0.03622902

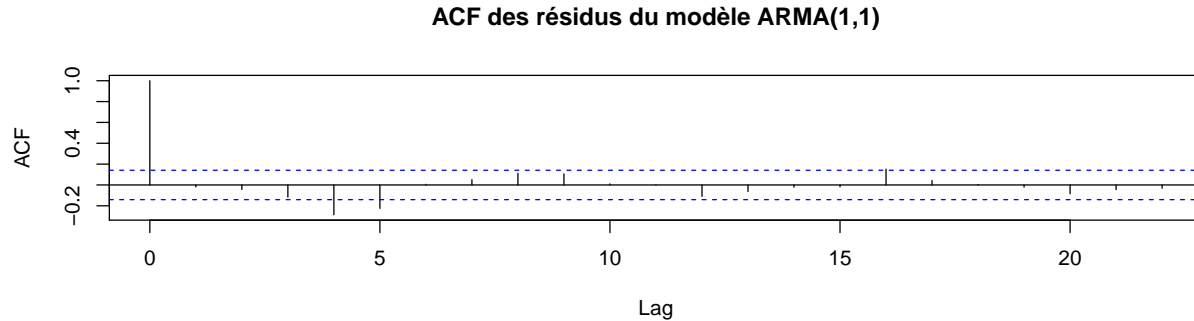
La fonction `confint` nous permet de choisir et donc d'enlever les coefficients pour lesquels 0 est dans l'intervalle de confiance. Et ainsi de suite, jusqu'à ne plus pouvoir le faire.

On remarque ici que la valeur 0 est comprise dans les intervalles de confiance pour les coefficients ar2, ar3, ar4, ma2, ma3 et ma4.

On décide donc d'enlever 3 coefficients AR ainsi que 3 coefficients MA.

On propose donc un nouveau modèle ARMA(1,1) :

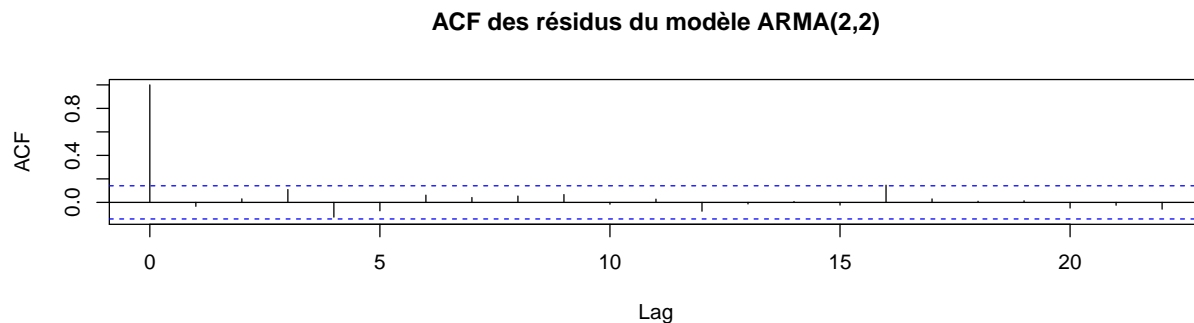
```
modele1.1 = arima(residus,order = c(1,0,1))
acf(modele1.1$residuals, main = 'ACF des résidus du modèle ARMA(1,1)')
```



On remarque pour le modèle ARMA(1,1) nous avons trois pics qui sortent de notre intervalle de confiance. On préfère donc ne pas considérer le bruit comme blanc. Le modèle ARMA(1,1) n'est donc pas optimal, nous ne le retiendrons pas.

Proposons maintenant un modèle ARMA(2,2) :

```
modele2.2 = arima(residus,order = c(2,0,2))
acf(modele2.2$residuals, main = 'ACF des résidus du modèle ARMA(2,2)')
```



$\rho(0) = 1$  et toutes les autres valeurs de  $\rho_n$  sont dans l'intervalle de confiance (on voit quand même que pour  $h=16$  le pic dépasse très légèrement mais on le considère comme non significatif), ils sont donc considérés comme nuls, ainsi on accepte la blancheur des résidus.

On regarde ensuite la significativité de chacun de nos coefficients pour voir si on peut proposer un modèle avec moins de coefficients.

```
confint(modele2.2)
```

##		2.5 %	97.5 %
##	ar1	1.04535888	1.52795646
##	ar2	-0.72658336	-0.39752794
##	ma1	-1.76834720	-1.23367223
##	ma2	0.23615073	0.76588657
##	intercept	-0.03570437	0.03502712

On remarque que nos coefficients sont significatif, on a trouvé le modèle ARMA le plus optimal.



**MA(5)? AR(5)? ARMA(4,4)? ARMA(2,2)?**

On souhaite maintenant savoir quel modèle sera le plus intéressant pour effectuer notre prévision. Pour ces quatre modèles, nous avons accepté la blancheur de leurs résidus respectifs. Il faut donc que nous comparions leur AIC respectifs.

```
aic = c(modele1$aic, modele2$aic, modele4.4$aic, modele2.2$aic)
names(aic) = c('MA(5)', 'AR(5)', 'ARMA(4,4)', 'ARMA(2,2)')
aic
```

```
##      MA(5)      AR(5) ARMA(4,4) ARMA(2,2)
## 1353.802 1366.235 1351.394 1353.200
```

Nous ne pensons pas retenir le modèle le modèle AR(5) : en effet, il a 6 coefficients à estimer et son AIC est le plus grand. De plus, le modèle ARMA(4,4) a 9 coefficients à estimer et malgré le fait que son AIC soit le plus petit, nous préférons prendre un modèle avec moins de coefficients à estimer : nous retenons donc pas ce modèle.

Pour finir, les modèles MA(5) et ARMA(2,2) possèdent des AIC très proches.

On décidera à l'aide de la prédiction, dans la partie suivante, quel modèle il est le plus judicieux de prendre entre le modèle MA(5) et ARMA(2,2). On regardera, de plus, si le modèle AR(5) prédit mieux les données que les autres modèles que nous avons retenus même si il a le plus grand AIC.

## Prévision

On commence par couper notre série temporelle `ts` en deux : une partie train avec les 200 premières observations qui serviront à apprendre le modèle et sur laquelle nous ferons la prédiction, et une partie de test avec les 5 dernières afin de vérifier que le modèle choisit est le bon.

```
res = ts(residus, frequency = 12)

res_train = window(res, 1, 16.6)
res_test = window(res, 16.6, 17)
```

On prédit maintenant nos 5 valeurs à partir des modèles ARMA(2,2), MA(5) et AR(5).

```
modele2.2 = arima(res_train, order = c(2,0,2))
pred2.2 = predict(modele2.2, n.ahead = 5)

modele5.0 = arima(res_train, order = c(5,0,0))
pred5.0 = predict(modele5.0, n.ahead = 5)

modele0.5 = arima(res_train, order = c(0,0,5))
pred0.5 = predict(modele0.5, n.ahead = 5)

preds = data.frame('ARMA(2,2)'=pred2.2$pred, 'MA(5)'=pred0.5$pred, 'AR(5)' = pred5.0$pred, 'Vraies valeurs' = res_test)
preds
```

##	ARMA.2.2.	MA.5.	AR.5.	Vraies.valeurs
## 1	5.6485632	4.0114112	2.7839283	3.536801
## 2	5.2338593	5.0328865	6.8972150	9.211679
## 3	3.5626042	3.1514337	3.7417929	3.086193
## 4	1.6494505	0.7403167	-0.6831001	6.310649
## 5	0.1273863	-0.0127021	-2.5137958	-3.844910

On calcule ensuite l'écart quadratique entre les vraies données et l'estimation :

```
ecart2.2 = sqrt(mean((pred2.2$pred-res_test)^2))
ecart5.0 = sqrt(mean((pred5.0$pred-res_test)^2))
ecart0.5 = sqrt(mean((pred0.5$pred-res_test)^2))

ecarts = c(ecart2.2, ecart5.0, ecart0.5)
names(ecarts) = c('Ecart ARMA(2,2)', 'Ecart AR(5)', 'Ecart MA(5)')
ecarts
```

##	Ecart ARMA(2,2)	Ecart AR(5)	Ecart MA(5)
##	3.406331	3.377507	3.561074

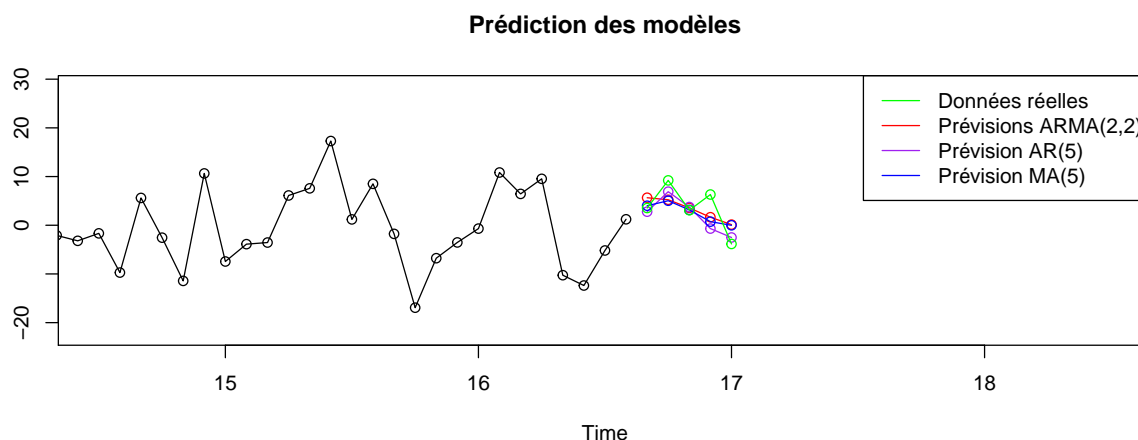
On remarque que l'écart quadratique est plus petit pour le modèle AR(5).

Pour une meilleure prédiction il semble donc préférable de choisir le modèle AR(5). Cependant, on remarque que l'écart quadratique pour les modèles ARMA(2,2) et AR(5) sont quand même très proche (0.03 de différence).

De plus, à la partie précédente, nous avons dit que le modèle ayant le plus petit AIC est le modèle ARMA(2,2). Il est donc préférable de choisir le modèle ARMA(2,2) pour prédire nos données.

Pour finir, nous décidons d'afficher tous ces résultats :

```
#On décide de ne pas afficher nos données sur tout l'ensemble de notre période afin de mieux voir nos prédictions
plot.ts(res_train,type='o',ylab='',xlim = c(14.5,18.5),main='Prédiction des modèles')
lines(pred2.2$pred,col='red',type='o')
lines(pred5.0$pred,col='purple',type='o')
lines(pred0.5$pred,col='blue',type='o')
lines(res_test,col='green',type='o')
legend(x='topright',legend=c('Données réelles','Prévisions ARMA(2,2)','Prévision AR(5)','Prévision MA(5)'))
```



## Conclusion de l'analyse de la série temporelle `ts`

Nous avons vu différentes choses au cours de notre analyse.

Pour commencer, notre série initiale `ts` n'est pas stationnaire : nous ne pouvons donc pas faire grand chose. On décide alors d'éliminer la tendance et la saisonnalité aperçue : nous obtenons donc les résidus de notre série.

A partir de là, nous avons regardé l'auto-corrélogramme de notre résidu et avons remarqué que notre résidu était stationnaire : nous pouvons donc proposer et estimer des modèles ARMA.

Pour cela, on regarde l'auto-corrélogramme ainsi que l'auto-corrélogramme de notre résidu afin de proposer des modèles AR pur ou MA pur. Nous avons vu que nous pouvions retenir un modèle AR(5) et MA(5) car leurs résidus respectifs peuvent être considéré comme blanc.

De plus, à l'aide de la méthode du compliqué au plus simple nous proposons deux modèles à retenir ARMA(4,4) et ARMA(2,2).

Nous avons, ensuite, fait une comparaison des AICs respectifs de chacun de nos modèles proposés : nous retenons ainsi deux modèles pour faire la prédiction : MA(5) et ARMA(2,2). On souhaite quand même regarder la prédiction du modèle AR(5) afin de savoir si il prédit mieux les deux autres modèles proposés même si il a le plus grand AIC.

Pour finir, nous décidons de prédire les 5 dernières valeurs de notre résidu afin de choisir quel modèle est le mieux adapté et le plus optimal. A l'aide du graphique ci-dessus et de l'écart quadratique entre les vraies données et celles estimés avec la prédiction on conclut que le meilleur modèle est le modèle ARMA(2,2).