

Cours de séries temporelles, M2 EURIA

M2 Euria

Pierre Ailliot

Année 2024-2025

Contents

1	Introduction	2
1.1	Séries temporelles : définition et objectifs	2
1.2	Séries temporelles et R	2
1.3	Tendances, composantes périodiques et composantes stationnaires	4
1.4	Plan du cours	5
2	Processus stationnaires	5
2.1	Quelques rappels sur la première partie du cours	5
2.1.1	Processus stationnaires	5
2.1.2	Processus gaussiens	6
2.1.3	Processus ARMA	7
2.1.4	Estimation des fonctions d'autocovariance et d'autocorrélation	8
2.1.5	Tests de bruit blanc	9
2.2	Exercices	12
3	Inférence statistique dans les modèles ARMA	13
3.1	Cas particulier des modèles AR	14
3.1.1	Quelques propriétés des processus AR(p)	14
3.1.2	Fonction d'autocovariance et estimation par la méthode des moments	15
3.1.3	Fonction de vraisemblance	16
3.1.4	Prévision	17
3.1.5	Modèles AR(p) et utilisation de R	19
3.1.6	Exercices sur le paragraphe 3.1	22
3.2	Extension aux modèles ARMA	23
3.2.1	Fonction d'autocovariance	24
3.2.2	Ajustement des modèles ARMA avec R	25

4	Modélisation des composantes non-stationnaires	30
4.1	Différentiation et modèle SARIMA	30
4.2	Modèle paramétrique	35
4.3	Lissage non-paramétrique	37
4.3.1	Moyennes mobiles et fonction <i>decompose</i> de R	37
4.3.2	Régression linéaire locale et fonction <i>stl</i> de R	40
5	References	41

1 Introduction

1.1 Séries temporelles : définition et objectifs

Une **série temporelle** (ou série chronologique) est une suite d'observations indexée par le temps. Dans le cadre de ce cours, nous supposons que les observations sont

- des nombres réels (séries temporelles quantitatives univariées). Nous ne considérerons donc pas les séries temporelles à valeurs entières, qualitatives ou multivariées, par exemple;
- disponibles à temps discret (par opposition aux modèles de processus à temps continu type EDS vus en M1).

Les observations seront notées $(y_t)_{t \in \{1, \dots, T\}}$. On supposera que le temps $t \in \{1, \dots, T\}$ est à valeurs entières pour simplifier les notations. En pratique, l'intervalle de temps entre deux observations (ou "échantillonnage temporel") pourra être la seconde, l'heure, le jour ou le trimestre par exemple. Le nombre d'observations T est la longueur de la série temporelle.

On a généralement en tête les deux objectifs suivants lorsqu'on analyse une série temporelle :

1. obtenir des prévisions pour les valeurs futures $y_{T+1}, \dots, y_{T+h}$ de la série temporelle (on parle de prévisions à horizon h);
2. obtenir une description simple du mécanisme qui a généré les observations $\{y_t\}$. Un tel résumé peut par exemple être utile pour comparer entre elles plusieurs séries temporelles ou simuler de nouvelles réalisations de $\{Y_t\}$ (générateur de scénarios économiques ou climatiques par exemple).

Exemples d'applications possibles en actuariat :

- modéliser l'évolution des coûts de réparation en assurance automobile;
- modéliser l'évolution temporelle d'indices financiers;
- modéliser l'évolution du nombre de sinistres climatiques;
- modéliser l'évolution de l'espérance de vie dans un portefeuille.

1.2 Séries temporelles et R

Une partie importante de ce cours sera dédiée à l'analyse pratique des séries temporelles en utilisant R . La compréhension des codes R donnés en exemple dans ce polycopié est donc importante.

Les objets de type *ts* (initiales de 'time series') sont utilisés pour définir et travailler avec les séries temporelles sous R . La fonction *ts* est la fonction de base pour définir des séries temporelles avec R (taper *?ts* dans R pour en comprendre le fonctionnement).

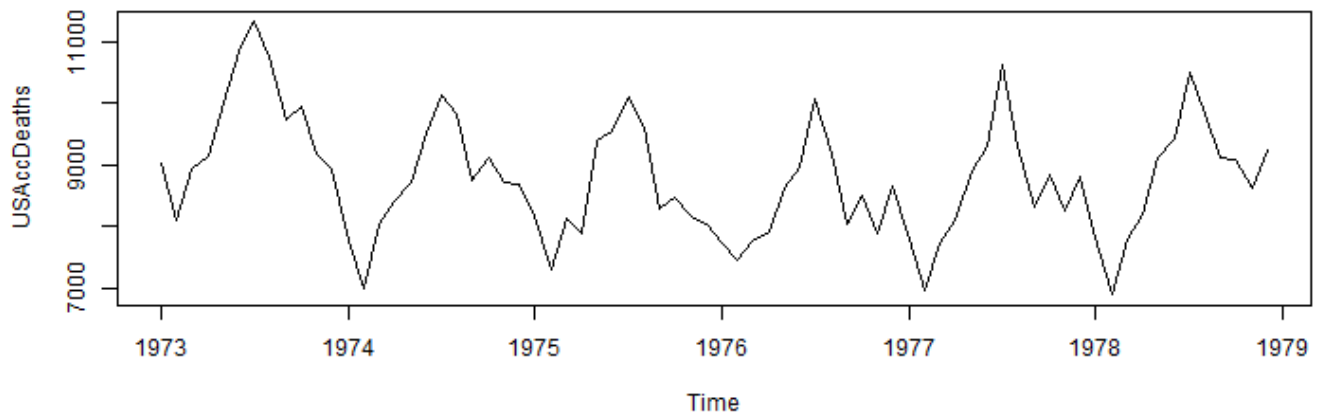
Certaines séries temporelles “classiques” sont disponibles dans R.

Illustration avec R.

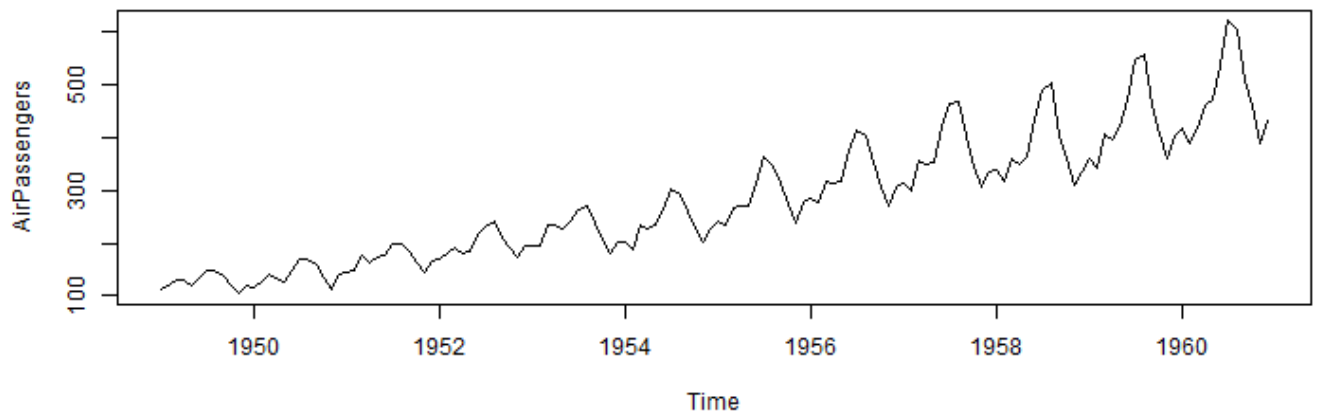
```
str(USAccDeaths) #objet de type 'ts'
```

```
## Time-Series [1:72] from 1973 to 1979: 9007 8106 8928 9137 10017 ...
```

```
plot(USAccDeaths)
```



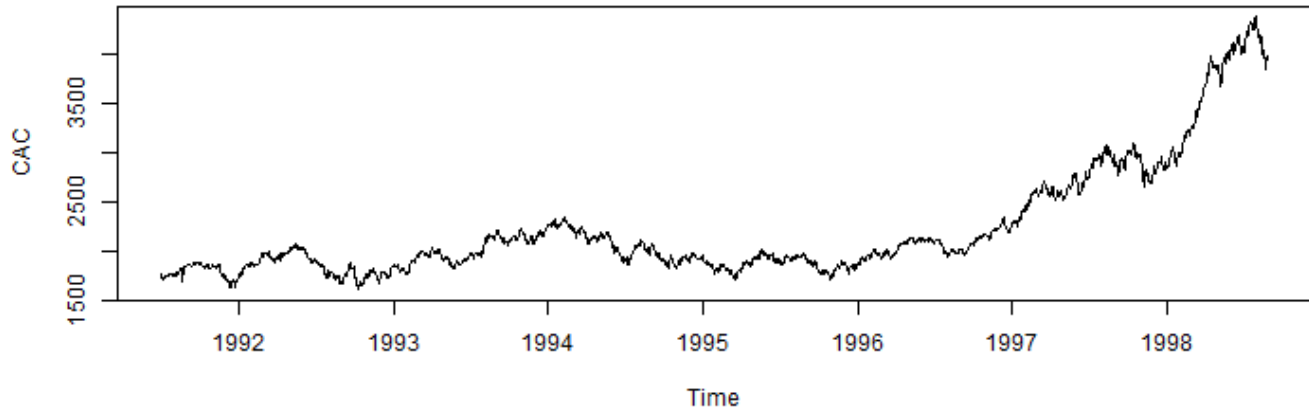
```
plot(AirPassengers)
```



```
str(EuStockMarkets) #objet de type 'mts' avec m pour 'multivariate'
```

```
## Time-Series [1:1860, 1:4] from 1991 to 1999: 1629 1614 1607 1621 1618 ...  
## - attr(*, "dimnames")=List of 2  
## ..$ : NULL  
## ..$ : chr [1:4] "DAX" "SMI" "CAC" "FTSE"
```

```
CAC=EuStockMarkets[,3]
plot(CAC)
```



1.3 Tendances, composantes périodiques et composantes stationnaires

On peut généralement identifier plusieurs composantes sur une série temporelle.

- Une **tendance** qui correspond à l'évolution à long terme (pluriannuelle) des observations. Par exemple, sur les séries temporelles du paragraphe précédent, on peut voir que le nombre de passagers dans les avions a tendance à augmenter d'année en année. Le cours du CAC 40 a également tendance à augmenter, avec des périodes de baisse associées par exemple aux crises financières.
- Des **composantes saisonnières**. Par exemple, sur les séries temporelles du paragraphe précédent, on peut voir qu'il y a plus de morts accidentelles en été que en hiver aux Etats-Unis et que le nombre de passagers dans les avions est également plus important en été. On parle alors de **fluctuations saisonnières**. Pour certaines séries temporelles (par exemple les températures ou le trafic routier), on peut également identifier des **fluctuations journalières** avec des valeurs plus fortes à certaines heures de la journée.
- Un résidu qui semble avoir un comportement "aléatoire".

Afin de modéliser ces différentes composantes, on utilise généralement un modèle additif de la forme

$$y_t = T_t + S_t + x_t$$

pour $t \in \{1, \dots, T\}$ avec $\{T_t\}$ la **tendance**, $\{S_t\}$ la **composante saisonnière** et $\{x_t\}$ le **résidu** sans tendance ni saisonnalité.

On suppose généralement que $\{T_t\}$ et $\{S_t\}$ sont des fonctions déterministes (c'est à dire sans composante aléatoire) alors que $\{x_t\}$ est la réalisation d'un **processus aléatoire** $\{X_t\}$. Comme $\{x_t\}$ est une série temporelle sans tendance ni saisonnalité, on suppose que le processus $\{X_t\}$ est **stationnaire**. Avec ces hypothèses, $\{y_t\}$ est également la réalisation d'un processus $\{Y_t\}$ qui n'est généralement pas stationnaire.

Etapes dans la modélisation d'une série temporelle

1. **Modélisation des composantes non-stationnaires.** Afin de modéliser une série temporelle, on commence généralement par la modélisation des composantes non-stationnaires. Cette étape a pour objectif en particulier de "stationnariser" la série temporelle initiale $\{y_t\}$. Par exemple, si on suppose que le modèle (9) est valide, alors on cherchera à estimer les fonctions $\{T_t\}$ et $\{S_t\}$. En retirant les composantes non-stationnaires estimées de la série temporelle initiale, on obtient une estimation de $\{x_t\}$.

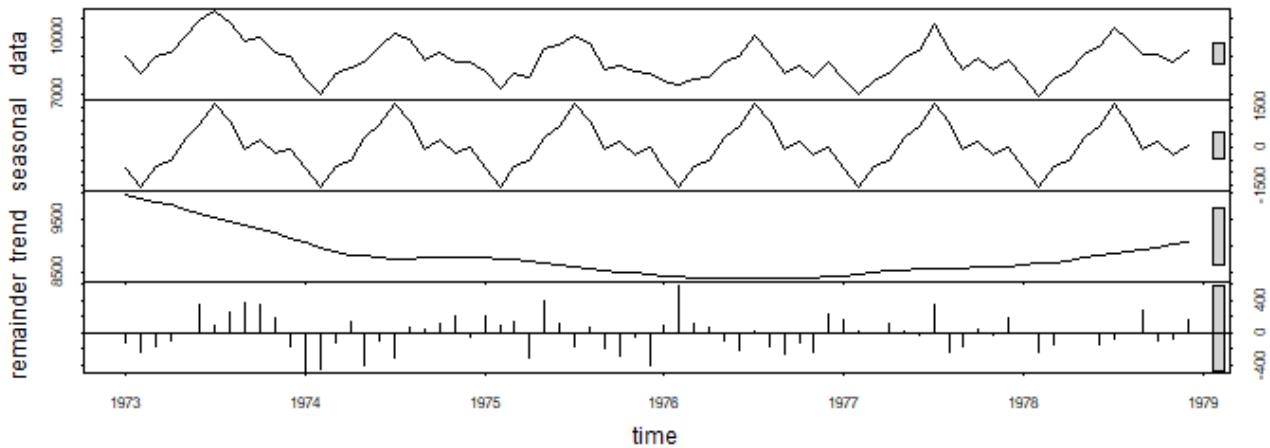
2. **Modélisation des composantes stationnaires.** On modélise ensuite la composante stationnaire $\{X_t\}$. Le modèle le plus répandu dans la littérature sur les séries temporelles est le modèle ARMA.

Finalement, après avoir validé le modèle proposé, on peut l'utiliser pour répondre aux objectifs initiaux (typiquement faire des prévisions ou un générateur).

Illustration avec R.

La fonction `stl` de R permet de décomposer une série temporelle sous la forme du modèle additif (9) (la méthode utilisée par la fonction `stl` sera discutée plus précisément dans la suite du cours).

```
plot(stl(USAccDeaths,s.window=12))
```



#12 est le nombre de mois par an (période de la composante saisonnière)

Pour la série temporelle `USAccDeaths`, la tendance ('trend') identifiée par `stl` montre une décroissance puis une croissance du nombre d'accidents, avec un minimum en 1976. La composante saisonnière ('seasonal') est une fonction périodique de période 1 an (12 mois) et montre un nombre d'accidents plus important en été que en hiver. Le résidu ('remainder') est obtenu en retirant la tendance et la composante saisonnière de la série temporelle initiale.

1.4 Plan du cours

Les processus stationnaires, et notamment les processus ARMA, sont au cœur de la démarche de la modélisation des séries temporelles. Le chapitre 2 rappelle les principales notions en lien avec les processus stationnaires/ARMA qui ont été vues dans la première partie du cours.

Nous discuterons ensuite dans le chapitre 3 l'inférence statistique dans les modèles ARMA, et notamment les points suivants : estimation des paramètres, validation de modèle, prévisions.

Finalement, le chapitre 4 sera consacré à la modélisation des composantes non-stationnaires. Outre l'estimation des fonctions $\{T_t\}$ et $\{S_t\}$ dans le modèle (9), nous discuterons les approches basées sur la différentiation.

2 Processus stationnaires

2.1 Quelques rappels sur la première partie du cours

2.1.1 Processus stationnaires

Définition 1 Un *processus* (à temps discret) est une suite de variables aléatoires $(X_t)_{t \in I}$ avec $I = \mathbb{N}$ (ou $I = \mathbb{Z}$).

Intuitivement, un processus stationnaire est un processus dont “la loi n’évolue pas au cours du temps”. Il existe différentes définitions formelles possibles. Dans le cadre de ce cours, nous nous intéresserons uniquement à la stationnarité du second ordre définie ci-dessous.

Définition 2 Un processus $X = (X_t)_{t \in I}$ est **stationnaire** (au second ordre) si

- $\forall t \in \mathbb{N}, E(X_t) = \mu$ (l’espérance n’évolue pas au cours du temps),
- $\forall (t, h) \in \mathbb{N}^2, \text{cov}(X_t, X_{t+h}) = \gamma(h)$ (les covariances n’évoluent pas au cours du temps).

Remarque 1 • Pour que la définition ci-dessus ait un sens, il faut que les moments d’ordre 2 soient finis. On fera cette hypothèse dans la suite du cours.

- Si le processus est stationnaire alors $\text{var}(X_t) = \text{cov}(X_t, X_t) = \gamma(0)$ ne dépend pas du temps.
- Si X est un processus stationnaire, alors le processus Z défini pour $t \in I$ par $Z_t = X_t - \mu$ est un processus stationnaire centré (c’est à dire tel que $E[Z_t] = 0$). **Dans la suite du cours, sauf mention contraire, on supposera que tous les processus sont centrés** (c’est à dire que $\mu = 0$ dans la définition ci-dessus).

Définition 3 Soit X un processus stationnaire. La fonction γ qui vérifie

$$\text{cov}(X_t, X_{t+h}) = \gamma(h)$$

pour $(t, h) \in \mathbb{N}^2$, est appelée **fonction d’autocovariance** de X . La fonction ρ définie pour $h \in \mathbb{N}$ par

$$\rho(h) = \frac{\gamma(h)}{\gamma(0)}$$

est appelée **fonction d’autocorrélation**.

Remarque 2 Avec les définitions ci-dessus, les fonctions d’autocorrélation et d’autocovariance sont définies sur $\mathbb{N}$. On peut étendre la définition de γ et ρ sur $\mathbb{Z}$ en posant pour $\gamma(h) = \gamma(-h)$ et $\rho(h) = \rho(-h)$ pour $h \in \mathbb{N}$. On a alors

$$\begin{aligned} \text{Cov}(X_t, X_{t+h}) &= \gamma(h) \\ \text{Cor}(X_t, X_{t+h}) &= \rho(h) \end{aligned}$$

si $t \in I$ et $t+h \in I$ (vrai aussi pour $h \leq 0$).

Remarque 3 Si γ est la fonction de covariance d’un processus stationnaire, alors est de **type positif** c’est à dire vérifie pour tout $(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^n$

$$\text{var}\left(\sum_{i=1}^n \lambda_i X_i\right) = \sum_{i,j=1}^n \lambda_i \lambda_j \gamma(i-j) \geq 0$$

2.1.2 Processus gaussiens

Définition 4 Un processus $(X_t)_{t \in I}$ est **gaussien** si pour toute suite $(t_i)_{i \in \{1, \dots, n\}} \in I^n$ d’instant, le vecteur aléatoire $(X_{t_1}, X_{t_2}, \dots, X_{t_n})$ est un vecteur gaussien.

Remarque 4 1. Pour montrer qu’un processus est gaussien $(X_t)_{t \in \mathbb{N}^*}$, il suffit de vérifier que le vecteur aléatoire $(X_1, X_2, \dots, X_n)$ est un vecteur gaussien pour tout $n \in \mathbb{N}^*$.

2. La loi d’un processus gaussien centré et stationnaire est caractérisée par sa fonction de covariance γ . On a

$$(X_{t_1}, X_{t_2}, \dots, X_{t_n}) \sim \mathcal{N}(0, \Sigma)$$

avec $\Sigma_{i,j} = \gamma(t_i - t_j)$.

2.1.3 Processus ARMA

Définition 5 On appelle **bruit blanc** un processus stationnaire centré $(\epsilon_t)_{t \in I}$ tel que la fonction d'autocovariance γ vérifie $\gamma(0) = \text{var}(\epsilon_t) < +\infty$ et $\gamma(h) = 0$ si $h \neq 0$. En particulier, si $(\epsilon_t)_{t \in I}$ est une suite de v.a. i.i.d avec $E[\epsilon_t] = 0$ et $\text{var}(\epsilon_t) < +\infty$, alors $(\epsilon_t)_{t \in I}$ est un bruit blanc (on parle parfois de **bruit blanc fort**). Si les v.a. sont gaussiennes, on parle de **bruit blanc gaussien**.

Définition 6 Un processus **stationnaire** X suit un modèle $\text{ARMA}(p, q)$ si il vérifie l'équation suivante

$$X_t - \alpha_1 X_{t-1} - \dots - \alpha_p X_{t-p} = \epsilon_t + \beta_1 \epsilon_{t-1} + \dots + \beta_q \epsilon_{t-q} \quad (1)$$

pour $t \in \mathbb{Z}$ avec ϵ un bruit blanc tel que $\text{var}(\epsilon_t) = \sigma^2$ et $(\alpha_1, \dots, \alpha_p, \beta_1, \dots, \beta_q, \sigma)$ les paramètres du modèle. Lorsque $q = 0$, on obtient le modèle $\text{AR}(p)$ et lorsque $p = 0$, on obtient le modèle $\text{MA}(q)$.

Proposition 1 Si les racines complexes du polynôme $A(z) = 1 - \alpha_1 z - \dots - \alpha_p z^p$ sont de module strictement supérieures à 1, alors il existe un processus stationnaire qui vérifie (1). De plus, on peut trouver une solution telle que $\text{cov}(X_s, \epsilon_t) = 0$ si $s < t$ (processus causal).

Remarque 5 On peut montrer que sous les hypothèses de la proposition précédente, le processus ARMA admet la représentation $\text{MA}(\infty)$

$$X_t = \epsilon_t + b_1 \epsilon_{t-1} + b_2 \epsilon_{t-2} + \dots$$

avec $(b_i)_{i \in \mathbb{N}^*}$ les coefficients du développement de $\frac{B(z)}{A(z)}$ en séries entières, c'est à dire tels que

$$\frac{B(z)}{A(z)} = 1 + b_1 z + b_2 z^2 + \dots$$

Cette représentation peut être utile par exemple pour simuler un processus ARMA à partir d'une simulation d'un bruit blanc $\{\epsilon_t\}$ en utilisant une approximation par une série tronquée

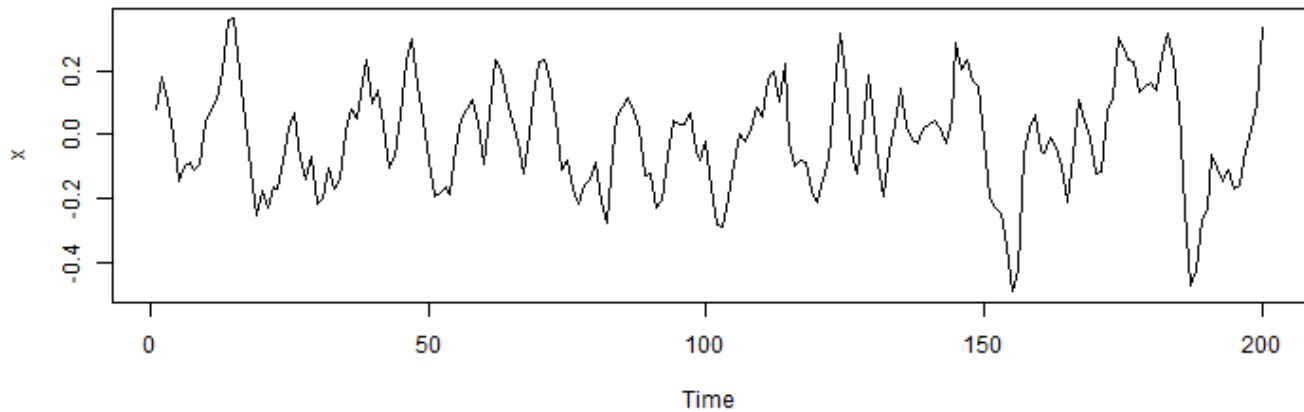
$$X_t \approx \epsilon_t + b_1 \epsilon_{t-1} + b_2 \epsilon_{t-2} + \dots + b_N \epsilon_{t-N}$$

avec N "grand". On peut aussi envisager de simuler un tel processus en utilisant la relation (1) à partir de données initiales $x_1, \dots, x_p$ choisies arbitrairement, puis en "oubliant" les premières valeurs obtenues de telle manière à "atteindre" le régime d'équilibre stationnaire. C'est cette dernière approche qui est utilisée par la fonction `arima.sim` de *R*.

Illustration avec R.

L'exemple ci-dessous montre un exemple de simulation d'un modèle $\text{ARMA}(2,1)$.

```
alpha=c(.5,.2) #paramètres AR
beta=.5 #paramètres MA
sig=.1 #écart-type du bruit
T=200 #longueur de la série temporelle
x=arima.sim(model=list(ar=alpha,ma=beta),sd=sig,n=T) #simulation modèle ARMA
plot(x) #représentation graphique
```



2.1.4 Estimation des fonctions d'autocovariance et d'autocorrélation

Dans cette section X désigne un processus stationnaire et $(x_1, \dots, x_T)$ une réalisation de X . On notera $\mu = E[X_i]$ (dans ce paragraphe, on ne suppose pas que $\mu = 0$), γ la fonction d'autocovariance, ρ la fonction d'autocorrélation et τ la fonction d'autocorrélation partielle de X .

Définition 7 On appelle

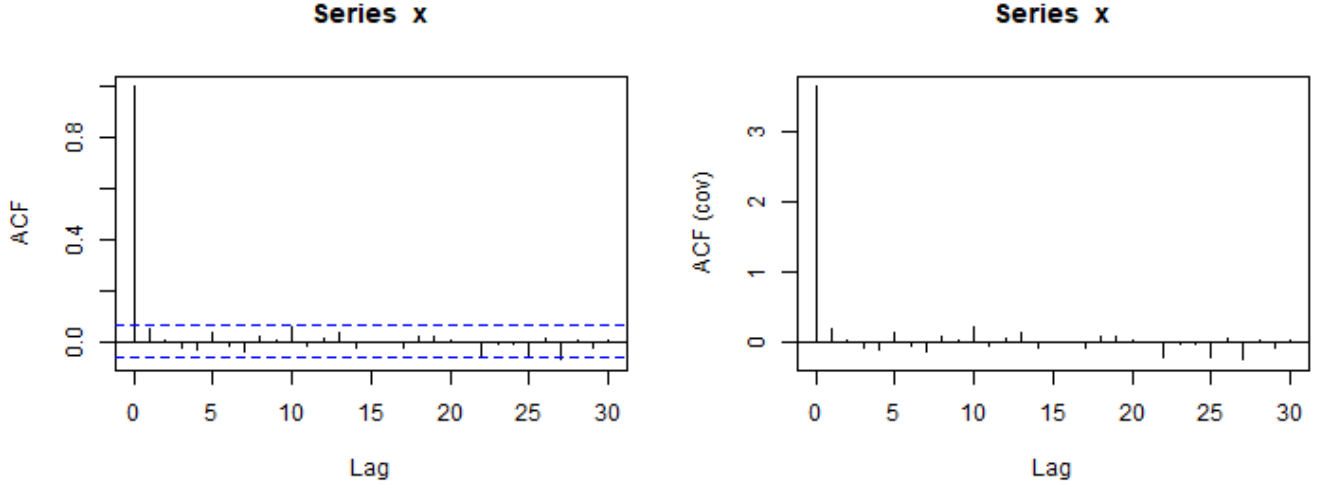
- **moyenne empirique** $\bar{X}_T = \frac{1}{T} \sum_{t=1}^T X_t$,
- **fonction d'autocovariance empirique** $\hat{\gamma}(h) = \frac{1}{T} \sum_{t=1}^{T-h} (X_t - \bar{X}_T)(X_{t+h} - \bar{X}_T)$ (définie pour $h \leq T$),
- **fonction d'autocorrélation empirique** $\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)}$,

On peut montrer que, sous des conditions assez générales (qui sont vérifiées en particulier par les processus ARMA), ces estimateurs sont consistants (c'est à dire qu'ils convergent en probabilité vers les quantités théoriques associés lorsque la longueur de la série observée T tend vers $+\infty$) et asymptotiquement gaussiens.

Illustration avec R.

Les fonctions `acf` et `pacf` de R permettent de tracer les fonctions d'autocovariance et d'autocorrélation d'une série temporelle. L'exemple ci-dessous illustre l'utilisation de ces fonctions sur une simulation d'un bruit blanc gaussien. La signification des lignes horizontales bleues qui apparaissent sur le premier graphique est discutée dans le paragraphe suivant.

```
T=1000
x=2*rnorm(T) #simulation d'un bruit blanc gaussien
par(mfrow=c(1,2))
acf(x) #tracé de la fonction d'autocorrélation
acf(x,type='covariance') #tracé de la fonction d'autocovariance
```



2.1.5 Tests de bruit blanc

Les bruits blancs apparaissent comme élément de base dans la construction de nombreux modèles (processus ARMA, mais aussi marche aléatoire, mouvement brownien ou processus d'Ornstein-Uhlenbeck). Il est donc important d'avoir des outils pour vérifier si une série temporelle peut être considérée comme une réalisation d'un bruit blanc. Intuitivement, une réalisation d'un bruit blanc est une série d'observations dans laquelle il n'y a pas de corrélation temporelle. On rappelle que si X est un bruit blanc alors $\rho(h) = 0$ pour $h \geq 1$. On s'attend alors à avoir $\hat{\rho}(h) \approx 0$ pour $h \geq 1$ si la longueur T de la série temporelle est suffisamment grande. La proposition suivante donne un sens plus précis à cette convergence. Nous verrons ensuite comment utiliser cette proposition pour vérifier si une série temporelle est un bruit blanc.

Proposition 2 Si X est un bruit blanc avec $E[X_t^4] < +\infty$, alors pour $h \geq 1$,

$$\sqrt{T} \begin{pmatrix} \hat{\rho}(1) \\ \vdots \\ \hat{\rho}(h) \end{pmatrix} \xrightarrow{\mathcal{L}} \mathcal{N}(0, I_h) \text{ lorsque } T \rightarrow +\infty$$

avec I_h la matrice identité d'ordre h .

De manière informelle, la proposition précédente indique que si T est grand alors les variables aléatoires $\sqrt{T}\hat{\rho}(1), \dots, \sqrt{T}\hat{\rho}(h)$ sont "approximativement i.i.d. de loi $\mathcal{N}(0, 1)$ ".

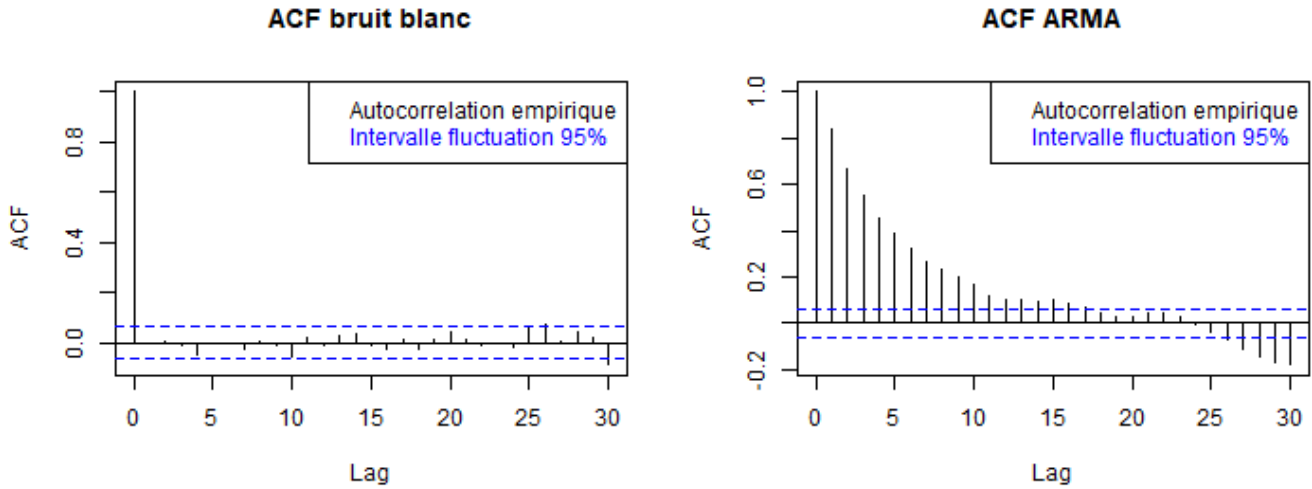
Outil graphique pour vérifier l'hypothèse de bruit blanc. En particulier, on déduit de la proposition précédente que si T est "grand" et X est un bruit blanc, alors pour $h \geq 1$ $\sqrt{T}\hat{\rho}(h)$ suit "approximativement" une loi $\mathcal{N}(0, 1)$. On peut utiliser ce résultat pour faire des **intervalles de fluctuation** asymptotiques (asymptotiques car valable pour T "grand"). Plus précisément, si X est un bruit blanc, alors pour $h \geq 1$

$$P\left(-\frac{1.96}{\sqrt{T}} \leq \hat{\rho}(h) \leq \frac{1.96}{\sqrt{T}}\right) \approx 0.95$$

et on s'attend donc à ce que le coefficient d'ordre h de la fonction d'autocorrélation empirique $\hat{\rho}(h)$ soit dans l'intervalle de fluctuation $[-\frac{1.96}{\sqrt{T}}, \frac{1.96}{\sqrt{T}}]$ avec une probabilité de 95%. Cet intervalle est tracé par défaut par la fonction `acf` de R (lignes horizontales bleues). Il donne une information sur l'erreur d'estimation (ou "fluctuation d'échantillonnage") de la fonction d'autocovariance et permet de vérifier visuellement si une série temporelle est un bruit blanc : si c'est le cas, la fonction d'autocorrélation empirique tracée en noir doit être dans l'intervalle de fluctuation tracé en bleu avec une probabilité de 95%. En pratique, si certains coefficients de la fonction d'autocorrélation empirique semblent "grands" par rapport à l'intervalle de fluctuation, alors c'est une indication que la série temporelle n'est pas un bruit blanc. Ceci est illustré sur les exemples ci-dessous avec R.

Illustration avec R.

```
T=1000 #longueur de la série temporelle
par(mfrow=c(1,2))
x=rnorm(T) #simulation d'un bruit blanc gaussien
acf(x,main='ACF bruit blanc') #tracé de la fonction d'autocorrélation empirique
legend('topright',c('Autocorrélation empirique','Intervalle fluctuation 95%')
      ,text.col=c('black','blue'))
x=arima.sim(model=list(ar=c(.5,.2),ma=.5),sd=.1,n=T) #simulation modèle ARMA
acf(x,main='ACF ARMA') #tracé de la fonction d'autocorrélation empirique
legend('topright',c('Autocorrélation empirique','Intervalle fluctuation 95%')
      ,text.col=c('black','blue'))
```



Test d'hypothèse. Le **test de Portmanteau** (portmanteau signifie “fourre-tout” en anglais) regroupe les différents coefficients de la fonction d'autocorrélation empirique en une seule statistique de test pour construire un test de bruit blanc. Le fondement probabiliste de ce test est donné par la proposition ci-dessous, qui se déduit de la proposition précédente.

Proposition 3 *Posons, pour $h \geq 1$,*

$$Q_{BP} = T \sum_{i=1}^h \hat{\rho}(i)^2$$

$$Q_{LB} = T(T+2) \sum_{i=1}^h \frac{\hat{\rho}(i)^2}{T-i}$$

Si X est un bruit blanc avec $E[X_t^4] < +\infty$, alors Q_{LB} et Q_{BP} convergent en loi vers une loi du χ^2 à h degrés de liberté lorsque T tend vers $+\infty$.

La quantité Q_{BP} est la statistique du test de “Box-Pierce” et Q_{LB} est la statistique du test de “Ljung-Box” qui sont deux variantes du test de “Portmanteau”. Ces deux tests peuvent être réalisés avec la fonction `Box.test` de R. Plus précisément, les hypothèses testées sont

$$H_0 : X \text{ est un bruit blanc } \textbf{contre} \ H_1 : X \text{ n'est pas un bruit blanc.}$$

On rappelle que H_0 signifie qu'il n'y a pas de corrélation temporelle significative dans les observations. Si H_0 est vraie et T suffisamment grand, alors d'après la proposition précédente $Q_{LB} \approx \chi_p^2$ et $Q_{BP} \approx \chi_p^2$. Pour un risque de première espèce α , on accepte alors H_0 si et seulement si

- $Q_{BP} \leq \chi_{p,1-\alpha}^2$ pour le test de Box-Pierce
- $Q_{LB} \leq \chi_{p,1-\alpha}^2$ pour le test de Ljung-Box

avec $\chi_{p,1-\alpha}^2$ le quantile d'ordre $1 - \alpha$ de la loi χ_p^2 .

La statistique du test de Box-Pierce est plus facile à interpréter (on regarde la norme 2 des h premiers coefficients de la fonction d'autocorrélation qui doit être proche de 0 pour un bruit blanc) mais on peut montrer que Q_{LB} est généralement plus proche d'une loi du χ^2 que Q_{BP} (et donc le test de Ljung-Box est "plus précis"). En pratique, si T est grand, les deux statistiques sont très proches (cf exemple ci-dessous).

La fonction `Box.test` renvoie la p-value du test définie par $P(X > q)$ avec $X \sim \chi_p^2$ et q la valeur prise par la statistique de test. On rappelle qu'on accepte H_0 avec un risque de première espèce α si et seulement si la p-value est **supérieure** à α .

Le choix de h est subjectif. Par défaut, la fonction `Box.test` de R prend la valeur $h = 1$. Le test est alors basé seulement sur le premier coefficient de la fonction d'autocorrélation. L'argument 'lag' permet d'augmenter la valeur de h .

Illustration avec R.

```
T=1000
x=rnorm(T)  #simulation d'un bruit blanc gaussien
Box.test(x)  #test de Box-Pierce avec h=1 par défaut (test uniquement sur rho(1))

##
## Box-Pierce test
##
## data:  x
## X-squared = 0.61166, df = 1, p-value = 0.4342

#la p-value est >5% dans 95% des simulations
Box.test(x,lag=5)  #test de Box-Pierce avec h=5

##
## Box-Pierce test
##
## data:  x
## X-squared = 1.6927, df = 5, p-value = 0.8898

Box.test(x,type="Ljung-Box")  #test de Ljung-Box, résultats proche du test de Box-Pierce

##
## Box-Ljung test
##
## data:  x
## X-squared = 0.6135, df = 1, p-value = 0.4335

x=arima.sim(model=list(ar=c(.5,.2),ma=.5),sd=.1,n=T)  #simulation d'un ARMA(2,1)
Box.test(x)  #on refuse H_0, x n'est pas un bruit blanc

##
## Box-Pierce test
##
## data:  x
## X-squared = 672.61, df = 1, p-value < 2.2e-16
```

Remarque 6 D'autres tests ont été vus dans la première partie du cours : test des signes, test des points de retournement, test des rangs.

2.2 Exercices

Exercice 1 Soit $(\epsilon_t)_{t \in \mathbb{N}} \sim_{iid} \mathcal{N}(0, \sigma^2)$ un bruit blanc gaussien fort. On considère les processus définis par les équations ci-dessous :

1. $A_t = \epsilon_t + \beta_1 \epsilon_{t-1}$ avec $\beta_1 \in \mathbb{R}$.
2. $B_0 \sim \mathcal{N}(0, \tau^2)$ indépendant de $\{\epsilon\}_{t \in \mathbb{N}}$ et $B_t = B_{t-1} + \epsilon_t$ pour $t \geq 1$.
3. $C_t = \epsilon_t \epsilon_{t-1}$.
4. $D_t = t + \epsilon_t$.
5. $E_t = t \epsilon_t$.

Pour chacun de ces processus, on répondra aux questions ci-dessous :

1. Est-ce que le processus est stationnaire? Si c'est le cas, on calculera les fonctions d'autocovariance et d'autocorrélation.
2. Est-ce que le processus est gaussien?
3. Simuler une trajectoire de longueur $T = 500$ en utilisant R. Représenter graphiquement la séquence simulée.
4. Tracer la fonction d'autocorrélation empirique de la séquence simulée en utilisant la fonction `acf`. Lorsque le processus est stationnaire, représenter également la fonction d'autocorrélation théorique sur le même graphique. Discuter les résultats obtenus.

Exercice 2 Soit $\{X_t\}$ un processus MA(3) qui vérifie

$$X_t = \epsilon_t + \beta_1 \epsilon_{t-1} + \beta_2 \epsilon_{t-2} + \beta_3 \epsilon_{t-3}$$

avec ϵ_t un bruit blanc, $\beta_1 = 1$, $\beta_2 = .5$ et $\beta_3 = -.5$.

1. Calculer les fonctions d'autocovariance et d'autocorrélation du processus $\{X_t\}$.
2. Vérifier votre résultat en utilisant la fonction `ARMAacf` de R.
3. Simuler une trajectoire de longueur $T = 1000$ du processus $\{X_t\}$ en utilisant la fonction `arima.sim` et tracer sur une même figure la fonction d'autocorrélation empirique et la fonction d'autocorrélation théorique.

Exercice 3 Soit $\{X_t\}$ un processus qui vérifie $\{X_0\} \sim \mathcal{N}(0, \tau^2)$ et pour $t \geq 1$

$$X_t = \alpha_1 X_{t-1} + \epsilon_t$$

avec $\{\epsilon_t\}$ un bruit blanc gaussien fort tel que ϵ_t est indépendant de $\{X_s\}_{1 \leq s < t}$.

1. Calculer $\text{var}(X_t)$ en fonction de $\text{var}(X_{t-1})$ et en déduire que si le processus $\{X_t\}$ est stationnaire alors $|\alpha_1| < 1$. On supposera que cette condition est vérifiée dans la suite de l'exercice.
2. Déduire de la question précédente que si le processus $\{X_t\}_{t \in \mathbb{N}}$ est stationnaire alors $\text{var}(X_t) = \frac{\sigma^2}{1 - \alpha_1^2}$ et donc en particulier $\tau^2 = \frac{\sigma^2}{1 - \alpha_1^2}$.

3. On suppose dans la suite de l'exercice que $|\alpha_1| < 1$ et $\tau^2 = \frac{\sigma^2}{1-\alpha_1^2}$. Montrer que le processus $\{X_t\}_{t \in \mathbb{N}}$ est stationnaire et calculer ses fonctions d'autocovariance et d'autocorrélation. Montrer que le processus $\{X_t\}_{t \in \mathbb{N}}$ est gaussien.
4. Justifier pourquoi le processus construit dans la question précédente est un processus AR(1) et proposer un algorithme permettant de simuler une trajectoire de ce processus.
5. Simuler une trajectoire de longueur $T = 1000$ du processus $\{X_t\}$ en utilisant l'algorithme de la question précédente et tracer sur une même figure la fonction d'autocorrélation empirique et la fonction d'autocorrélation théorique. On pourra prendre $\alpha_1 = 0.9$ puis $\alpha_1 = -0.9$ et $\text{var}(\epsilon_t) = 0.1$.

Exercice 4 Soit $(X_t)_{t \in \mathbb{N}}$ un processus stationnaire de moyenne μ , γ sa fonction d'autocovariance et $\bar{X}_T = \frac{1}{T} \sum_{t=1}^T X_t$ la moyenne empirique.

1. Soit $(\lambda_1, \dots, \lambda_T) \in \mathbb{R}^T$. Exprimer $\text{var}(\sum_{t=1}^T \lambda_t X_t)$ et déduire que $\sum_{s,t=1}^n \lambda_s \lambda_t \gamma(s-t) \geq 0$
2. Montrer que $\bar{X}_T$ est un estimateur sans biais de μ .
3. Exprimer $\text{var}(\bar{X}_T)$. Que devient cette formule si X est un bruit blanc? Commenter.
4. Montrer que si $\sum_{h \geq 0} |\gamma(h)| < +\infty$ alors $\bar{X}_T$ converge en moyenne d'ordre 2 vers μ lorsque T tend vers $+\infty$.

Exercice 5 1. Vérifier, en utilisant des simulations d'un bruit blanc, que

- (a) pour $h \geq 1$, $\sqrt{T}\hat{\rho}(h)$ suit approximativement une loi $\mathcal{N}(0, 1)$ (avec T la longueur de la série temporelle et $\hat{\rho}$ la fonction d'autocorrélation empirique);
 - (b) l'intervalle $[-\frac{1.96}{\sqrt{T}}, \frac{1.96}{\sqrt{T}}]$ contient bien 95% des coefficients d'autocorrélation empiriques $\hat{\rho}(h)$ pour $h \geq 1$;
 - (c) on accepte bien l'hypothèse H_0 du test de Portmanteau avec une probabilité égale à 95% pour un risque de première espèce égal à 5%.
2. Vérifier, en utilisant des simulations d'un processus MA(1), que
- (a) le coefficient $\hat{\rho}(1)$ n'est plus dans l'intervalle $[-\frac{1.96}{\sqrt{T}}, \frac{1.96}{\sqrt{T}}]$ avec une probabilité de 95%;
 - (b) l'hypothèse H_0 du test de Portmanteau est "généralement" rejetée.
3. Dans le modèle de Black-Scholes, la série temporelle des log-rendements $X_t = \ln(\frac{S_t}{S_{t-1}}) = \ln(S_t) - \ln(S_{t-1})$ de l'actif sous-jacent est supposée être une suite de variables aléatoires gaussiennes i.i.d. Vérifier, à l'aide d'outils graphiques et de tests statistiques, si cette hypothèse est réaliste pour l'actif financier de votre choix (vous pourrez télécharger les données de votre choix sur R ou utiliser les données existantes dans R comme `EuStockMarkets`).

3 Inférence statistique dans les modèles ARMA

Les modèles ARMA sont les modèles les plus classiques pour décrire des séries temporelles stationnaires. Ces modèles sont constitués de deux parties, à savoir

- une partie AutoRegressive (notée AR);
- une partie moyenne mobile (notée MA pour Moving Average);

Avant d'étudier les modèles ARMA généraux, puis leurs extensions dans le cas non-stationnaire, nous allons nous intéresser aux modèles AR.

3.1 Cas particulier des modèles AR

3.1.1 Quelques propriétés des processus AR(p)

Commençons par rappeler la définition d'un modèle AR(p).

Définition 8 *Un processus stationnaire $\{X\}_{t \geq 1}$ suit un modèle AR(p) si il vérifie la relation de récurrence*

$$X_t = \alpha_1 X_{t-1} + \dots + \alpha_p X_{t-p} + \epsilon_t \quad (2)$$

pour $t \geq p+1$ avec ϵ un bruit blanc avec $\text{var}(\epsilon_t) = \sigma^2$. $(\alpha_1, \dots, \alpha_p) \in \mathbb{R}^p$ et $\sigma > 0$ sont les paramètres du modèle. $p \in \mathbb{N}$ est appelé ordre du processus autorégressif.

Dans le modèle AR, on suppose que la valeur X_t du processus à l'instant t est obtenue en faisant une combinaison linéaire des p valeurs précédentes et d'un résidu aléatoire. Ces hypothèses sont similaires à celles faites dans le modèle de régression linéaire, les valeurs passées du processus jouant le rôle de variable explicatives (d'où le nom "autorégressif").

Rappelons également qu'il n'est pas toujours possible de construire un processus stationnaire qui vérifie (2). Dans certains cas, il est possible de construire une solution mais cette solution n'est pas causale (ϵ_t dépend de $(X_1, \dots, X_{t-1})$). Dans les chapitres suivants, on considérera uniquement les modèles ARMA pour lesquels il existe une solution causale. La proposition ci-dessous, qui est un cas particulier de la Proposition 1 donne une condition nécessaire et suffisante pour que cela soit possible.

Proposition 4 - *On peut construire un processus AR(p) stationnaire causal qui vérifie (2) si et seulement si les racines (complexes) de son polynôme caractéristique P défini par*

$$P(x) = 1 - (\alpha_1 x + \dots + \alpha_p x^p)$$

sont de module strictement supérieur à 1.

Sous les conditions de la proposition précédente, il est possible de construire une solution causale telle que $\text{cov}(\epsilon_t, X_{t-s}) = 0$ pour $s \geq 1$. Dans la suite du paragraphe, on fera l'hypothèse plus forte que ϵ_t est **indépendant** du passé du processus $(X_s)_{1 \leq s < t}$. On vérifie alors que pour $t \geq p+1$

$$P(X_t | X_{t-1} = x_{t-1}, \dots, X_1 = x_1) = P(X_t | X_{t-1} = x_{t-1}, \dots, X_{t-p} = x_{t-p})$$

et donc le processus X est **Markovien d'ordre p** . De plus, on a

$$\begin{aligned} E(X_t | X_{t-1}, \dots, X_1) &= E(\alpha_1 X_{t-1} + \dots + \alpha_p X_{t-p} + \epsilon_t | X_{t-1}, \dots, X_{t-p}) \\ &= \alpha_1 X_{t-1} + \dots + \alpha_p X_{t-p} \end{aligned}$$

$$\text{var}(X_t | X_{t-1}, \dots, X_1) = \sigma^2.$$

La meilleure prédiction de X_t (au sens des moindres carrés) connaissant le passé du processus est donc une combinaison linéaire des p valeurs précédentes et la variance conditionnelle (ou "volatilité") est constante au cours du temps (modèle 'homoscédastique').

On supposera généralement dans la suite que le bruit $\{\epsilon_t\}$ qui apparaît dans la définition des modèles ARMA est un bruit blanc fort gaussien (c'est à dire une suite de variables aléatoires iid de loi $\mathcal{N}(0, \sigma^2)$). On parle alors de "modèle ARMA gaussien". Ceci permet de simplifier les développements théoriques et les aspects numériques (comme pour le modèle de régression linéaire par exemple). Dans ce cas particulier, on retiendra que $\{X_t\}$ un processus **gaussien** et **Markovien** tel que la loi conditionnelle de X_t sachant $X_{t-1} = x_{t-1}, \dots, X_1 = x_1$ est donnée par

$$P(X_t | X_{t-1} = x_{t-1}, \dots, X_1 = x_1) \sim \mathcal{N}(\alpha_1 x_{t-1} + \dots + \alpha_p x_{t-p}, \sigma^2)$$

3.1.2 Fonction d'autocovariance et estimation par la méthode des moments

Une première approche pour estimer les paramètres d'un modèle AR(p) est basée sur l'expression de la fonction d'autocovariance en fonction des paramètres du modèle et la méthode des moments.

La proposition ci-dessous permet de calculer récursivement la fonction d'autocovariance d'un processus AR(p).

Proposition 5 Soit X un processus AR(p) causal de paramètres $\alpha_1, \dots, \alpha_p, \sigma^2$ et γ sa fonction d'autocovariance. On a

$$\gamma(0) = \sum_{k=1}^p \alpha_k \gamma(k) + \sigma^2 \quad (\text{YW}(0))$$

et γ vérifie les *équations de Yule-Walker*

$$\gamma(j) = \sum_{i=1}^p \alpha_i \gamma(j-i) \quad (\text{YW}(j))$$

pour $j > 0$.

Preuve 1 Par définition des processus AR, on a

$$X_t = \sum_{i=1}^p \alpha_i X_{t-i} + \epsilon_t$$

On en déduit que pour $j \in \mathbb{N}$

$$\text{cov}(X_t, X_{t-j}) = \text{cov}\left(\sum_{i=1}^p \alpha_i X_{t-i} + \epsilon_t, X_{t-j}\right)$$

et donc que

$$\gamma(j) = \sum_{i=1}^p \alpha_i \gamma(j-i) + \text{cov}(\epsilon_t, X_{t-j})$$

Supposons que la solution est causale de telle manière que $j > 0$ on a $\text{cov}(\epsilon_t, X_{t-j}) = 0$. L'expression précédente se simplifie en

$$\gamma(j) = \sum_{i=1}^p \alpha_i \gamma(j-i).$$

Pour $j = 0$, remarquons que

$$\text{cov}(\epsilon_t, X_t) = \text{cov}\left(\epsilon_t, \sum_{i=1}^p \alpha_i X_{t-i} + \epsilon_t\right)$$

et donc $\text{cov}(\epsilon_t, X_t) = \sigma^2$. On en déduit que

$$\gamma(0) = \sum_{i=1}^p \alpha_i \gamma(j-i) + \sigma^2.$$

Conséquences

1. **Calcul de la fonction d'autocovariance si on connaît les paramètres du modèle.** En utilisant les équations (YW(j)) pour $j \in \{0, \dots, p\}$ on obtient que $(\gamma(0), \dots, \gamma(p))$ est solution d'un système de $p+1$ équations linéaires dont les coefficients dépendent des paramètres du modèle $(\alpha_1, \dots, \alpha_p, \sigma^2)$. En résolvant ce système, on peut calculer $(\gamma(0), \dots, \gamma(p))$ puis les coefficients $\gamma(k)$ pour $k > p$ par récurrence en utilisant (YW(k)) (cf Exercice 7 pour le cas particulier des modèles AR(2)).

2. **Estimation des paramètres du modèle par la méthode des moments.** Réciproquement, lorsque les paramètres sont inconnus, on peut utiliser les équations (YW(j)) pour estimer les paramètres du modèle. Pour cela, on remplace la fonction d'autocovariance théorique γ par son estimateur empirique $\hat{\gamma}$ dans les équations (YW(j)) pour $j \in \{0, \dots, p\}$. Cela conduit à un système d'équations linéaires pour les paramètres inconnus $(\alpha_1, \dots, \alpha_p, \sigma^2)$. En résolvant ce système, on obtient les estimateurs de Yule-Walker de $\alpha_1, \dots, \alpha_p, \sigma^2$. Cet estimateur est proposé par la fonction `ar` de R (option `method="yule-walker"`) (cf Exercice 7 pour le cas particulier des modèles AR(2)).

3. On déduit facilement de la proposition précédente que la fonction d'autocorrélation ρ d'un processus AR(p) vérifie, pour $j \geq 1$,

$$\rho(j) = \sum_{i=1}^p \alpha_i \rho(j-i) \quad (\text{YW}(j))$$

Ceci permet de calculer la fonction d'autocorrélation ρ lorsqu'on connaît les paramètres $(\alpha_1, \dots, \alpha_p)$, et vice versa.

3.1.3 Fonction de vraisemblance

La méthode du maximum de vraisemblance est utilisée par défaut par R pour ajuster les modèles ARMA. Pour pouvoir définir la fonction de vraisemblance, il faut faire une hypothèse paramétrique sur la loi des résidus : on suppose dans ce paragraphe que c'est un bruit blanc fort gaussien (modèles ARMA gaussiens). Il s'agit d'une hypothèse plus forte que dans le paragraphe précédent (la méthode de Yule-Walker suppose uniquement que le résidu est un bruit blanc).

On rappelle que la fonction de vraisemblance est définie par

$$L(\theta) = p(x_1, \dots, x_T; \theta)$$

avec $p(x_1, \dots, x_T; \theta)$ la densité de la loi jointe du vecteur aléatoire $(X_1, \dots, X_T)$ et $\theta = (\alpha_1, \dots, \alpha_p, \sigma) \in \Theta$ le paramètre inconnu. On remarquera que dans le cas général

$$L(\theta) \neq \prod_{t=1}^T p(x_t; \theta)$$

à cause de la dépendance entre les observations successives.

On peut calculer la fonction de vraisemblance en utilisant deux raisonnements différents.

- **En utilisant les propriétés Markoviennes.**

Par définition des lois conditionnelles, on a

$$p(x_1, \dots, x_T; \theta) = p(x_T | x_1, \dots, x_{T-1}; \theta) p(x_1, \dots, x_{T-1}; \theta)$$

avec $p(x_T | x_1, \dots, x_{T-1}; \theta)$ la densité de la loi conditionnelle de X_T sachant que $X_1 = x_1, \dots, X_{T-1} = x_{T-1}$. En itérant le raisonnement, on vérifie que

$$p(x_1, \dots, x_T; \theta) = p(x_1, \dots, x_p; \theta) \prod_{t=p+1}^T p(x_t | x_1, \dots, x_{t-1}; \theta).$$

Notons

$$\begin{aligned} \tilde{L}(\theta) &= p(x_{p+1}, \dots, x_T | x_1, \dots, x_p; \theta) \\ &= \prod_{t=p+1}^T p(x_t | x_1, \dots, x_{t-1}; \theta) \end{aligned}$$

la vraisemblance conditionnée par les p premières observations. Pour un processus AR(p) gaussien, on a vu que pour $t \geq p+1$

$$\begin{aligned} P(X_t | X_{t-1} = x_{t-1}, \dots, X_1 = x_1) &= P(X_t | X_{t-1} = x_{t-1}, \dots, X_{t-p} = x_{t-p}) \\ &\sim \mathcal{N}(\alpha_1 x_{t-1} + \dots + \alpha_p x_{t-p}, \sigma^2) \end{aligned}$$

et donc

$$p(x_t|x_1, \dots, x_{t-1}; \theta) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x_t - \alpha_1 x_{t-1} - \dots - \alpha_p x_{t-p})^2}{2\sigma^2}\right).$$

Ceci permet d'exprimer la vraisemblance conditionnée par les p premières observations $\tilde{L}$.

On peut ensuite en déduire une expression de la vraisemblance L en utilisant que la loi jointe des p premières observations $p(x_1, \dots, x_p; \theta)$ est un vecteur gaussien centré dont on peut exprimer la matrice de covariance en fonction de θ grâce à la proposition 5.

- **En utilisant les processus Gaussiens.** Sous l'hypothèse de bruit blanc fort gaussien, X est un processus gaussien et le vecteur aléatoire $(X_1, \dots, X_T)$ est un vecteur gaussien. De plus, on a $E[X_t] = 0$ et $cov(X_i, X_j) = \gamma(|i - j|)$ avec la fonction d'autocovariance γ qui peut s'exprimer en fonction des paramètres du modèle (cf Proposition 5). On peut ensuite calculer la vraisemblance en utilisant l'expression de la densité d'un vecteur gaussien.

Pour estimer les paramètres en utilisant la méthode du maximum de vraisemblance, il est possible de maximiser soit la fonction L soit la fonction $\tilde{L}$. On peut obtenir une solution analytique pour

$$\tilde{\theta} = \operatorname{argmax}_{\theta \in \Theta} \tilde{L}(\theta)$$

car il s'agit d'un problème des moindres carrés usuels (cf Exercice 6 pour le cas particulier des modèles AR(1)). Par contre, il est nécessaire d'utiliser un algorithme d'optimisation numérique pour calculer

$$\hat{\theta} = \operatorname{argmax}_{\theta \in \Theta} L(\theta).$$

On peut vérifier que les deux fonctions de vraisemblance conduisent à des estimateurs du maximum de vraisemblance qui sont proches dès que T est suffisamment grand

$$\ln(L(\theta)) = \ln(p(X_1, \dots, X_p; \theta)) + \sum_{t=p+1}^T \ln(p(X_t|X_{t-1}, \dots, X_{t-p}; \theta)) \approx \sum_{t=p+1}^T \ln(p(X_t|X_{t-1}, \dots, X_{t-p}; \theta)) = \ln(\tilde{L}(\theta))$$

Ce sera discuté par simulation dans le cadre des modèles AR(1) dans l'exercice 6.

En général, l'ordre du modèle p du modèle AR est inconnu. Afin d'estimer p , on peut utiliser le critère

$$AIC = -2 \ln(L(\hat{\theta})) + 2k$$

avec k le nombre de paramètres inconnus. On cherche ensuite la valeur de p qui minimise ce critère : ceci permet de trouver un compromis entre une bonne adéquation du modèle aux données (mesurée par la vraisemblance) et un faible nombre de paramètres.

3.1.4 Prévision

Dans ce paragraphe, on suppose que $\{X_t\}$ est un processus stationnaire qui suit un modèle $AR(p)$ de paramètres connus $\alpha_1, \dots, \alpha_p$ et σ . On cherche à prévoir (ou prédire) la valeur prise par le processus à l'instant futur $t + h$ à partir de la connaissance des valeurs prises par ce processus jusqu'à l'instant t , c'est à dire de $(x_1, \dots, x_t)$. $h \in \mathbb{N}^*$ est appelé 'horizon de prévision' dans le vocabulaire des séries temporelles. Toute l'information est contenue dans la loi conditionnelle ('forecast distribution') $P(X_{t+h}|X_t = x_t, \dots, X_1 = x_1)$. Cette loi conditionnelle se calcule aisément pour les processus AR(p) gaussiens.

- Pour $h = 1$ et $t \geq p + 1$, on a

$$P(X_{t+1}|X_t = x_t, \dots, X_1 = x_1) \sim \mathcal{N}(\alpha_1 x_t + \dots + \alpha_p x_{t-p+1}, \sigma^2).$$

- Pour $h = 2$, on a

$$X_{t+2} = \alpha_1 X_{t+1} + \dots + \alpha_p X_{t+2-p} + \epsilon_{t+2}.$$

Comme on dispose des observations seulement jusqu'à l'instant t , on va remplacer X_{t+1} dans l'expression précédente, en utilisant à nouveau la définition du modèle $AR(p)$,

$$X_{t+1} = \alpha_1 X_t + \dots + \alpha_p X_{t+1-p} + \epsilon_{t+1}.$$

On obtient finalement

$$X_{t+2} = \alpha_1(\alpha_1 X_t + \dots + \alpha_p X_{t+1-p} + \epsilon_{t+1}) + \alpha_2 X_t + \dots + \alpha_p X_{t+2-p} + \epsilon_{t+2}$$

et donc au final une expression de la forme

$$X_{t+2} = \tilde{\alpha}_1 X_t + \dots + \tilde{\alpha}_p X_{t+1-p} + \tilde{\epsilon}_{t+2}$$

avec $\tilde{\epsilon}_{t+2} = \alpha_1 \epsilon_{t+1} + \epsilon_{t+2} \sim \mathcal{N}(0, \tilde{\sigma}^2)$ indépendant de $(X_s)_{s \leq t}$ avec $(\tilde{\alpha}_1, \dots, \tilde{\alpha}_p)$ qui s'expriment en fonction de $(\alpha_1, \dots, \alpha_p)$. On en déduit que

$$P(X_{t+2}|X_t = x_t, \dots, X_1 = x_1) \sim \mathcal{N}(\tilde{\alpha}_1 x_t + \dots + \tilde{\alpha}_p x_{t-p+1}, \tilde{\sigma}^2).$$

- En itérant le raisonnement précédent, on peut obtenir une expression de la forme

$$P(X_{t+h}|X_t = x_t, \dots, X_0 = x_0) \sim \mathcal{N}(\tilde{\alpha}_1(h)x_t + \dots + \tilde{\alpha}_p(h)x_{t-p+1}, \tilde{\sigma}(h)^2). \quad (3)$$

avec $(\tilde{\alpha}_1, \dots, \tilde{\alpha}_p, \tilde{\sigma}(h))$ qui s'expriment en fonction de $(\alpha_1, \dots, \alpha_p, \sigma)$.

On déduit en particulier de l'expression (3) que

$$E[X_{t+h}|X_1, \dots, X_t] = \tilde{\alpha}_1(h)X_t + \dots + \tilde{\alpha}_p(h)X_{t-p+1}$$

Par définition de l'espérance conditionnelle, c'est la meilleure prévision possible de X_{t+h} connaissant le processus jusqu'à l'instant t , en terme d'erreur quadratique.

Ceci permet de faire une prévision ponctuelle d'une valeur future du processus. On peut également calculer un intervalle de prévision (ou prédiction) qui contient la valeur future X_{t+h} avec une probabilité fixée α . Par exemple, pour $\alpha = 95\%$ on obtient

$$P(\tilde{\alpha}_1(h)x_t + \dots + \tilde{\alpha}_p(h)x_{t-p+1} - 1.96\sigma(h) \leq X_{t+h} \leq \tilde{\alpha}_1(h)x_t + \dots + \tilde{\alpha}_p(h)x_{t-p+1} + 1.96\sigma(h) | X_1 = x_1, \dots, X_t = x_t) = 0.95$$

Ceci permet de quantifier l'incertitude sur la prévision sous la forme de l'intervalle de prévision à 95% donné par

$$IP_{0.95} = [\tilde{\alpha}_1(h)x_t + \dots + \tilde{\alpha}_p(h)x_{t-p+1} - 1.96\sigma(h); \tilde{\alpha}_1(h)x_t + \dots + \tilde{\alpha}_p(h)x_{t-p+1} + 1.96\sigma(h)].$$

A noter que cet intervalle de prédiction est calculé en supposant que les paramètres sont connus. En général, ils sont inconnus et estimés à partir des données. L'intervalle de prédiction ne prend pas en compte l'incertitude liée à l'estimation des paramètres.

Les expressions se simplifient pour les modèles $AR(1)$. Par exemple, pour $h = 2$, on obtient

$$\begin{aligned} X_{t+2} &= \alpha X_{t+1} + \epsilon_{t+2} \\ &= \alpha^2 X_t + \alpha \epsilon_{t+1} + \epsilon_{t+2} \\ &= \alpha^2 X_t + \tilde{\epsilon}_{t+1} \end{aligned}$$

avec $\tilde{\epsilon}_{t+1} = \alpha \epsilon_{t+1} + \epsilon_{t+2}$. On en déduit que $P(X_{t+2}|X_0 = x_0, \dots, X_t = x_t) \sim \mathcal{N}(\alpha^2 x_t, \sigma^2(1 + \alpha^2))$. Et de manière générale, pour $h \geq 1$, on obtient

$$P(X_{t+h}|X_0 = x_0, \dots, X_t = x_t) \sim \mathcal{N}(\alpha^h x_t, \sigma(h)^2) \quad (4)$$

avec $\sigma(h)^2 = \sigma^2(1 + \alpha^2 + \dots + \alpha^{2(h-1)}) = \sigma^2 \frac{1 - \alpha^{2h}}{1 - \alpha^2}$. On peut remarquer que la prévision ponctuelle

$$E[X_{t+h}|X_0, \dots, X_t] = \alpha^h X_t$$

converge à une vitesse exponentielle vers l'espérance du processus (ici $E[X_t] = 0$) lorsque h croît (en finance on parle de 'phénomène de retour à la moyenne'). La variance de prévision

$$var[X_{t+h}|X_0, \dots, X_t] = \sigma^2 \frac{1 - \alpha^{2h}}{1 - \alpha^2}$$

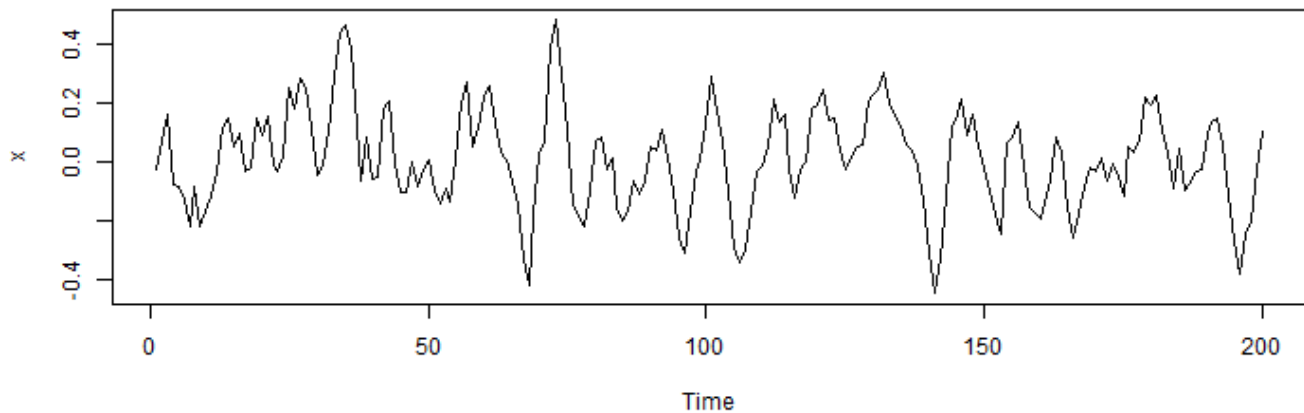
augmente avec h et converge vers la variance de la loi stationnaire $var(X_t) = \frac{\sigma^2}{1 - \alpha^2}$ lorsque $h \rightarrow +\infty$. En pratique, cela signifie que l'influence de la dernière observation décroît de manière exponentielle et que la meilleure prévision à long terme d'un processus $AR(1)$ est donnée par la loi stationnaire. Ce résultat se généralise aux modèles $ARMA$.

3.1.5 Modèles AR(p) et utilisation de R

Il existe différentes fonctions dans R pour travailler avec les modèles AR. Certains de ces fonctions peuvent être plus généralement utilisées pour les modèles ARIMA.

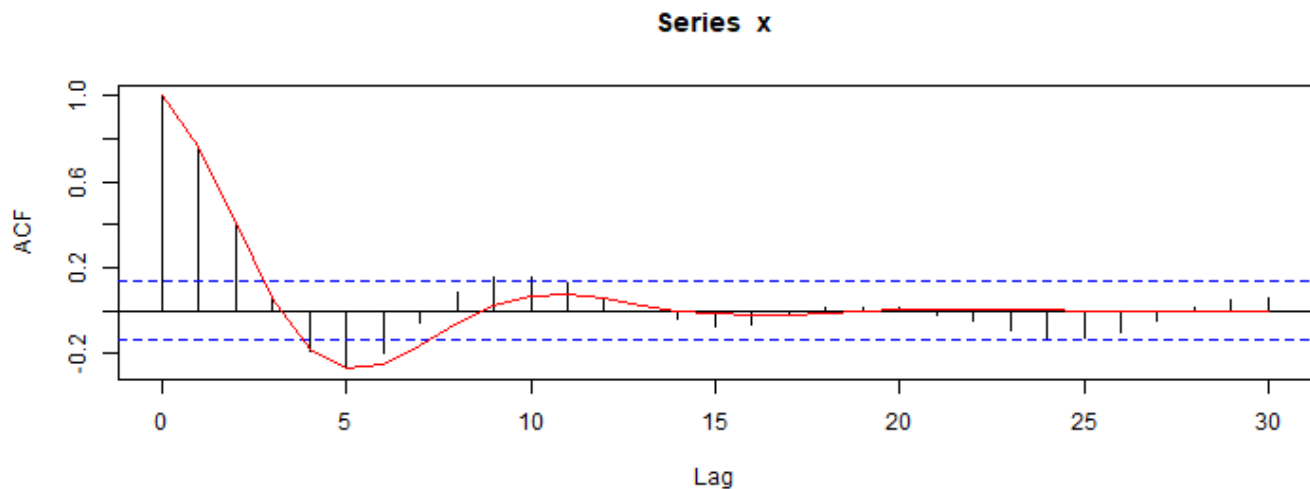
- **Simulation d'un modèle AR(p).** La fonction `arima.sim` permet de simuler les modèles arima en général, et donc les modèles ar en particulier.

```
#illustration avec le modèle AR(3)
#x(t)=x(t-1)-0.2x(t-2)-.2x(t-3)+e(t)
#avec var(e_t)=0.1^2
alpha=c(1,-.2,-.2)
sig=.1
p=3
x=arima.sim(model=list(ar=alpha),sd=sig,n=200)
plot(x)
```



- Calcul de la fonction d'autocorrélation théorique du modèle.

```
lmax=30
acfth=ARMAacf(ar=alpha,lag.max = lmax) #fonction autocorrélation théorique
acfemp=acf(x,lag.max = lmax) #fonction autocorrélation empirique
lines(acfemp$lag,acfth,col='red') #comparaison
```



- **Ajustement d'un modèle AR avec la fonction `ar`.** La fonction `ar` permet d'ajuster un modèle $AR(p)$ général. En général, l'ordre du modèle p est inconnu et la fonction `ar` cherche la valeur optimale de p en utilisant le critère AIC. L'option `aic=FALSE` permet d'enlever cette recherche du p optimal. L'argument `order.max` indique alors à R l'ordre p du modèle $AR(p)$ à ajuster. Par défaut, la fonction `ar` centre la série temporelle (en enlevant la moyenne empirique) avant d'ajuster le modèle AR. Cela revient à ajuster le modèle avec intercept

$$X_t - \mu = \alpha_1(X_{t-1} - \mu) + \dots + \alpha_p(X_{t-p} - \mu) + \epsilon_t \quad (5)$$

et à estimer μ par la moyenne empirique de la série temporelle. L'option `demean = FALSE` indique à R qu'il ne faut pas centrer les données avant d'ajuster les données.

```
ar(x,AIC=FALSE,order.max=p)  #si on connaît p
```

```
##
## Call:
## ar(x = x, order.max = p, AIC = FALSE)
##
## Coefficients:
##      1      2      3
## 0.9863 -0.1931 -0.1953
##
## Order selected 3  sigma^2 estimated as  0.009572
```

```
ar(x)  #si on ne connaît pas p
```

```
##
## Call:
## ar(x = x)
##
## Coefficients:
##      1      2      3
## 0.9863 -0.1931 -0.1953
##
## Order selected 3  sigma^2 estimated as  0.009572
```

```
y=x+1 #simulation processus AR non-centré
fit=ar(y)
fit$x.mean #estimation de la moyenne
```

```
## [1] 1.007406
```

- **Ajustement d'un modèle AR avec la fonction *arima*.** La fonction *arima* est la fonction de base pour ajuster des modèles ARMA (et plus généralement SARIMA) dans R. On verra dans la suite du cours qu'un modèle $AR(p)$ est un modèle $ARIMA(p, d, q)$ avec $d = 0$ (nombre de différentiations) et $q = 0$ (ordre de la partie MA). Par défaut, la fonction *arima* centre également la série temporelle (en enlevant la moyenne empirique) avant d'ajuster le modèle AR.

```
fit=arima(y,order=c(3,0,0)) #ajustement modèle AR(3)
fit
```

```
##
## Call:
## arima(x = y, order = c(3, 0, 0))
##
## Coefficients:
##          ar1          ar2          ar3  intercept
##      0.9865   -0.1870   -0.2055      1.0079
## s.e.  0.0690    0.0974    0.0694     0.0168
##
## sigma^2 estimated as 0.009246:  log likelihood = 183.89,  aic = -357.78
```

```
confint(fit)
```

```
##              2.5 %          97.5 %
## ar1      0.8513638  1.121675375
## ar2     -0.3778133  0.003809042
## ar3     -0.3414970 -0.069436730
## intercept 0.9750482  1.040708017
```

- **Prédiction.**

```
fit=ar(x,AIC=FALSE,order.max=p)
predict(fit,n.ahead=3) #prévision des 3 valeurs suivantes
```

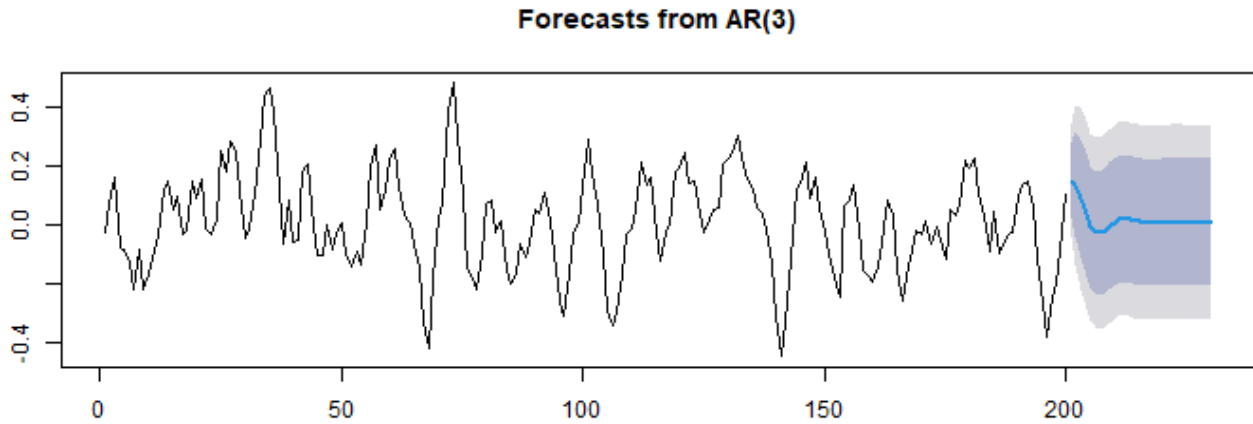
```
## $pred
## Time Series:
## Start = 201
## End = 203
## Frequency = 1
## [1] 0.14956375 0.13610991 0.08825671
##
## $se
## Time Series:
## Start = 201
## End = 203
## Frequency = 1
## [1] 0.09783629 0.13741393 0.15716486
```

```
library(forecast) #le package forecast permet de faire les représentations graphiques simplement
```

```
## Warning: le package 'forecast' a été compilé avec la version R 4.2.2
```

```
## Registered S3 method overwritten by 'quantmod':
##   method      from
## as.zoo.data.frame zoo
```

```
pm <- forecast(fit, h=30) #représentation graphique
plot(pm) # intervalles prédiction à 80% et 95%
```



3.1.6 Exercices sur le paragraphe 3.1

Exercice 6 On considère dans cet exercice le modèle $AR(1)$.

1. Dans la première partie de cet exercice, on travaille avec la fonction de vraisemblance conditionnée $\tilde{L}(\theta)$ définie dans le cours. Donner une expression analytique de l'estimateur du maximum de vraisemblance $\tilde{\theta} = \operatorname{argmax}_{\theta \in \Theta} (\tilde{L}(\theta))$.
2. Ecrire une fonction `R` `ar1.estim=function(X)` qui calcule l'estimation du maximum de vraisemblance $\tilde{\theta}$ associée à la série temporelle X . Tester votre fonction sur une série temporelle X simulée selon un modèle $AR(1)$.
3. Vérifiez que vous retrouvez les mêmes résultats avec votre fonction `ar1.estim` et la commande `ar(X, aic = FALSE, order.max = 1, demean=FALSE, method='ols')`
4. Il est également possible de calculer un estimateur du maximum de vraisemblance (non conditionnelle) avec la fonction `ar` et l'argument `method='mle'` (maximisation de la fonction L définie dans le cours). Comparer les deux méthodes d'estimation par simulation sur des séries temporelles de différentes longueurs. On vérifiera que les deux estimateurs sont convergents et donnent des résultats "similaires" lorsque la longueur de la série temporelle augmente.
5. Ecrire une fonction `predictionAR1=function(x,a,sig,hmax)` qui calcule l'espérance et l'écart-type de la distribution $L(X_{t+h}|X_0 = x_0, \dots, X_t = x)$ pour $h \in \{1, \dots, hmax\}$ pour un processus $AR(1)$ de paramètres a et σ .
6. Montrer à l'aide de simulations que l'intervalle $IP_{0.95}$ défini dans le cours est bien un intervalle de prévision à 95%.

Exercice 7 On considère le modèle $AR(2)$ défini par

$$X_t = \alpha_1 X_{t-1} + \alpha_2 X_{t-2} + \epsilon_t.$$

1. Vérifier que ce modèle admet une solution stationnaire lorsque $\alpha_1 = 0.9$ et $\alpha_2 = -0.3$.
2. Montrer que $\gamma(0), \dots, \gamma(5)$ est solution d'un système d'équations dont les coefficients dépendent de α_1, α_2 et σ^2 . Résoudre le système en utilisant `R` lorsque $\alpha_1 = 0.9$ et $\alpha_2 = -0.3$ et $\sigma = 1$.
3. Vérifier vos résultats en utilisant des simulations et la fonction `arima.sim` puis la fonction `ARMAacf` (toujours avec $\alpha_1 = 0.9, \alpha_2 = -0.3$ et $\sigma = 1$).

4. Ecrire une fonction `fitYWAR2=function(x)` qui ajuste un modèle $AR(2)$ à une série temporelle x en utilisant la méthode des moments décrite dans le paragraphe 3.1.2. Vérifier si on retrouve bien les mêmes résultats qu'avec la fonction `ar` (avec l'argument `method="yule-walker"`) sur des simulations.
5. Calculer la loi conditionnelle $\mathcal{L}(X_{t+2}|X_t = x_t, \dots, X_1 = x_1)$. A quoi peut servir ce résultat?

Exercice 8 Un modèle classique pour les taux d'intérêt (modèle de Vasicek) est basé sur l'EDS (processus d'Ornstein-Uhlenbeck)

$$dR_t = -\alpha(R_t - \mu)dt + \tau dW_t$$

avec $\{W_t\}$ un mouvement Brownien standard et μ, α et $\tau > 0$ des paramètres.

On peut montrer que si $\{R_t\}$ est solution de cette EDS alors

$$R_t = R_{t-1} \exp(-\alpha) + \mu(1 - \exp(-\alpha)) + \tau \int_{t-1}^t \exp(\alpha(s-t)) dW_s$$

avec $\int_{t-1}^t \exp(\alpha(s-t)) dW_s \sim \mathcal{N}\left(0, \frac{1 - \exp(-2\alpha)}{2\alpha}\right)$.

On suppose que l'on observe le processus à temps discret et on note $\{r_1, \dots, r_T\}$ la série temporelle associée (une unité de temps entre deux observations successives). Dans la suite de l'exercice, on suppose que $\mu = 0$.

1. Montrer que $\{R_t\}_{t \in \mathbb{N}}$ suit un modèle $AR(1)$ dont on exprimera les paramètres α_1^2 et σ en fonction de α et τ .
2. Pour quelles valeurs de α existe-t-il une solution stationnaire pour ce modèle $AR(1)$?
3. Télécharger les données historiques de l'EURIBOR :
https://data.ecb.europa.eu/data/datasets/FM/FM.M.U2.EUR.RT.MM.EURIBOR1MD_.HSTA
 Importer les données dans R. On créera un objet de type `ts` qui contient les données EURIBOR à 1 mois et on tracera la série temporelle.
4. Séparer les données en deux parties : la séquence d'apprentissage correspondra aux données jusqu'à 2012 (inclus) et la séquence de validation correspondra aux données à partir de 2013.
5. Tracer la fonction d'autocorrélation pour la séquence d'apprentissage. Cela vous semble-t-il raisonnable de supposer que la série temporelle suit un modèle $AR(1)$?
6. Ajuster un modèle $AR(1)$ sur la séquence d'apprentissage puis valider le modèle ajusté:
 - (a) tracer sur un même graphique la fonction d'autocorrélation empirique (calculée sur les données) et la fonction d'autocorrélation du modèle $AR(1)$ ajusté. Discuter.
 - (b) calculer le résidu empirique $\hat{\epsilon}_t = x_t - \hat{\alpha}_1 x_{t-1}$ puis vérifier si il a les propriétés attendues pour le résidu $\epsilon_t = X_t - \alpha_1 X_{t-1}$ d'un modèle $AR(1)$ gaussien.
7. Utiliser le modèle ajusté pour prédire la séquence de validation. Discuter les résultats obtenus.

3.2 Extension aux modèles ARMA

Les méthodes discutées dans le paragraphe précédent se généralisent aux modèles ARMA. Nous ne rentrerons pas dans les détails mathématiques dans le cadre de ce cours, mais nous nous focaliserons sur les aspects pratiques. Dans ce paragraphe, $\{X_t\}$ désigne un processus $ARMA(p, q)$ causal. Nous verrons dans le chapitre suivant que le modèle $ARMA(p, q)$ est aussi le modèle $ARIMA(p, 0, q)$.

3.2.1 Fonction d'autocovariance

On a vu que la fonction d'autocovariance d'un modèle AR peut être calculée en utilisant les équations de Yule-Walker.

Nous avons également vu en exercice (et dans la première partie du cours) que la fonction d'autocovariance d'un processus $MA(q)$ est définie par

$$X_t = \epsilon_t + \beta_1 \epsilon_{t-1} + \dots + \beta_q \epsilon_{t-q} \quad (6)$$

s'exprime sous la forme

$$\gamma(h) = \sigma^2 \sum_{j=0}^{q-h} \beta_j \beta_{j+h}$$

pour $h \in \{0, \dots, q\}$, avec la convention $\beta_0 = 1$, et $\gamma(h) = 0$ pour $h > q$.

Dans le cas des modèles $ARMA(p, q)$ généraux, il est possible de donner une expression pour la fonction d'autocovariance à partir de l'expression $MA(\infty)$ du processus, mais ces expressions ne sont pas toujours faciles à utiliser.

Une autre approche consiste à s'inspirer des méthodes vues pour les processus AR et MA. Plus précisément, on a

$$X_t - \alpha_1 X_{t-1} - \dots - \alpha_p X_{t-p} = \epsilon_t + \beta_1 \epsilon_{t-1} + \dots + \beta_q \epsilon_{t-q}$$

et donc, après multiplication par X_{t-h} pour $h \in \mathbb{N}$, on obtient

$$\text{cov}(X_t, X_{t-h}) - \alpha_1 \text{cov}(X_{t-1}, X_{t-h}) - \dots - \alpha_p \text{cov}(X_{t-p}, X_{t-h}) = \text{cov}(\epsilon_t, X_{t-h}) + \beta_1 \text{cov}(\epsilon_{t-1}, X_{t-h}) + \dots + \beta_q \text{cov}(\epsilon_{t-q}, X_{t-h})$$

c'est à dire

$$\gamma(h) - \alpha_1 \gamma(h-1) - \dots - \alpha_p \gamma(h-p) = \text{cov}(\epsilon_t, X_{t-h}) + \beta_1 \text{cov}(\epsilon_{t-1}, X_{t-h}) + \dots + \beta_q \text{cov}(\epsilon_{t-q}, X_{t-h}) \quad (7)$$

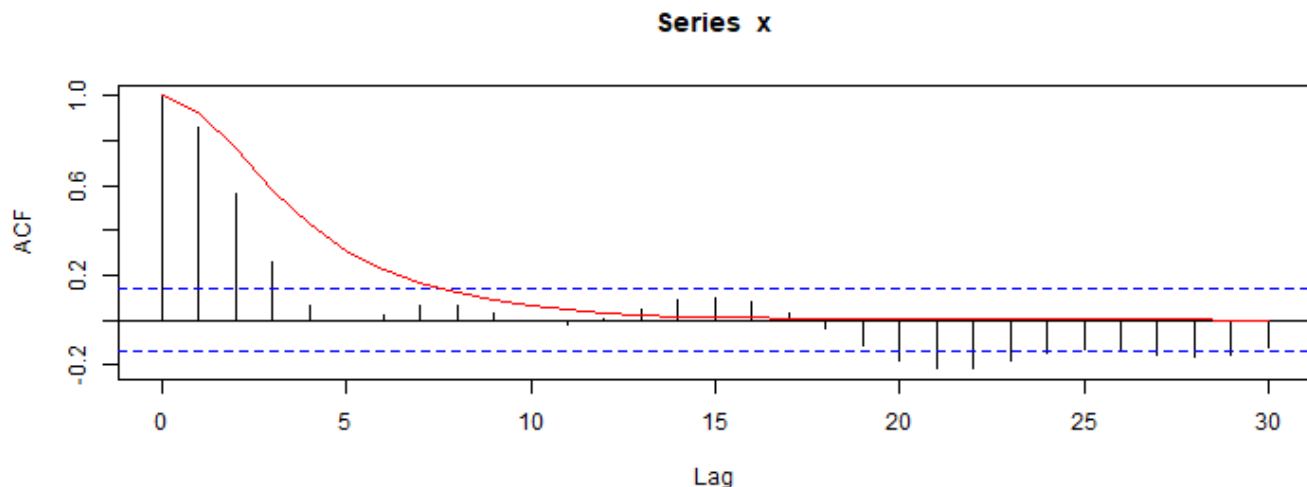
Par l'hypothèse de causalité, on a $\text{cov}(\epsilon_s, X_t) = 0$ si $s > t$, ce qui permet de simplifier l'expression précédente selon les valeurs de h . En particulier, on peut remarquer qu'on retrouve l'équation de Yule-Walker lorsque $h > q$. Il est aussi possible d'utiliser que, pour $s \in \mathbb{N}$,

$$\text{cov}(X_t, \epsilon_s) - \alpha_1 \text{cov}(X_{t-1}, \epsilon_s) - \dots - \alpha_p \text{cov}(X_{t-p}, \epsilon_s) = \text{cov}(\epsilon_t, \epsilon_s) + \beta_1 \text{cov}(\epsilon_{t-1}, \epsilon_s) + \dots + \beta_q \text{cov}(\epsilon_{t-q}, \epsilon_s) \quad (8)$$

avec $\text{cov}(\epsilon_s, \epsilon_{s'}) = \sigma^2 \mathbf{1}(s = s')$. En combinant les expressions (7) et (8) pour différentes valeurs de s , on peut obtenir un système d'équations pour la fonction d'autocovariance γ . L'exemple particulier des modèles $ARMA(1, 1)$ est traité dans l'exercice 9.

Il est possible d'utiliser la fonction R `ARMAacf` pour calculer numériquement la fonction d'autocorrélation d'une modèle $ARMA(p, q)$ lorsqu'on connaît ses paramètres.

```
lmax=30
alpha=c(1,-0.2) #partie AR
beta=c(1,1) #partie MA
x=arima.sim(model=list(ar=alpha,ma=beta),sd=1,n=200)
acfth=ARMAacf(ar=alpha,ma=beta,lag.max = lmax) #fonction autocorrélation théorique
acfemp=acf(x,lag.max = lmax) #fonction autocorrélation empirique
lines(acfemp$lag,acfth,col='red') #comparaison
```



3.2.2 Ajustement des modèles ARMA avec R

Estimation des paramètres lorsque p et q sont connus La fonction *arima* est la fonction de base pour ajuster des modèles ARMA (et plus généralement SARIMA) dans R. Lorsqu'on veut utiliser cette fonction pour ajuster un modèle ARMA, les principaux arguments d'entrée de la fonction sont les suivants

- *x* : série temporelle
- *order* : valeurs de p et q (supposés connus)
- *include.mean* : booléen TRUE/FALSE qui indique si il faut centrer la série avant d'ajuster le modèle

La méthode par défaut utilisée pour ajuster le modèle est la méthode du maximum de vraisemblance. Le calcul de la vraisemblance est fait en utilisant le filtre de Kalman et la fonction de vraisemblance est maximisée en utilisant une méthode d'optimisation numérique (fonction *optim* de R). Cette méthode repose sur l'hypothèse que les résidus $\{\epsilon_t\}$ sont gaussiens.

Parmi les sorties de la fonction *arima*, on récupère stocké dans une liste

- *coef* : estimation des paramètres $\{\alpha_i\}_{i \in \{1, \dots, p\}}$ et $\{\beta_j\}_{j \in \{1, \dots, q\}}$,
- *sigma2* : estimation de $\sigma^2 = \text{var}(\epsilon_t)$,
- *aic* : valeur de AIC
- *residuals* : estimation de la séquence d'innovation (ou résidu) $\{\epsilon_t\}$.

```
p=2 #valeur de p
q=2 #valeur de q
alpha=c(.9,-.2) #alpha
beta=c(.8,.5) #beta
sig=.1 #sigma
n=10^3
x=arima.sim(model=list(ar=alpha,ma=beta),sd=sig,n=n) #simulation ARMA
arima(x,order=c(p,0,q)) #ajustement modèle et affichage résultats
```

```
##
## Call:
## arima(x = x, order = c(p, 0, q))
```

```
##
## Coefficients:
##          ar1          ar2          ma1          ma2  intercept
##          0.8597   -0.1602   0.8262   0.4979   -0.0147
## s.e.    0.0570    0.0528   0.0501   0.0392    0.0244
##
## sigma^2 estimated as 0.01002:  log likelihood = 880.87,  aic = -1749.73
```

*#NB. Un modèle ARMA(p,q) est un modèle ARIMA(p,0,q)
#on retrouve des estimations 'proche' des vraies valeurs*

Sélection de modèle Les valeurs de p et q sont généralement inconnues. Une méthode possible pour choisir ces valeurs consiste à utiliser le critère AIC (ou BIC). On commence alors par se fixer des valeurs maximales p_{max} et q_{max} pour p et q puis on parcourt tous les modèles $ARMA(p,q)$ possibles avec $p \in \{0, \dots, p_{max}\}$ et $q \in \{0, \dots, q_{max}\}$ et on conserve le modèle qui a le plus petit AIC (ou BIC). On parle alors de recherche ‘exhaustive’ car on parcourt tous les modèles possibles, ce qui est peut être coûteux en temps de calcul (cf cours de modèle linéaire).

```
pmax=5 #valeur max pour p
qmax=5 #valeur max pour q
AIC=matrix(0,nrow=pmax+1,ncol=qmax+1) #tableau pour stocker les valeurs du AIC
for (p in 0:pmax){
  for (q in 0:qmax){
    AIC[p+1,q+1]=arima(x,order=c(p,0,q))$aic #calcul du AIC
  }
}
```

```
## Warning in arima(x, order = c(p, 0, q)): problème de convergence possible :
## optim renvoie un code = 1
```

AIC

```
##          [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,]   505.5559  -560.5532 -1285.741 -1613.522 -1702.167 -1730.313
## [2,] -1063.4317 -1518.4424 -1742.971 -1750.475 -1748.561 -1746.869
## [3,] -1604.2076 -1673.4239 -1749.734 -1748.597 -1746.606 -1744.797
## [4,] -1702.4506 -1700.8339 -1748.168 -1746.662 -1744.705 -1744.700
## [5,] -1701.2001 -1715.7247 -1746.890 -1744.759 -1744.879 -1743.155
## [6,] -1722.7627 -1731.3730 -1744.982 -1744.749 -1742.695 -1744.125
```

```
p=which(AIC == min(AIC), arr.ind = TRUE)[1]-1 #p avec AIC optimal
q=which(AIC == min(AIC), arr.ind = TRUE)[2]-1 #q avec AIC optimal
c(p,q)
```

```
## [1] 1 3
```

La fonction *auto.arima* du package *forecast* permet également de faire de la sélection de modèle en utilisant les critères AIC et BIC. Cette fonction permet plus généralement de faire de la sélection de modèle dans les modèles SARIMA qui incluent une composante saisonnière et une tendance (cf chapitre suivant) : les arguments *stationary = TRUE* et *seasonal = FALSE* permettent de restreindre la recherche du modèle optimal aux modèles ARMA. Par défaut, afin de réduire les temps de calcul, la fonction *auto.arima* utilise un algorithme de recherche approché de type stepwise et se restreint à des modèles $ARMA(p,q)$ avec $p \geq max.p = 5$, $q \geq max.q = 5$ et $p + q \leq max.order = 5$. Il faut changer certains arguments de la fonction si on veut faire une recherche exhaustive parmi tous les modèles $ARMA(p,q)$ avec $p \in \{0, \dots, p_{max}\}$ et $q \in \{0, \dots, q_{max}\}$.

```
auto.arima(x,stationary = TRUE,seasonal = FALSE)
```

```
## Series: x
## ARIMA(1,0,3) with zero mean
##
## Coefficients:
##          ar1      ma1      ma2      ma3
##      0.6280  1.0613  0.7249  0.1387
## s.e.  0.0392  0.0481  0.0616  0.0445
##
## sigma^2 = 0.01006: log likelihood = 881.06
## AIC=-1752.12   AICc=-1752.06   BIC=-1727.58
```

#argument par défaut : rapide, mais modèle non optimal

```
auto.arima(x,stationary = TRUE,seasonal = FALSE,stepwise=FALSE,max.order=10,approximation = FALSE)
```

```
## Series: x
## ARIMA(1,0,3) with zero mean
##
## Coefficients:
##          ar1      ma1      ma2      ma3
##      0.6280  1.0613  0.7249  0.1387
## s.e.  0.0392  0.0481  0.0616  0.0445
##
## sigma^2 = 0.01006: log likelihood = 881.06
## AIC=-1752.12   AICc=-1752.06   BIC=-1727.58
```

#recherche exhaustive : on retrouve les résultats précédents

```
auto.arima(x,stationary = TRUE,seasonal = FALSE,ic="bic",max.order=10)
```

```
## Series: x
## ARIMA(1,0,3) with zero mean
##
## Coefficients:
##          ar1      ma1      ma2      ma3
##      0.6280  1.0613  0.7249  0.1387
## s.e.  0.0392  0.0481  0.0616  0.0445
##
## sigma^2 = 0.01006: log likelihood = 881.06
## AIC=-1752.12   AICc=-1752.06   BIC=-1727.58
```

#en utilisant BIC et l'algorithme approché

#BIC sélectionne des modèles avec moins de paramètres

#(Pénalité plus forte sur le nombre de paramètres)

Un autre critère usuel pour la sélection de modèle est basé sur la validation croisée : on divise le jeu de données en deux parties, on conserve la première séquence pour ajuster le modèle (ensemble d'apprentissage) et la deuxième séquence pour calculer un critère de validation (ensemble d'apprentissage), usuellement l'erreur quadratique de prédiction à un pas de temps (MSE).

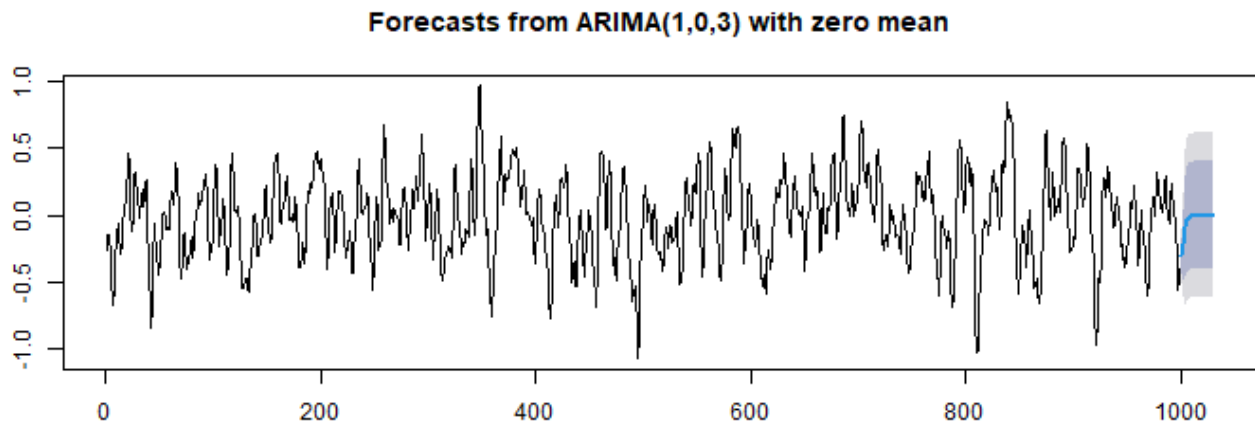
Prédiction.

Les fonctions *predict* et *forecast* permettent de faire des prédictions dans les modèles ARMA. Les algorithmes utilisés sont relativement complexes et ne sont pas détaillés dans le cadre de ce cours.

```
fit=arima(x,order=c(p,0,q)) #ajustement du modèle ARMA(p,q)
predict(fit,n.ahead=3) #prévision des 3 valeurs suivantes
```

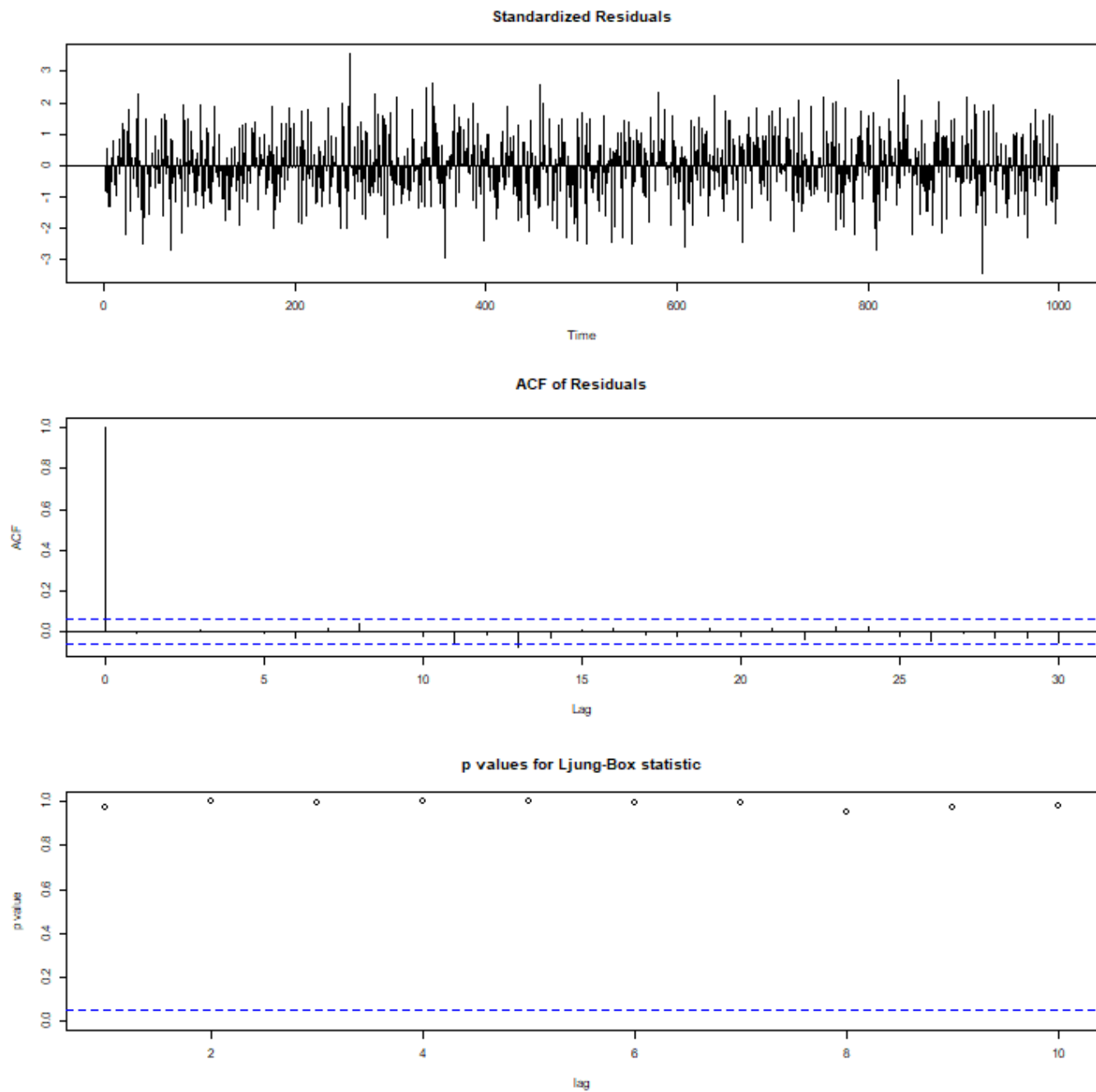
```
## $pred
## Time Series:
## Start = 1001
## End = 1003
## Frequency = 1
## [1] -0.3102733 -0.2247898 -0.1484421
##
## $se
## Time Series:
## Start = 1001
## End = 1003
## Frequency = 1
## [1] 0.1000779 0.1964395 0.2655141
```

```
fit=auto.arima(x,stationary = TRUE,seasonal = FALSE)
#ajustement du modèle arma, inclut la sélection de modèle
pm <- forecast(fit, h=30) #représentation graphique
plot(pm) # intervalles prédiction à 80% et 95%
```

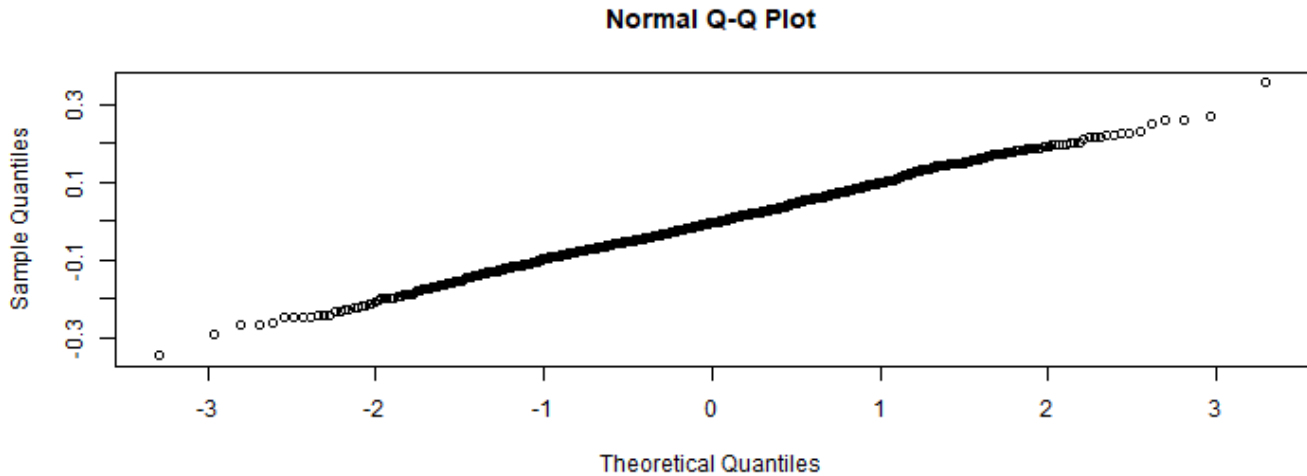


Validation de modèle Une fois le modèle ajusté, il est important de vérifier que les différentes hypothèses qui ont été faites sont valides. Pour cela, on va utiliser le résidu $\{\hat{\epsilon}_t\}$ estimé par la fonction *arima* et vérifier si cette séquence vérifie les hypothèses faites sur $\{\epsilon_t\}$. La fonction *tsdiag* propose différents graphiques pour vérifier si la séquence $\{\hat{\epsilon}_t\}$ est bien un bruit blanc (cf ci-dessous) : le premier graphique montre la série temporelle $\{\hat{\epsilon}_t\}$ et permet de vérifier visuellement si il existe une structure temporelle dans le résidu, le deuxième graphique montre la fonction d'autocorrélation empirique. On peut aussi vérifier la normalité du résidu en faisant un test de normalité ou un qqplot gaussien.

```
tsdiag(fit) #vérifie si résidu est un bruit blanc
```



```
qqnorm(fit$residuals) #vérifie si résidu gaussien
```



```
shapiro.test(fit$residuals)  #test si résidu gaussien
```

```
##
##  Shapiro-Wilk normality test
##
## data:  fit$residuals
## W = 0.99863, p-value = 0.6386
```

Exercice 9 On considère le modèle $ARMA(1,1)$ défini par

$$X_t - \alpha_1 X_{t-1} = \beta_1 \epsilon_{t-1} + \epsilon_t$$

1. A quelle condition ce modèle admet une solution stationnaire?
2. Montrer que $(\gamma(0), \gamma(1), \gamma(2))$ est solution d'un système d'équations dont les coefficients dépendent de α_1 , β_1 et σ^2 .
3. Vérifier vos résultats en utilisant des simulations lorsque $\alpha_1 = 0.8$, $\beta_1 = 0.5$ et $\sigma = 1$.
4. Ecrire une fonction `fitARMA11=function(x)` qui ajuste un modèle $ARMA(1,1)$ à une série temporelle x en utilisant la méthode des moments et les équations de la question 2.
5. Comparer les estimateurs obtenus avec votre fonction `fitARMA11` avec ceux obtenus avec la fonction `arima` en utilisant des simulations.

4 Modélisation des composantes non-stationnaires

Lorsque la série temporelle à modéliser n'est pas stationnaire, on commence généralement par chercher une transformation qui permet d'enlever les composantes non-stationnaire.

Il existe de nombreuses approches pour "stationnariser" une série temporelle. Dans la suite de ce chapitre, nous allons brièvement en présenter trois d'entre elles.

4.1 Différentiation et modèle SARIMA

Si $\{Y_t\}_{t \in \mathbb{Z}}$ est un processus, on note

- ΔY le processus défini par $\Delta Y_t = Y_t - Y_{t-1}$ pour $t \in \mathbb{Z}$,

- $\Delta^2 Y = \Delta(\Delta Y)$ le processus défini par $\Delta^2 Y_t = \Delta Y_t - \Delta Y_{t-1} = Y_t - 2Y_{t-1} + Y_{t-2}$ pour $t \in \mathbb{Z}$,
- et plus généralement par récurrence $\Delta^r Y = \Delta(\Delta^{r-1} Y)$ pour $r \in \{2, 3, \dots\}$.

L'exercice ci-dessous montre qu'appliquer l'opérateur de différentiation un certain nombre de fois permet de rendre stationnaire certaines séries temporelles non-stationnaires.

Exercice 10 Dans cet exercice $\{X_t\}_{t \in \mathbb{Z}}$ est un processus stationnaire.

1. (différentiation d'un processus stationnaire). Montrer que ΔX est un processus stationnaire, puis en déduire que $\Delta^p X$ est un processus stationnaire pour $p \in \mathbb{N}^*$.
2. (marche aléatoire généralisée). Soit $\{Y_t\}_{t \in \mathbb{Z}}$ un processus qui vérifie $Y_t = Y_{t-1} + X_t$ avec X_t indépendant de Y_{t-1} pour $t \in \mathbb{Z}$. Le processus Y est-il stationnaire? Le processus ΔY est-il stationnaire?
3. (tendance linéaire). Soit $\{Y_t\}_{t \in \mathbb{Z}}$ un processus qui vérifie $Y_t = \nu_0 + \nu_1 t + X_t$. Le processus Y est-il stationnaire? Le processus ΔY est-il stationnaire?
4. (tendance quadratique). Soit $\{Y_t\}_{t \in \mathbb{Z}}$ un processus qui vérifie $Y_t = \nu_0 + \nu_1 t + \nu_2 t^2 + X_t$. Le processus Y est-il stationnaire? Le processus ΔY est-il stationnaire? Le processus $\Delta^2 Y$ est-il stationnaire?

On peut retenir en particulier de l'exercice ci-dessus que l'application de l'opérateur Δ^p permet d'enlever une tendance (localement) polynomiale d'ordre p dans une série temporelle. Lorsqu'on différencie d fois une série temporelle avant de modéliser la série différenciée par un modèle ARMA, on obtient le modèle ARIMA (I pour Integrated). Ceci est formalisé dans la définition ci-dessous.

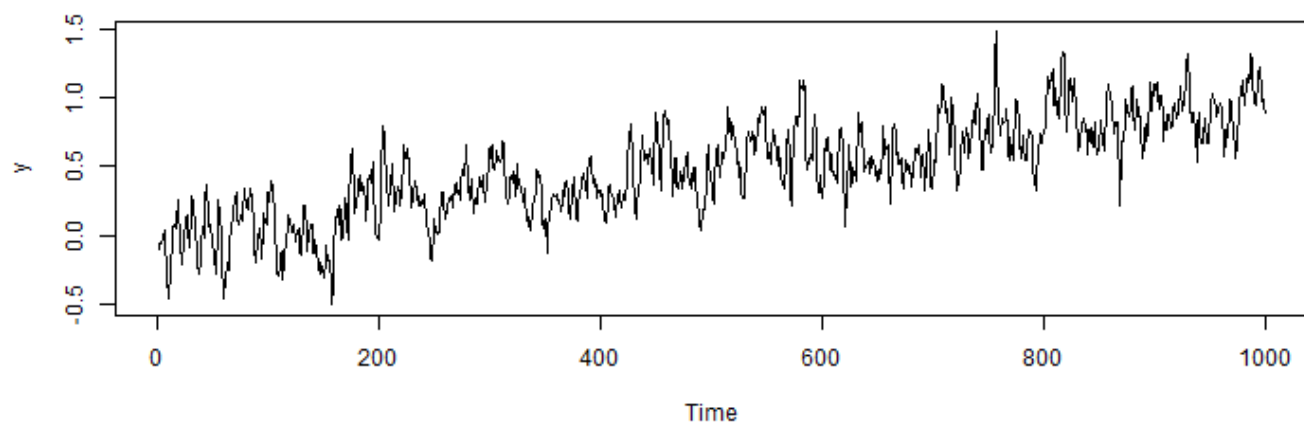
Définition 9 Le processus Y suit un modèle ARIMA(p, d, q) si $\Delta^d Y$ suit un modèle ARMA(p, q)

Exercice 11 Soit $\{Y_t\}_{t \in \mathbb{Z}}$ un processus qui vérifie $Y_t = \nu_0 + \nu_1 t + X_t$ avec X un processus ARMA(p, q). Montrer que $\Delta Y - \nu_1$ suit un modèle ARMA(p', q') et expliciter les paramètres de ce modèle.

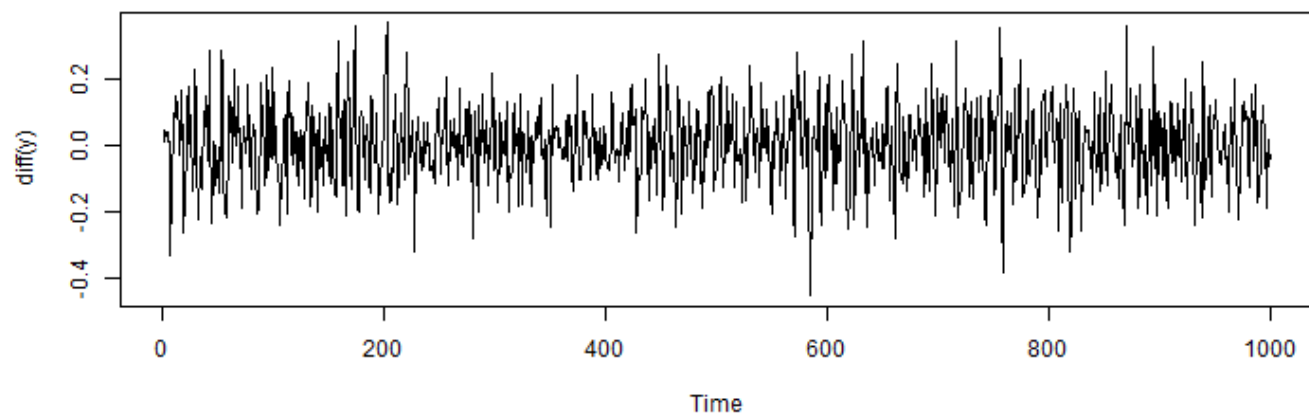
Illustration avec R.

Les codes ci-dessous illustrent le résultat de l'exercice dans le cas particulier où $p = q = 1$.

```
p=1 #valeur de p
q=1 #valeur de q
alpha=c(.7) #alpha
beta=c(.5) #beta
sig=.1 #sigma
T=10^3
x=arima.sim(model=list(ar=alpha,ma=beta),sd=sig,n=T) #simulation modèle ARMA(1,1)
y=x+(1:T)/T #ajout tendance linéaire
plot(y) #pas stationnaire
```



```
plot(diff(y)) #stationnaire
```



```
arima(diff(y),order=c(p,0,q+1))
```

```
##
## Call:
## arima(x = diff(y), order = c(p, 0, q + 1))
##
## Coefficients:
##      ar1      ma1      ma2  intercept
##    0.696 -0.4478 -0.5522      1e-03
## s.e.  0.025  0.0278  0.0277      1e-04
##
## sigma^2 estimated as 0.01001:  log likelihood = 879.84,  aic = -1749.68
```

```
#Ajustement d'un modèle ARIMA avec la fonction Arima du package forecast
Arima(y,order=c(p,1,q+1),include.drift = TRUE)
```

```
## Series: y
```

```
## ARIMA(1,1,2) with drift
##
## Coefficients:
##      ar1      ma1      ma2  drift
##      0.696 -0.4478 -0.5522 1e-03
## s.e.  0.025  0.0278  0.0277 1e-04
##
## sigma^2 = 0.01005: log likelihood = 879.84
## AIC=-1749.68  AICc=-1749.62  BIC=-1725.15
```

#L'option include.drift = TRUE ajoute un paramètre de moyenne pour la série différenciée

Soit $D \in \mathbb{N}^*$ et $\{Y_t\}_{t \in \mathbb{Z}}$ un processus. On note $\Delta_D Y$ la série temporelle définie par $\Delta_D Y_t = Y_t - Y_{t-D}$ pour $t \in \mathbb{Z}$ (attention $\Delta_D Y \neq \Delta^D Y$!). On peut appliquer l'opérateur Δ_D pour enlever une composante saisonnière de période D . En effet, supposons que

$$Y_t = S_t + Z_t$$

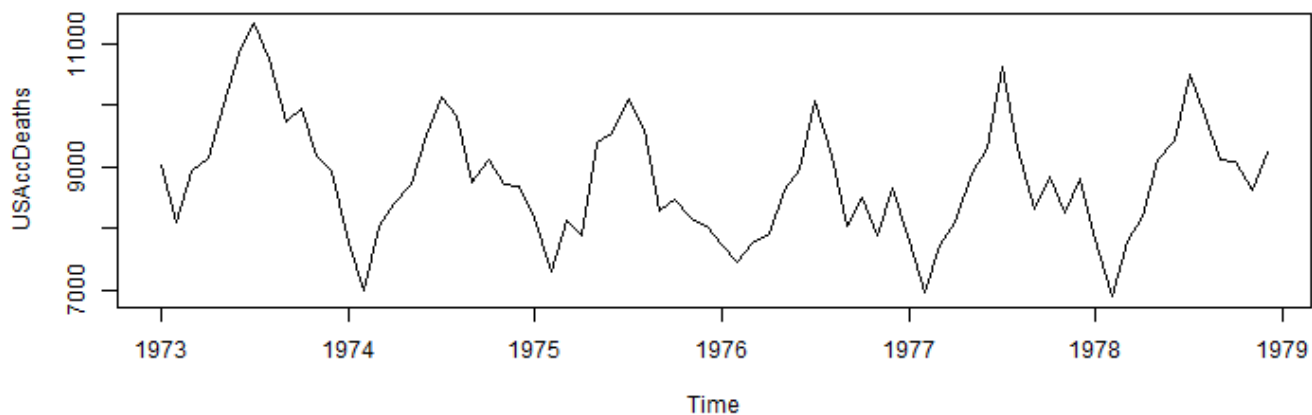
pour $t \in \{1, \dots, T\}$ avec $\{Z_t\}$ un processus sans composante saisonnière et $\{S_t\}$ la **composante saisonnière** qui est une fonction (déterministe) périodique de période D c'est à dire telle que $S_{t+D} = S_t$. On a alors

$$\Delta_D Z_t = Z_t - Z_{t-d}$$

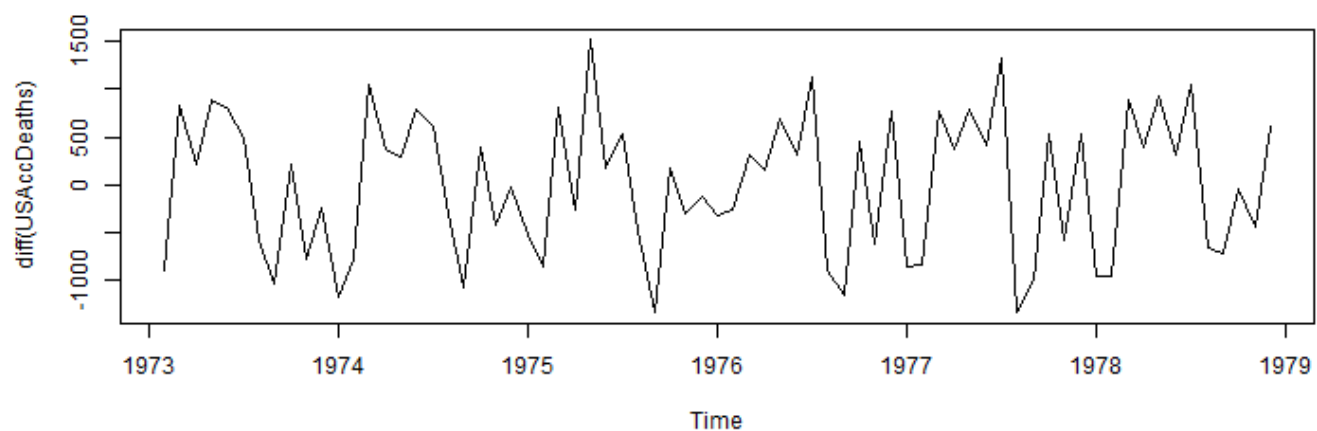
qui est un processus sans composante saisonnière. Cette idée est à la base des modèles SARIMA (S pour Seasonal). La définition générale des modèles SARIMA dépasse le cadre de ce cours.

Illustration avec R.

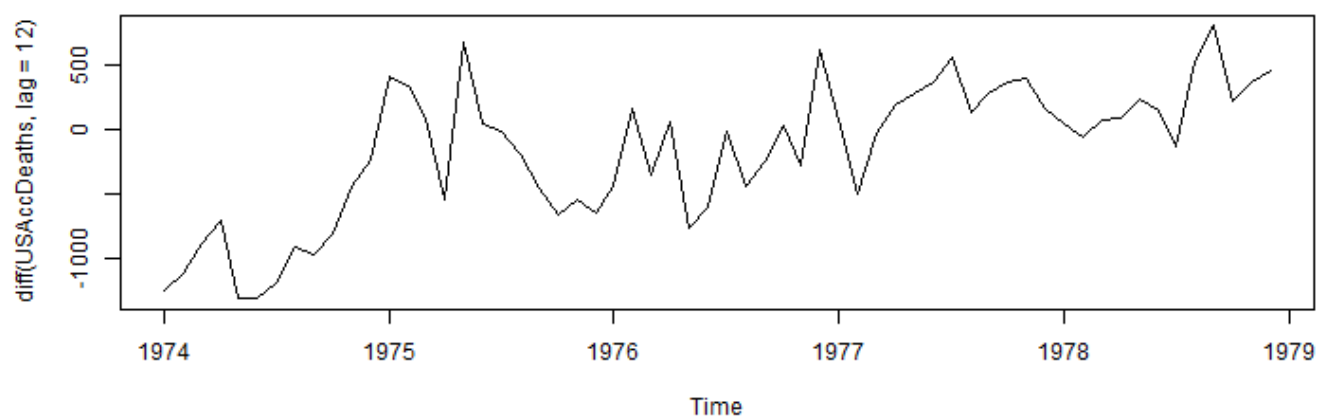
```
plot(USAccDeaths) #série avec tendance et saisonnalité
```



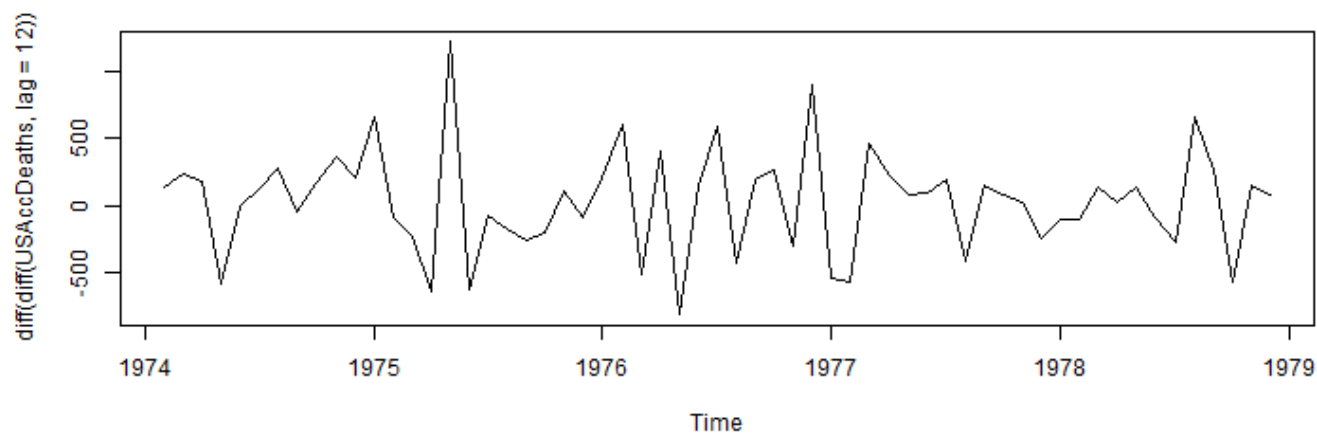
```
plot(diff(USAccDeaths)) #série sans tendance mais avec saisonnalité?
```



```
plot(diff(USAccDeaths,lag=12)) #série sans saisonnalité mais avec tendance?
```



```
plot(diff(diff(USAccDeaths,lag=12))) #série sans saisonnalité et sans tendance?
```



```
#à noter que l'application de différentiations raccourcit la longueur de la série temporelle
length(USAccDeaths)
```

```
## [1] 72
```

```
length(diff(diff(USAccDeaths,lag=12)))
```

```
## [1] 59
```

Exercice 12 Soit $\{Y_t\}_{t \in \mathbb{Z}}$ un processus. Montrer que $\Delta(\Delta_d X) = \Delta_d(\Delta_X)$.

4.2 Modèle paramétrique

Une autre approche classique pour modéliser une série temporelle non-stationnaire $\{y_t\}$ consiste à chercher un modèle paramétrique pour décrire les composantes non-stationnaires. Comme mentionné en introduction, on utilise généralement un modèle additif de la forme

$$Y_t = T_t + S_t + X_t \quad (9)$$

pour $t \in \{1, \dots, T\}$ avec $\{T_t\}$ la **tendance**, $\{S_t\}$ la **composante saisonnière** et $\{X_t\}$ un processus stationnaire. $\{Y_t\}$ et $\{X_t\}$ sont des processus stochastiques alors que $\{S_t\}$ et $\{T_t\}$ sont des fonctions déterministes. On suppose de plus que $\{S_t\}$ est périodique ($S_t = S_{t+D}$ avec D le nombre d'observations par cycle, par exemple $D = 12$ pour des données mensuelles avec un cycle annuel ou $D = 24$ pour des données horaires avec un cycle journalier). On peut supposer, sans perte de généralité, que $\sum_{t=1}^D S_t = 0$.

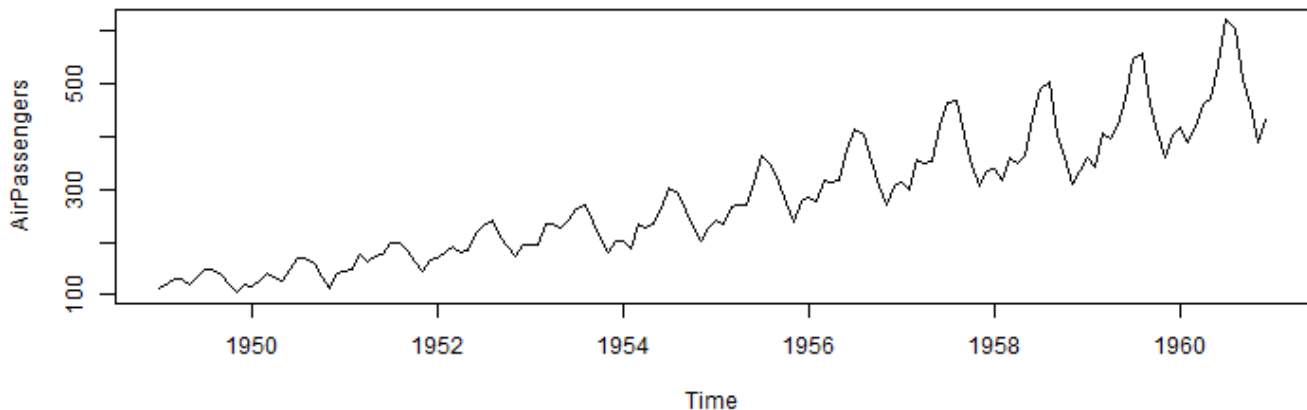
Il est parfois plus pertinent d'utiliser un modèle multiplicatif de la forme

$$Y_t = T_t S_t X_t \quad (10)$$

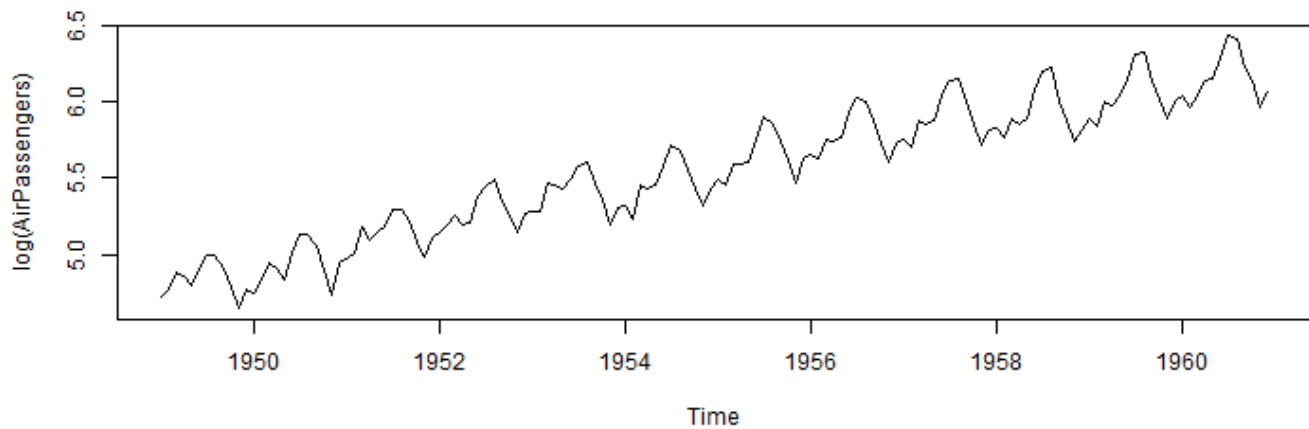
et on peut alors se ramener à un modèle additif en utilisant la fonction $\ln$.

Illustration avec R.

```
plot(AirPassengers) #l'amplitude du cycle saisonnier varie au cours du temps
```



```
#un modèle additif n'est pas adapté
plot(log(AirPassengers)) #l'amplitude du cycle saisonnier est 'stabilisée'
```



#un modèle multiplicatif est plus adapté?

On choisit ensuite une forme paramétrique pour les fonctions $\{T_t\}$ et $\{S_t\}$. Par exemple, on peut modéliser $\{T_t\}$ par un polynôme de degré r c'est à dire supposer que

$$T_t = \nu_0 + \nu_1 t + \dots + \nu_r t^r.$$

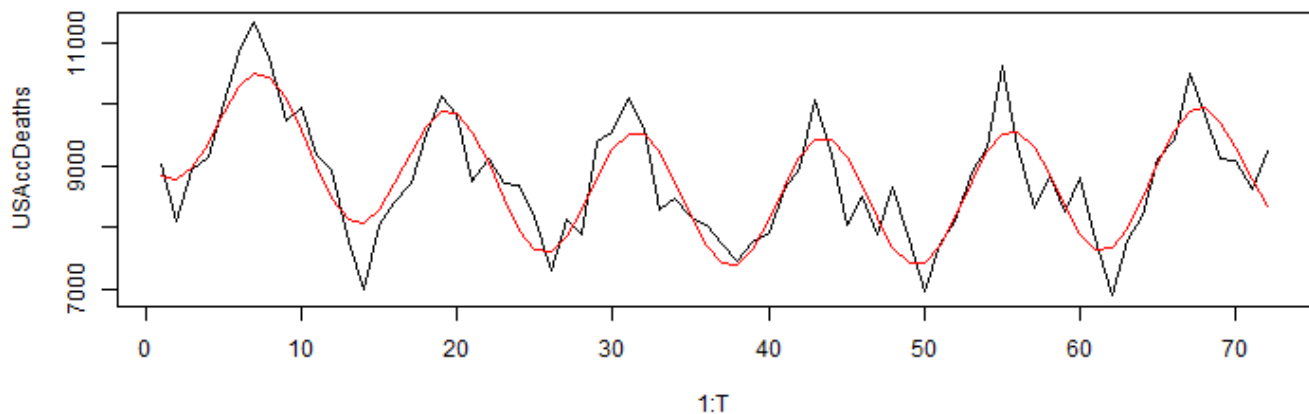
La fonction $\{S_t\}$ est usuellement représentée en utilisant un polynôme trigonométrique (série de Fourier tronquée), c'est à dire sous la forme

$$S_t = \kappa_0 + \kappa_1 \cos\left(\frac{2\pi t}{D}\right) + \lambda_1 \sin\left(\frac{2\pi t}{D}\right) + \kappa_2 \cos\left(\frac{4\pi t}{D}\right) + \lambda_2 \sin\left(\frac{4\pi t}{D}\right) + \dots$$

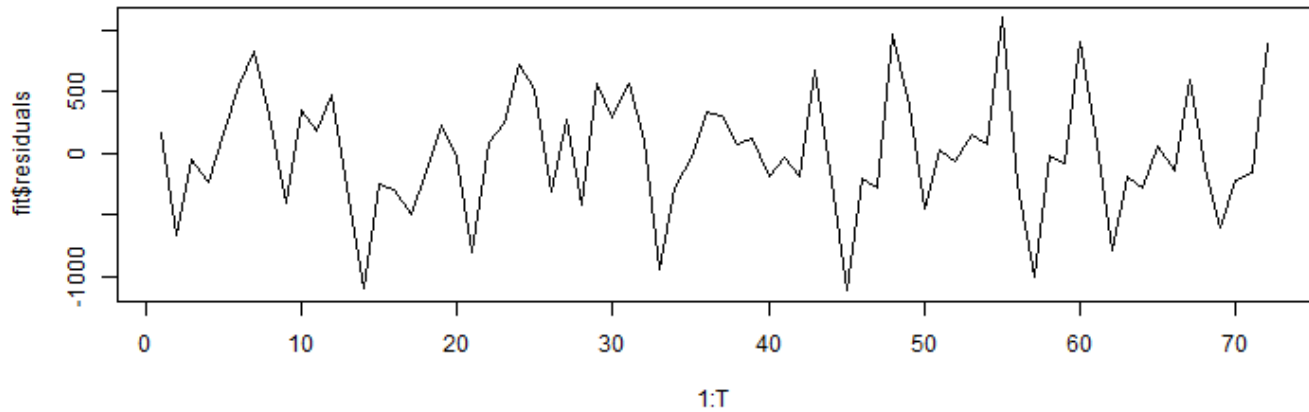
Les différents paramètres peuvent être estimés par la méthode des moindres carrés.

Illustration avec R.

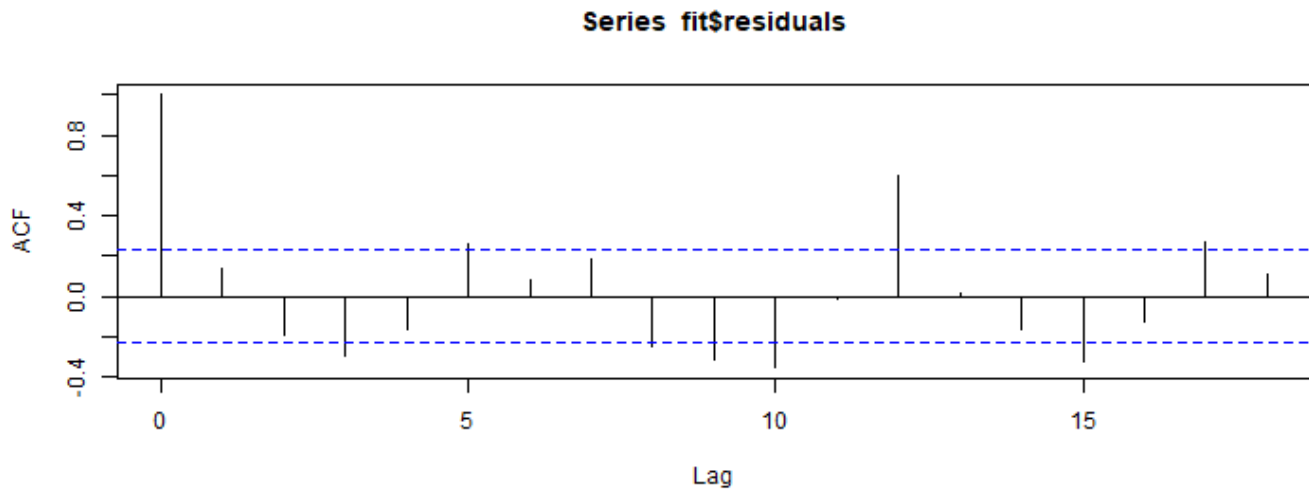
```
T=length(USAccDeaths)
data=data.frame(USAccDeaths,t=1:T,t2=(1:T)^2,cos=cos(2*pi*(1:T)/12),sin=sin(2*pi*(1:T)/12))
fit=lm(USAccDeaths~.,data=data)
plot(1:T,USAccDeaths,type='l') #série temporelle initiale
lines(1:T,fit$fitted.values,col='red') #tendance et composantes saisonnières estimées
```



```
#La composante saisonnière ne semble pas être bien estimée
plot(1:T,fit$residuals,type='l') #série résiduelle
```



```
acf(fit$residuals)
```



#on retrouve une corrélation positive à 12 mois : il reste une composante saisonnière

4.3 Lissage non-paramétrique

On se place toujours dans le cadre du modèle additif (9). Dans certains cas, il est difficile de trouver une forme paramétrique adaptée pour décrire les composantes non stationnaires $\{T_t\}$ et $\{S_t\}$. On peut alors utiliser des estimateurs non-paramétriques.

4.3.1 Moyennes mobiles et fonction *decompose* de R

Définition 10 Si $\{y_t\}_{t \in \{1, \dots, T\}}$ est une série temporelle et $l \in \mathbb{N}$, on appelle *moyenne mobile* la série temporelle définie par $\bar{y}_t^l = \frac{1}{2l+1} \sum_{i=t-l}^{t+l} y_t$ pour $t \in \{l+1, \dots, T-l\}$. $L = 2l+1$ est appelée 'largeur de fenêtre'. A noter que $\bar{y}_t^L$ n'est pas défini pour $t \leq l$ et $t > T-l$.

On peut étendre la définition ci-dessus au cas où L est paire de différentes manières. Par exemple, avec la fonction *filter* de R “the length of the filter should be odd, but if it is even, more of the filter is forward in time than backward”. L’exercice ci-dessous suggère que l’application d’une moyenne mobile peut permettre de faire ressortir la tendance.

Exercice 13 On suppose que le processus $\{Y_t\}$ suit un modèle additif de la forme

$$Y_t = \nu_0 + \nu_1 t + S_t + X_t$$

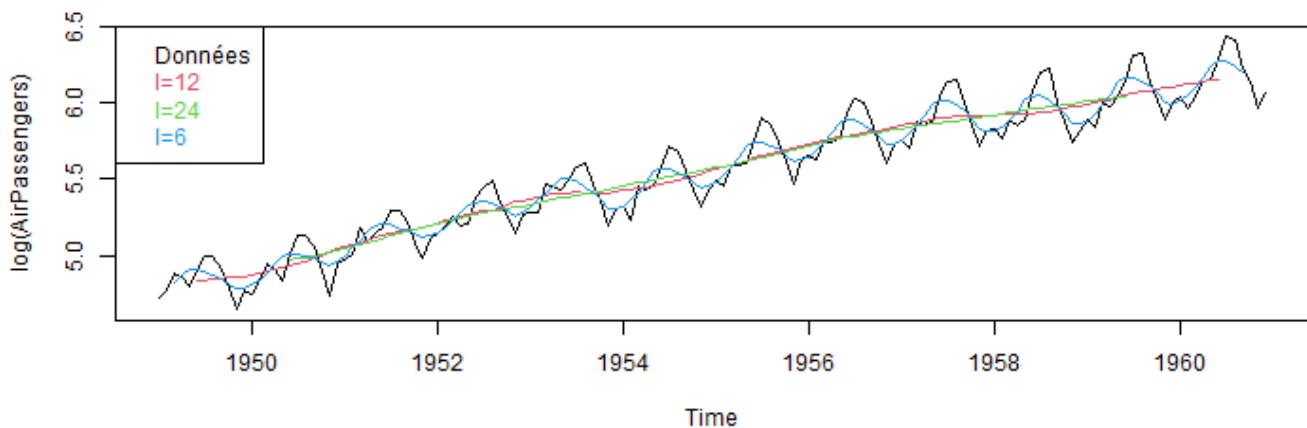
avec $(\nu_0, \nu_1) \in \mathbb{R}^2$ et $\{S_t\}$ une fonction périodique de période $L = 2l + 1$ telle que $\sum_{t=1}^L S_t = 0$.

1. Exprimer la moyenne mobile $\bar{Y}_t^L$ et fonction de $\bar{X}_t^L$.
2. En déduire que $\bar{Y}_t^L$ suit un modèle additif dont on précisera la tendance et la composante saisonnière.

De manière générale, appliquer une moyenne mobile permet de ‘lisser’ une série temporelle en ‘filtrant’ les composantes haute fréquence (c’est à dire qui évoluent rapidement dans le temps) et en faisant ressortir les composantes basse fréquence (c’est à dire qui évoluent lentement dans le temps comme la tendance). On choisit généralement la largeur de la fenêtre L comme un multiple de D pour gommer les fluctuations saisonnières. Plus L est grand, et plus la moyenne mobile est ‘lisse’.

Illustration avec R.

```
plot(log(AirPassengers)) #Estimons la tendance par moyenne mobile
L=12 #fenêtre de largeur 1 an
hatx=stats::filter(log(AirPassengers),filter=rep(1/(L),L))
lines(hatx,col=2) #estimateur non paramétrique de la tendance
L=36 #fenêtre de largeur 3 ans
hatx=stats::filter(log(AirPassengers),filter=rep(1/(L),L))
lines(hatx,col=3) #lissage plus important
L=6 #fenêtre de largeur 6 mois
hatx=stats::filter(log(AirPassengers),filter=rep(1/(L),L))
lines(hatx,col=4) #composante saisonnière toujours présente
legend('topleft',legend=c('Données','l=12','l=24','l=6'),text.col = 1:4)
```



Une fois la tendance estimée, on peut la retirer de la série temporelle initiale. On est alors ramené à estimer la composante saisonnière dans le modèle additif sans tendance

$$Y_t = S_t + X_t \quad (11)$$

pour $t \in \{1, \dots, T\}$ avec $\{X_t\}$ un processus stationnaire. On peut supposer sans perte de généralité que $E[X_t] = 0$. On a alors $E[Y_t] = S_t = E[Y_{t'}]$ si $t = t' \bmod D$ et un estimateur naturel de S_t est donc la moyenne des observations disponibles à une date donnée

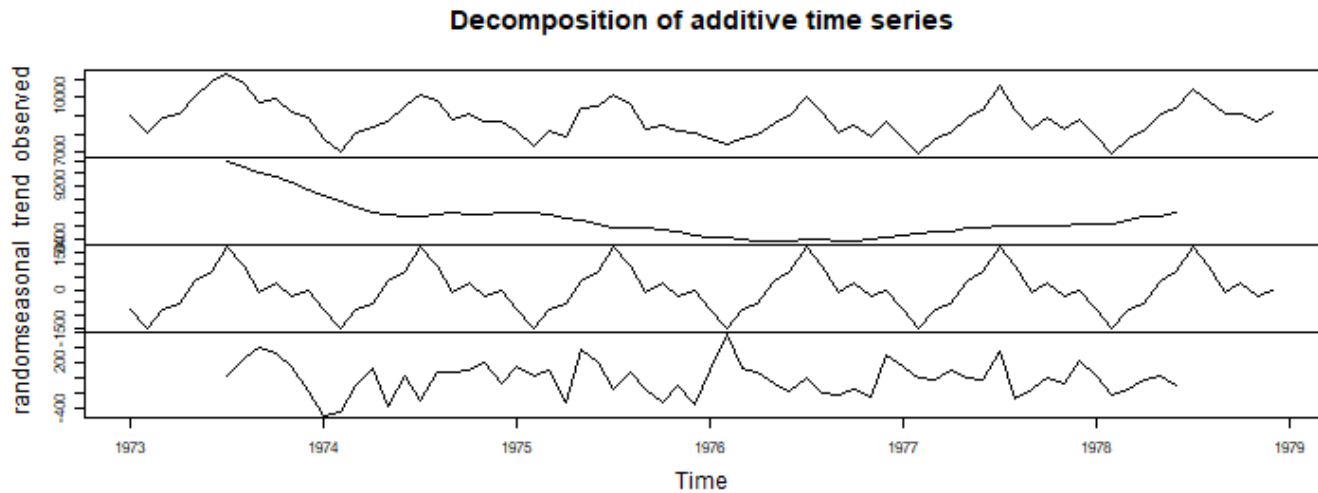
$$\hat{S}_t = \frac{1}{\text{card}\{t' \in \{1, \dots, T\} | t = t' \bmod D\}} \sum_{t' \in \{1, \dots, T\} | t = t' \bmod D} Y_{t'}$$

(par exemple la moyenne des données disponibles en janvier fournira une estimation de la composante saisonnière en janvier pour des données mensuelles). Cet estimateur est généralement appelé “moyennes saisonnières”.

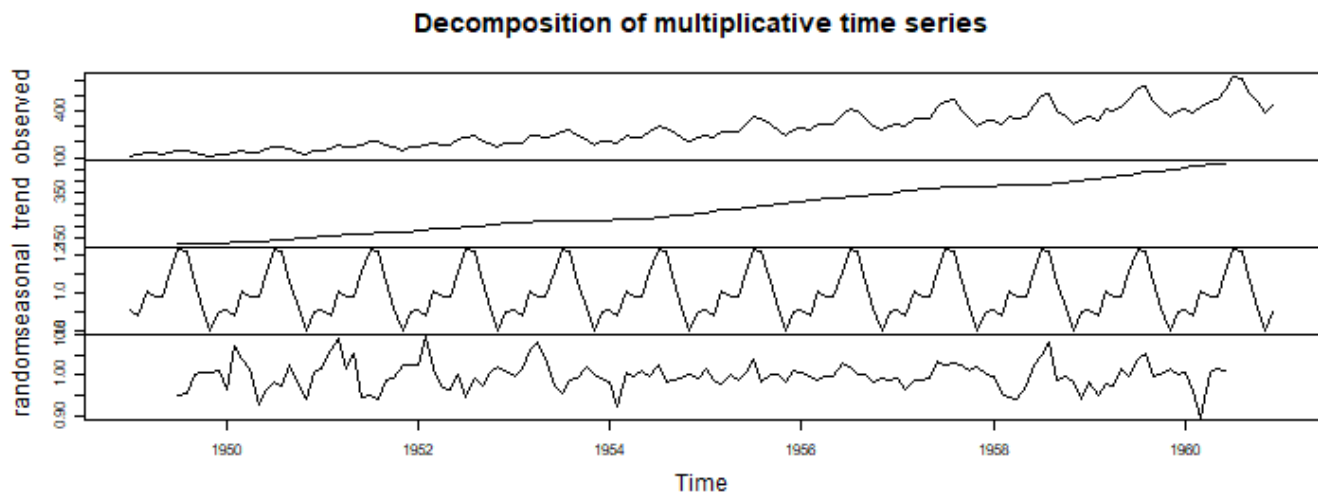
L’approche décrite ci-dessus, à savoir l’application d’une moyenne mobile pour estimer la tendance (avec une fenêtre de largeur 1 an par défaut) puis calcul des moyennes saisonnières sur la série sans tendance, est implémentée dans la fonction `decompose` de R.

Illustration avec R.

```
plot(decompose(USAccDeaths))
```



#NB. l'argument type permet de spécifier directement un modèle multiplicatif
`plot(decompose(AirPassengers, type='multiplicative'))`

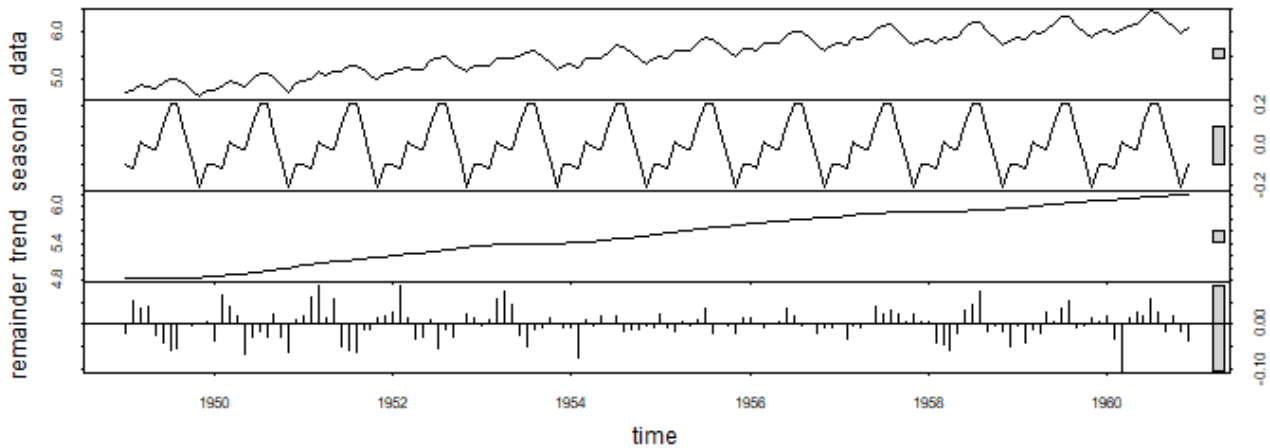


Un inconvénient des méthodes basées sur les moyennes mobiles est la perte d'observations au début et à la fin de la séquence ($\hat{y}_t^{2l+1}$ n'est pas définie pour $t \leq l + 1$ et $t \geq T - l$). En particulier, ces méthodes ne permettent pas de faire des prédictions pour $t > T$.

4.3.2 Régression linéaire locale et fonction *stl* de R

La fonction *stl* utilise le même principe que la fonction *decompose*, mais remplace les moyennes mobiles par une méthode plus sophistiquée de lissage basée la régression polynomiale locale ajustée la méthode des moindres carrés https://en.wikipedia.org/wiki/Local_regression

```
plot(stl(log(AirPassengers), "per"))
```



Exercice 14 On considère la série temporelle qui décrit les températures journalières depuis 1945 à Brest (Guipavas) disponible sur la page du cours. A noter qu'il faut diviser les données de température (colonne nommée TG dans le jeu de données) par 10 pour obtenir des degrés Celsius. Les données proviennent du site <https://www.ecad.eu/>.

1. Importer les données dans R et créer un objet de type *ts*.
2. Tracer la série temporelle. La série vous semble-t-elle être stationnaire? Peut-on identifier une tendance? Peut-on identifier une composante saisonnière?
3. Découper la série temporelle en 2 parties : la période 1945-2022 sera utilisée pour l'étape de modélisation, et le mois de janvier 2023 sera utilisé en validation.
4. Analyser la série temporelle à l'aide de la fonction *decompose*.
 - (a) Vérifier que la tendance estimée par la fonction *decompose* est égale à la moyenne mobile avec une fenêtre de largeur 1 an.
 - (b) Tracer sur une même figure la moyenne mobile avec des fenêtres de largeur 1 an, 5 ans, 10 ans et 30 ans. Discuter les résultats obtenus : quelle largeur de fenêtre vous semble être la plus adaptée pour estimer la tendance?
 - (c) Enlever la tendance estimée par la fonction *decompose* de la série initiale puis calculer la série des moyennes saisonnières. Vérifier qu'on retrouve bien la composante saisonnière estimée par la fonction *decompose*. Discuter les résultats obtenus.
5. Proposer un modèle paramétrique pour décrire les composantes non-stationnaires. On notera $\{x_t\}$ la série "stationnarisée". Comparer avec les résultats données par *decompose*.
6. Proposer un modèle ARMA pour décrire la série temporelle $\{x_t\}$ et valider le modèle proposé.
7. Utiliser le modèle des deux questions précédentes pour prédire les températures du mois de janvier 2023. Discuter les résultats obtenus.

5 References

Brockwell P.J. et Davis R.A. (2001) Introduction to Time Series and Forecasting, Springer.

Shumway, R. H., & Stoffer, D. S. (2017). Time series analysis and its applications: with R examples. Springer.
<https://www.stat.pitt.edu/stoffer/tsa4/tsa4.pdf>

Ce cours est largement inspiré des cours ci-dessous, disponibles en ligne

<https://www.math.sciences.univ-nantes.fr/~philippe/download/coursTime-seriesM2.pdf>

<http://math.univ-lille1.fr/~viano/economcours.pdf>

http://perso.univ-rennes1.fr/valerie.monbet/ST_M1/CoursST2017_1.pdf

<http://eric.univ-lyon2.fr/~jjacques/Download/Cours/ST-Cours.pdf>