

# EXAMEN DU 1<sup>ER</sup> JUIN 2011

## CORRECTION

**Exercice 1** Soient  $Z$  et  $W$  deux bruits blancs forts gaussiens indépendants, de variances  $\sigma_Z^2$  et  $\sigma_W^2$ . Soient  $\phi$  et  $\theta$  deux nombres réels de valeur absolue strictement inférieure à 1. On définit un  $AR(1)$   $U$  et un  $MA(1)$   $T$  par :

$$\forall t \in \mathbb{Z}, U_t = \phi U_{t-1} + Z_t,$$

et

$$\forall t \in \mathbb{Z}, T_t = W_t + \theta W_{t-1}.$$

Enfin, on définit  $Y$  par :

$$\forall t \in \mathbb{Z}, Y_t = U_t + T_t.$$

### 1. Partie théorique

- (a) Le processus  $Y$  est-il stationnaire ? Calculer sa fonction d'autocovariance.

**Solution.** On calcule, pour tous  $t$  et  $h$  dans  $\mathbb{Z}$  :

$$\mathbb{E}(Y_t) = \mathbb{E}(U_t) + \mathbb{E}(T_t) = 0,$$

et en utilisant que  $U$  et  $T$  sont indépendants (car  $Z$  et  $W$  le sont) :

$$\gamma_Y(h) = \mathbb{E}(Y_t Y_{t-h}) = \mathbb{E}(U_t U_{t-h}) + \mathbb{E}(T_t T_{t-h}) = \gamma_U(h) + \gamma_T(h).$$

Or comme  $|\phi| < 1$ ,  $U$  est causal et :

$$\gamma_U(h) = \frac{\sigma_Z^2}{1 - \phi^2} \phi^{|h|}.$$

Par ailleurs, pour le  $MA(1)$   $T$  on sait que :

$$\gamma_T(h) = \begin{cases} \sigma_W^2(1 + \theta^2) & \text{si } h = 0 \\ \sigma_W^2\theta & \text{si } |h| = 1 \\ 0 & \text{si } |h| \geq 2 \end{cases}.$$

Donc :

$$\gamma_Y(h) = \begin{cases} \frac{\sigma_Z^2}{1 - \phi^2} + \sigma_W^2(1 + \theta^2) & \text{si } h = 0 \\ \frac{\sigma_Z^2}{1 - \phi^2} \phi + \sigma_W^2\theta & \text{si } |h| = 1 \\ \frac{\sigma_Z^2}{1 - \phi^2} \phi^{|h|} & \text{si } |h| \geq 2 \end{cases}.$$

- (b) Donner un modèle d'état pour l'observation de  $(Y_t)_{t \geq 0}$ .

**Solution.** On s'inspire du modèle d'état pour les  $ARMA$  : on va faire évoluer séparément les parties  $AR$  de  $U$  et  $T$  puis observer une somme de moyennes mobiles de ces deux parties  $AR$ . On pose :

$$\vec{X}_t = \begin{pmatrix} U_t \\ W_t \\ W_{t-1} \end{pmatrix}.$$

Alors  $X_t$  satisfait l'équation d'état :

$$\forall t \geq 0, \quad \overrightarrow{X_{t+1}} = F \overrightarrow{X_t} + \overrightarrow{V_t},$$

avec :

$$\overrightarrow{V_t} = \begin{pmatrix} Z_t \\ W_{t+1} \\ 0 \end{pmatrix},$$

et :

$$F = \begin{pmatrix} \phi & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

Puis on obtient  $Y$  par l'équation d'observation (sans bruit d'observation) suivante :

$$Y_t = G \overrightarrow{X_t},$$

avec :

$$G = \begin{pmatrix} 1 & 1 & \theta \end{pmatrix}.$$

Il faut voir enfin que  $(\overrightarrow{X_0}, (\overrightarrow{V_t})_{t \geq 0})$  forme une famille de v.a décorrélées, ce qui est évident, vue la causalité de  $U$  et  $T$  et le fait que  $Z$  et  $W$  sont indépendants. Remarquons enfin que la matrice de covariance du bruit d'état est :

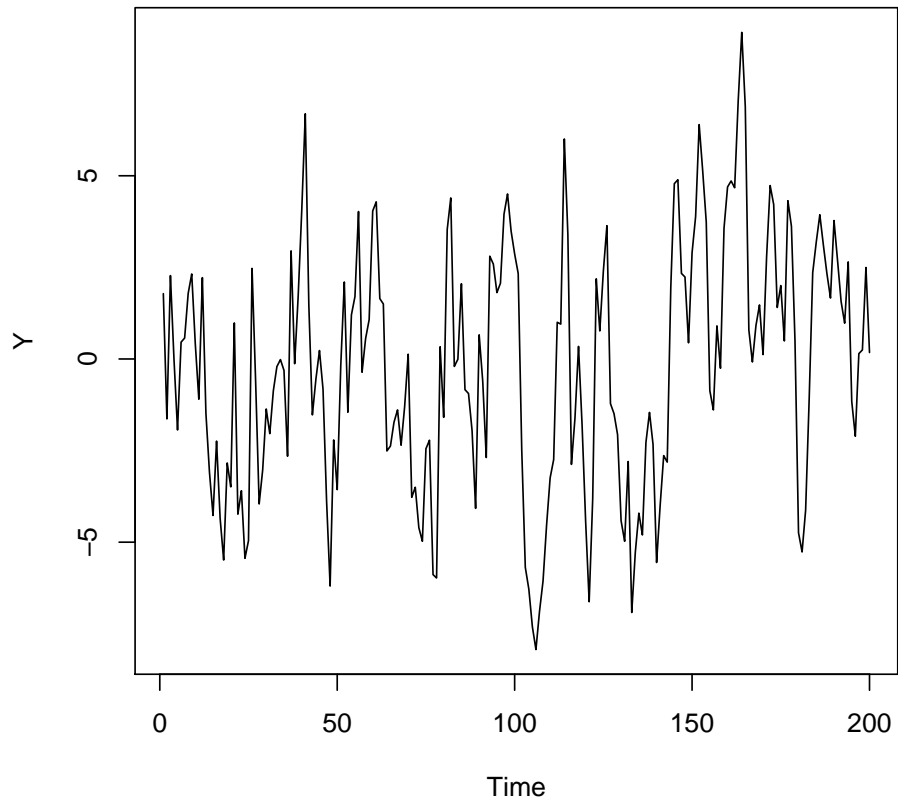
$$Q = \begin{pmatrix} \sigma_Z^2 & 0 & 0 \\ 0 & \sigma_W^2 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

## 2. Partie pratique

- (a) Simuler une réalisation de  $(Y_1, \dots, Y_n)$  pour  $n = 200$ ,  $\sigma_Z = 2$ ,  $\sigma_W = 1$ ,  $\phi = 4/5$  et  $\theta = 1/2$  et la visualiser. Tracer sur un même graphique l'autocovariance théorique du processus  $Y$  et l'autocovariance empirique de cette réalisation, pour les lags  $h = 0, \dots, 40$ . Commenter.

**Solution.**

```
> n <- 200
> sigmaZ <- 2
> sigmaW <- 1
> phi <- 4/5
> theta <- 1/2
> Y <- arima.sim(n, model = list(ar = phi), sd = sigmaZ) + arima.sim(n,
+   model = list(ma = theta), sd = sigmaW)
> plot(Y)
```

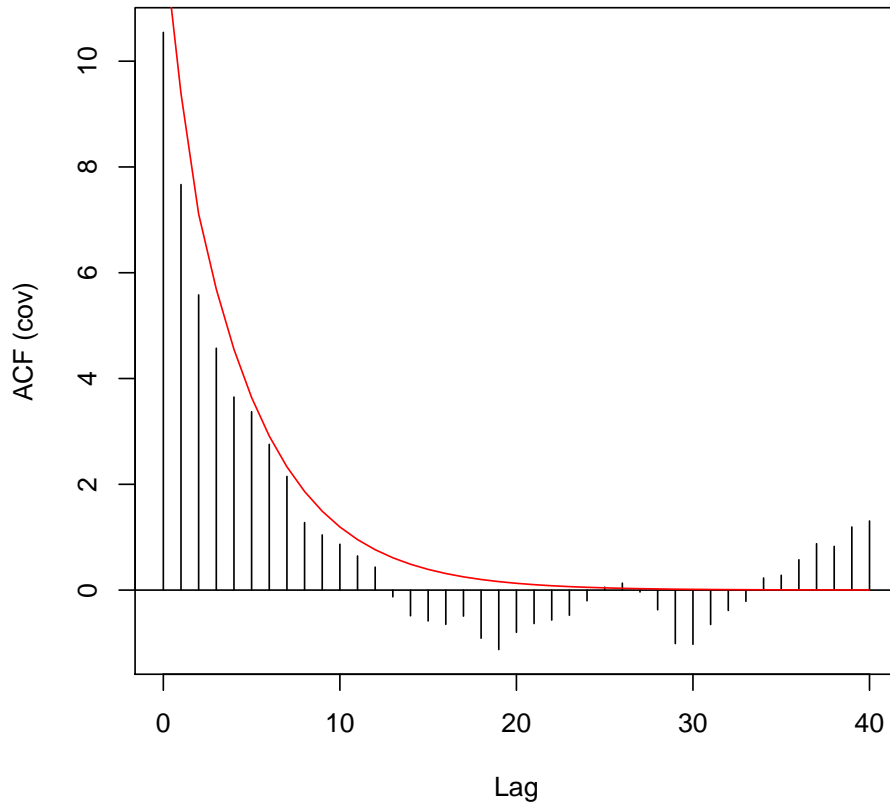


```

> gamma <- rep(0, 41)
> gamma[1] <- sigmaZ^2/(1 - phi^2) + sigmaW^2 * (1 + theta^2)
> gamma[2] <- sigmaZ^2/(1 - phi^2) * phi + sigmaW^2 * theta
> gamma[3:41] <- sigmaZ^2/(1 - phi^2) * phi^(2:40)
> acf(Y, type = "cov", lag.max = 40)
> lines(0:40, gamma, col = "red")

```

### Series Y



- (b) En utilisant un filtre de Kalman, prédire les valeurs de  $Y_{n+1}$  et  $Y_{n+2}$ , assorties d'intervalles de confiance individuels de coefficient de sécurité 95%.

**Solution.** Il reste à comprendre l'initialisation du filtre. Comme  $U$  et  $W$  sont centrées,  $P_0(\vec{X}_0) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ . Enfin,  $U$  et  $W$  sont décorrélés, ainsi que  $W_t$  et  $W_{t-1}$ . Par conséquent,

$$\text{Cov}(\vec{X}_0 - P_0(\vec{X}_0)) = \begin{pmatrix} \sigma_Z^2/(1 - \phi^2) & 0 & 0 \\ 0 & \sigma_W^2 & 0 \\ 0 & 0 & \sigma_W^2 \end{pmatrix}.$$

```
> kalmanexo1 <- function(phi, theta, sigmaZ, sigmaW, Y) {
+   n <- length(Y)
+   F <- matrix(c(phi, 0, 0, 0, 0, 0, 0, 1, 0), 3, 3)
+   G <- matrix(c(1, 1, theta), 1, 3)
+   Q <- diag(c(sigmaZ^2, sigmaW^2, 0))
+   Cpredetat <- diag(c(sigmaZ^2/(1 - phi^2), sigmaW^2, sigmaW^2))
+   Ppredetat <- matrix(0, 3, 1)
+   inno <- rep(0, n)
+   varpred <- rep(0, n)
+   pred <- rep(0, n)
+   for (i in 1:n) {
+     Ppredobs <- G %*% Ppredetat
+     pred[i] <- Ppredobs
```

```

+       inno[i] <- if (is.na(Y[i]))
+         0
+       else Y[i] - Ppredobs
+       S <- G %*% Cpredetat %*% t(G)
+       varpred[i] <- S
+       gain <- if (is.na(Y[i]))
+         matrix(0, 3, 1)
+       else Cpredetat %*% t(G) %*% solve(S)
+       Pestetat <- Ppredetat + gain %*% inno[i]
+       Cestetat <- Cpredetat - gain %*% S %*% t(gain)
+       Ppredetat <- F %*% Pestetat
+       Cpredetat <- F %*% Cestetat %*% (t(F))
+       Cpredetat[1, 1] <- Cpredetat[1, 1] + 1
+     }
+     attributes(inno) <- attributes(Y)
+     attributes(varpred) <- attributes(Y)
+     return(list(inno = inno, var.pred = varpred, pred = pred))
+   }
> Ybis <- c(Y, NA, NA)
> out <- kalmanexol(phi, theta, sigmaZ, sigmaW, Ybis)
> pred <- out$pred[(n + 1):(n + 2)]
> sd.pred <- sqrt(out$var.pred[(n + 1):(n + 2)])
> c((pred + qnorm(0.025) * sd.pred)[1], (pred + qnorm(0.975) *
+   sd.pred)[1])
[1] 1.262586 5.182514
> c((pred + qnorm(0.025) * sd.pred)[2], (pred + qnorm(0.975) *
+   sd.pred)[2])
[1] 0.0680612 5.0880184

```

**Exercice 2** La série `co2.rda` représente la concentration moyenne mensuelle en CO<sub>2</sub> à Mauna Loa, Hawaï, mesurée (en parties pour million) de janvier 1965 à décembre 1980.

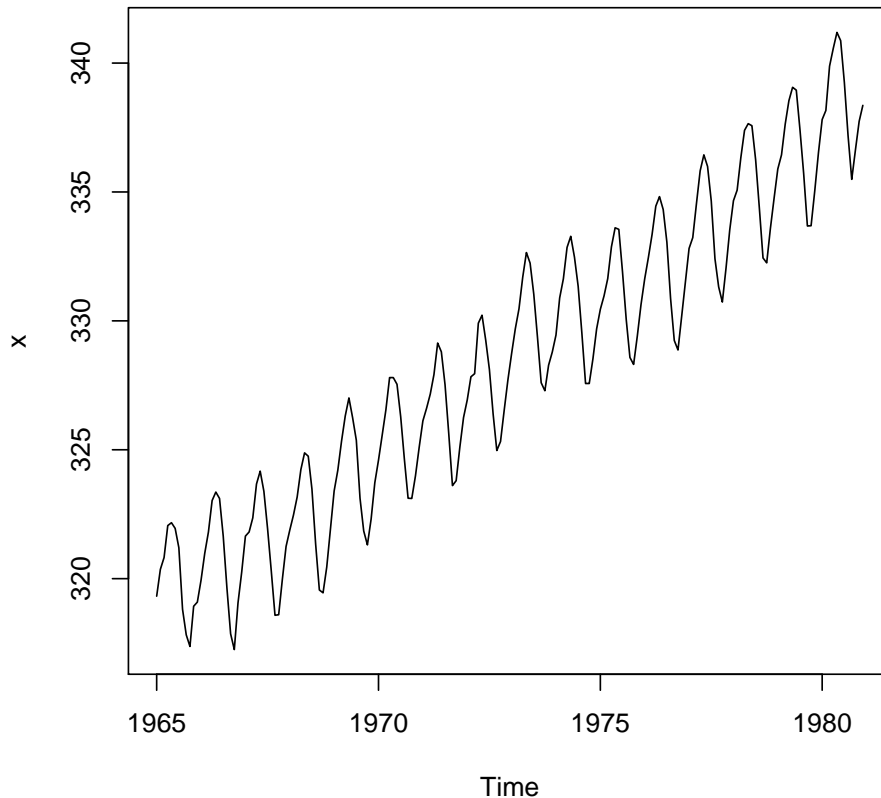
1. Tracer la série et la décrire (tendance, phénomène saisonnier, bruit).

**Solution.**

```

> load("co2.rda")
> x <- co2
> n <- length(x)
> plot(x)

```



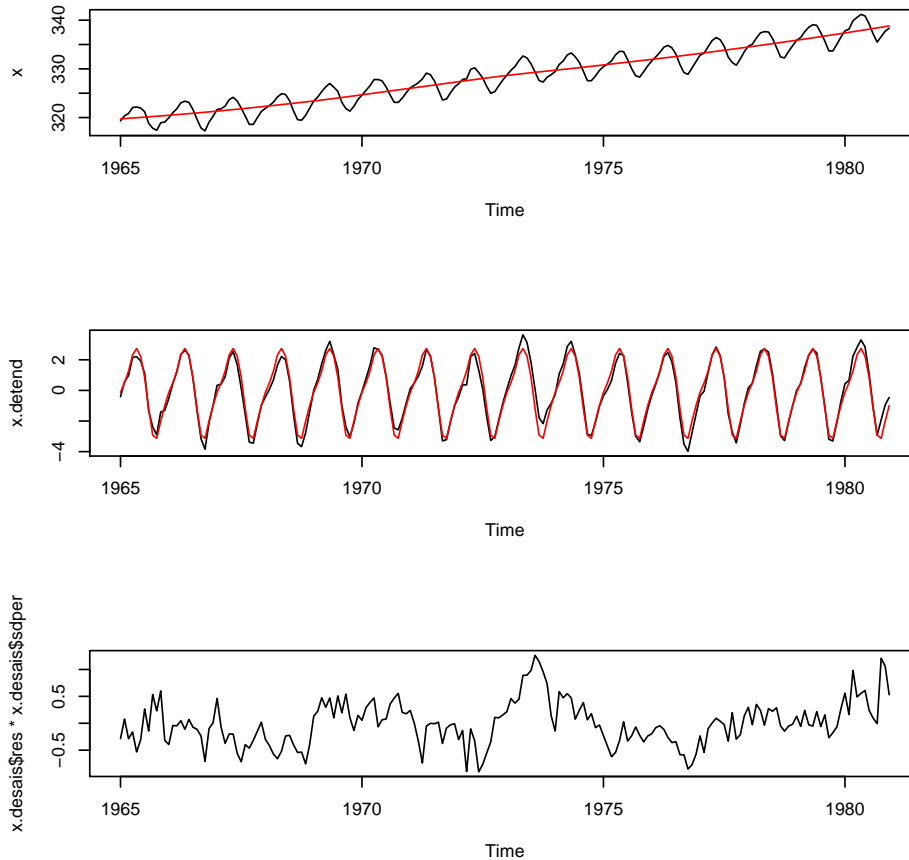
On observe une tendance croissante grossièrement linéaire et une saisonnalité de période 12. Examinons le résultat d'une désaisonnalisation classique pour voir l'allure du bruit restant.

```
> desais <- function(x, per) {
+   m <- rep(0, per)
+   sdx <- rep(0, per)
+   n <- length(x)
+   for (i in 1:per) {
+     m[i] <- mean(x[seq(i, n, per)])
+     sdx[i] <- sd(x[seq(i, n, per)])
+   }
+   sais <- rep(m, length.out = n)
+   attributes(sais) <- attributes(x)
+   sdper <- rep(sdx, length.out = n)
+   attributes(sdper) <- attributes(x)
+   res <- (x - sais)/sdper
+   list(motif = m, sais = sais, res = res, sdper = sdper)
+ }
> x.tend <- (loess(x ~ time(x)))$fitted
> attributes(x.tend) <- attributes(x)
> x.detend <- x - x.tend
> x.desais <- desais(x.detend, 12)
> par(mfrow = c(3, 1))
> plot(x)
```

```

> lines(x.tend, col = "red")
> plot(x.detend)
> lines(x.desais$sais, col = "red")
> plot(x.desais$res)

```



On pourrait envisager un modèle additif avec un bruit *ARMA* (option non envisagée dans la suite de l'énoncé).

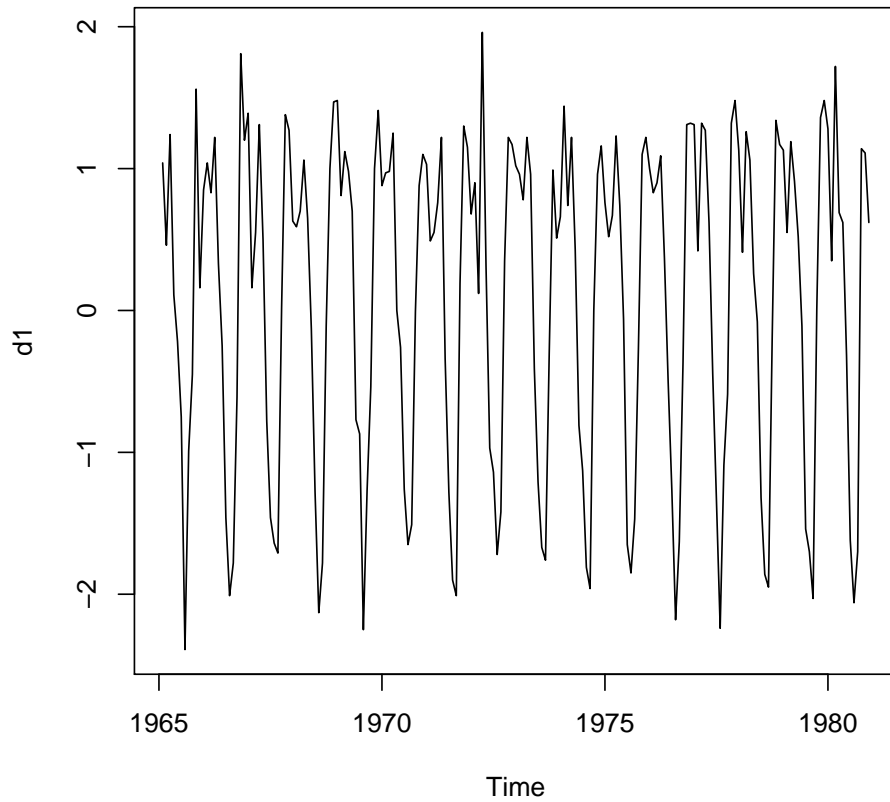
2. Ajuster un *SARIMA* sur cette série.

**Solution.** Il est clair sur la série initiale qu'il faut différentier au moins une fois, soit à l'échelle de l'échantillon, soit à l'échelle de la période. Examinons la série  $(1 - B)X$ .

```

> d1 <- diff(x)
> plot(d1)

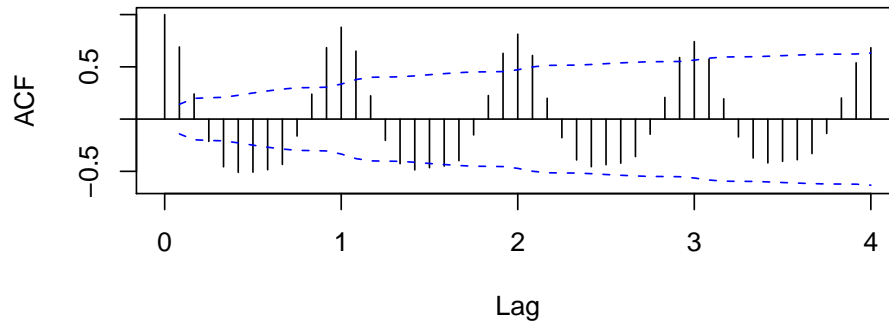
```



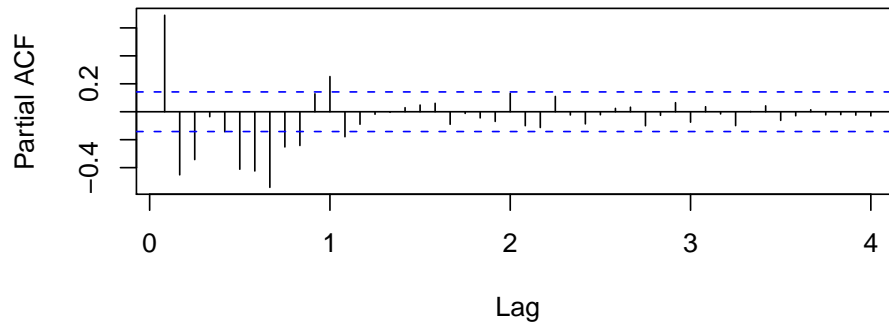
```
> par(mfrow = c(2, 1))  
> acf(d1, lag.max = 48, ci.type = "ma")  
> pacf(d1, lag.max = 48)
```



**Series d1**

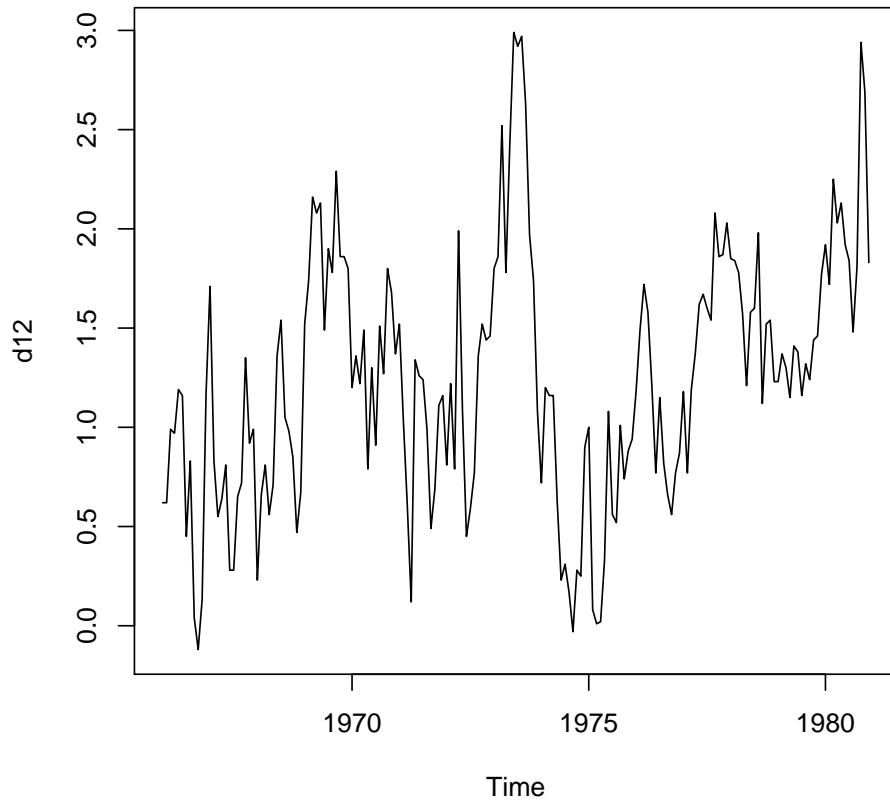


**Series d1**



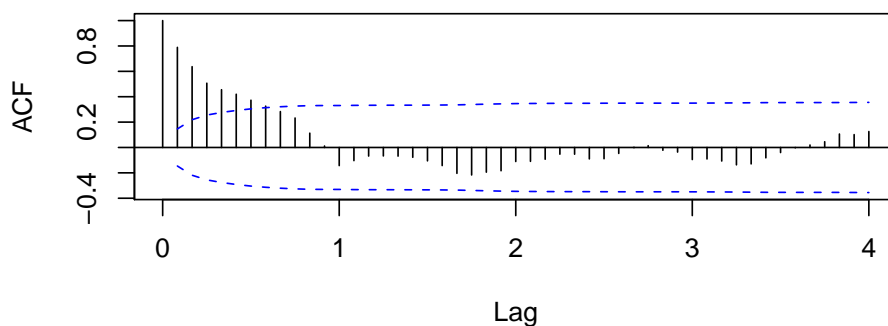
L'ACF et la PACF entre les lags 0 et 6 sont trop laides, on laisse tomber ce modèle.  
Examinons la série  $(1 - B^{12})X$ .

```
> d12 <- diff(x, lag = 12)  
> plot(d12)
```

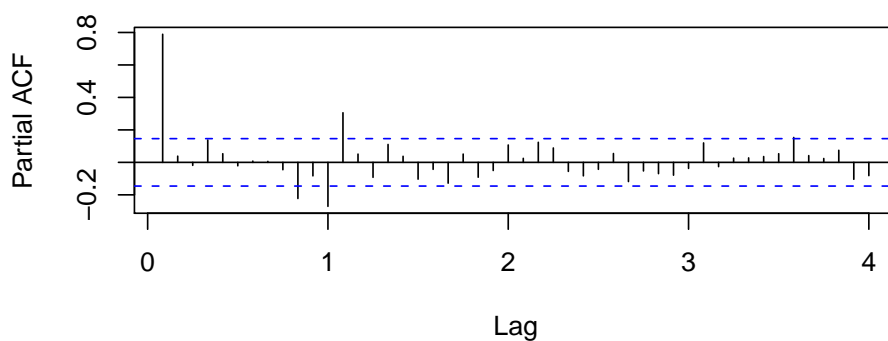


```
> par(mfrow = c(2, 1))  
> acf(d12, lag.max = 48, ci.type = "ma")  
> pacf(d12, lag.max = 48)
```

Series d12



Series d12



```
> out <- arima(x, order = c(1, 0, 0), seasonal = list(order = c(1,
+ 1, 0), period = 12), xreg = (1:n))
> ts.diag <- function(out, lag.max = NULL, lagLB = NULL) {
+   res <- residuals(out)
+   n <- length(res)
+   if (is.null(lag.max))
+     lag.max <- min(max(12, (max(out$arima[3], out$arima[4]) +
+       1) * out$arima[5]), n - 1)
+   if (is.null(lagLB))
+     lagLB <- lag.max
+   fitdf <- sum(out$arima[1:4])
+   par(mfrow = c(3, 1))
+   plot(res)
+   acf(res, lag.max = lag.max)
+   pacf(res, lag.max = lag.max)
+   Box.test(res, lag = lagLB, fitdf = fitdf, type = "Ljung-Box")
+ }
> ts.diag(out, lag.max = 48)
```

Box-Ljung test

data: res

X-squared = 52.0381, df = 46, p-value = 0.2506

```
> out
```

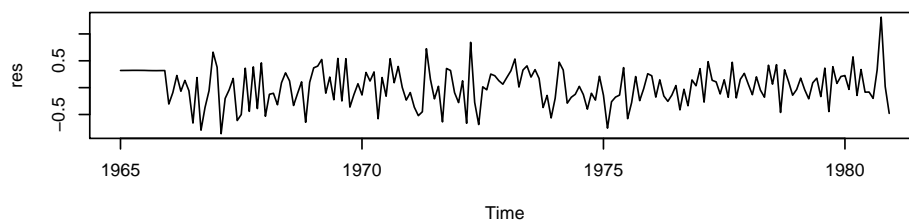
Call:

```
arima(x = x, order = c(1, 0, 0), seasonal = list(order = c(1, 1, 0), period = 12),  
      xreg = (1:n))
```

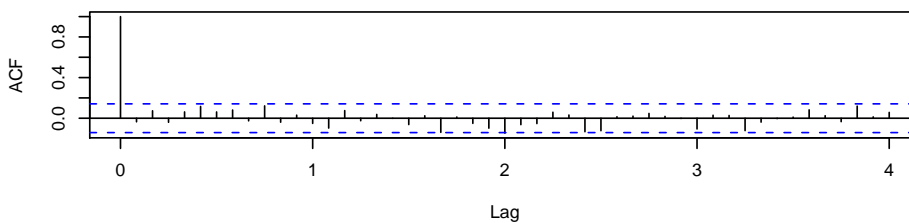
Coefficients:

	ar1	sar1	(1:n)
	0.8546	-0.4993	0.1038
s.e.	0.0393	0.0690	0.0096

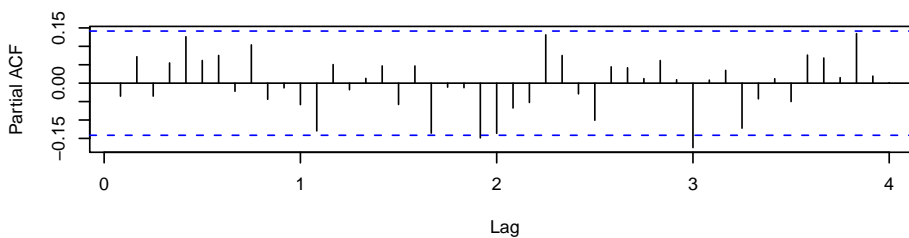
sigma^2 estimated as 0.1154: log likelihood = -63.39, aic = 134.77



Series res

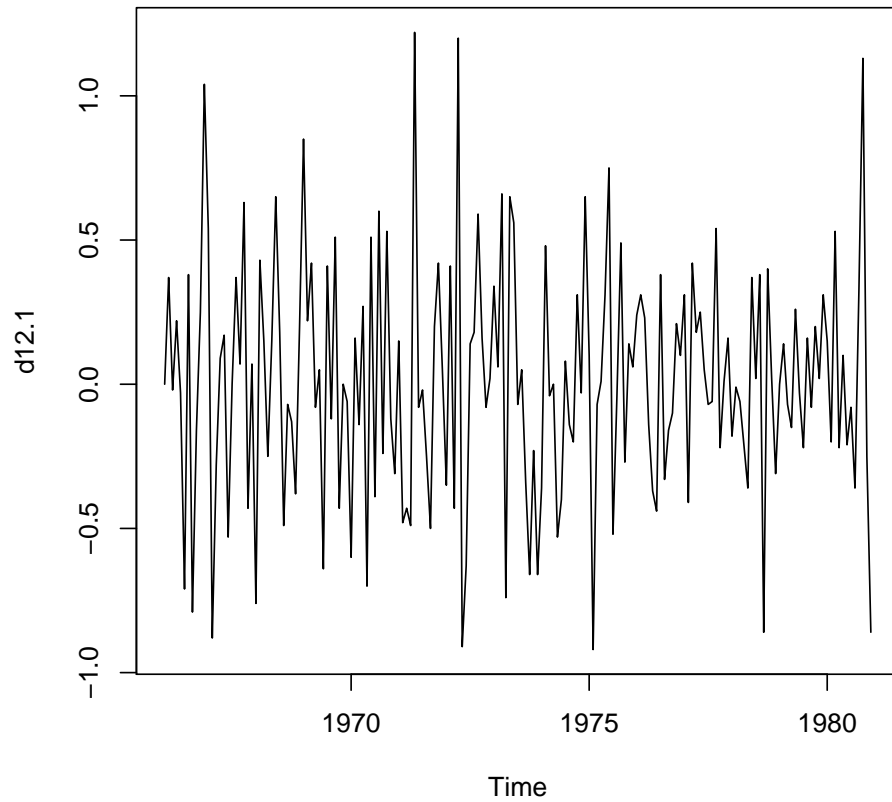


Series res



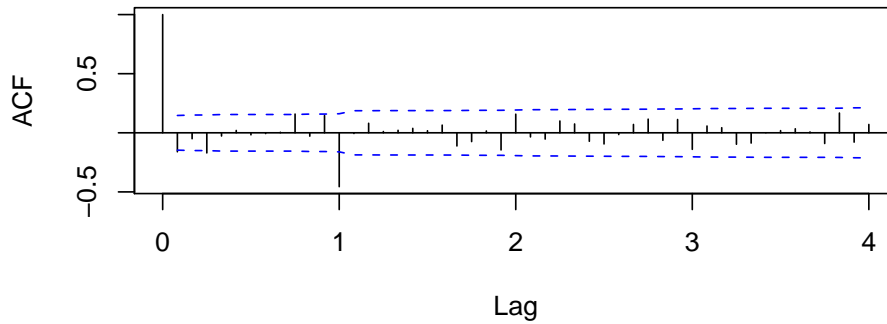
On ne rejette pas ce modèle. On ne peut pas le simplifier, donc on le garde tel quel.  
Enfin, examinons la série  $(1 - B)(1 - B^{12})X$ .

```
> d12.1 <- diff(d12)  
> plot(d12.1)
```

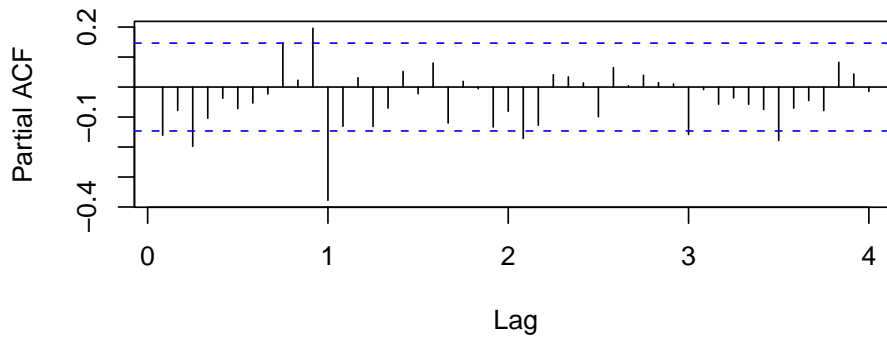


```
> par(mfrow = c(2, 1))  
> acf(d12.1, lag.max = 48, ci.type = "ma")  
> pacf(d12.1, lag.max = 48)
```

### Series d12.1



### Series d12.1



Suggère  $P = 1$  ou  $Q = 1$  et  $p = 3$  ou  $q = 3$

```
> out1 <- arima(x, order = c(3, 1, 0), seasonal = list(order = c(1,
+ 1, 0), period = 12), xreg = (1:n)^2)
> out1
```

Call:

```
arima(x = x, order = c(3, 1, 0), seasonal = list(order = c(1, 1, 0), period = 12),
      xreg = (1:n)^2)
```

Coefficients:

	ar1	ar2	ar3	sar1	(1:n)^2
	-0.1914	-0.0979	-0.1893	-0.5124	2e-04
s.e.	0.0745	0.0757	0.0769	0.0673	5e-04

sigma^2 estimated as 0.1160: log likelihood = -63.12, aic = 138.24

```
> out2 <- arima(x, order = c(0, 1, 3), seasonal = list(order = c(0,
+ 1, 1), period = 12), xreg = (1:n)^2)
> out2
```

Call:

```
arima(x = x, order = c(0, 1, 3), seasonal = list(order = c(0, 1, 1), period = 12),
      xreg = (1:n)^2)
```

Coefficients:

	ma1	ma2	ma3	sma1	(1:n)^2
	-0.3307	-0.0394	-0.1406	-0.8602	2e-04
s.e.	0.0777	0.0819	0.0752	0.0952	1e-04

sigma^2 estimated as 0.09639: log likelihood = -52.77, aic = 117.54

Entre le  $SARIMA(3,1,0) \times (1,1,0)_{12}$  et le  $SARIMA(0,1,3) \times (0,1,1)_{12}$ , on préfère le second qui a un plus petit AIC. De plus, il semble qu'on puisse abandonner le régresseur.

```
> out <- arima(x, order = c(0, 1, 3), seasonal = list(order = c(0,
+ 1, 1), period = 12))
> ts.diag(out, lag.max = 48)
```

Box-Ljung test

data: res

X-squared = 54.3794, df = 44, p-value = 0.1358

```
> out
```

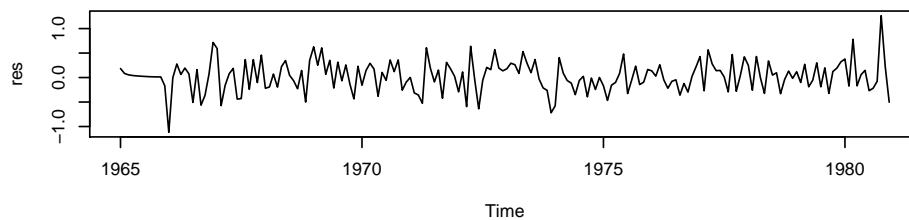
Call:

```
arima(x = x, order = c(0, 1, 3), seasonal = list(order = c(0, 1, 1), period = 12))
```

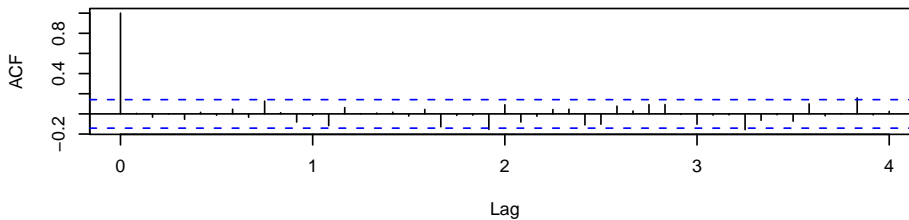
Coefficients:

	ma1	ma2	ma3	sma1
	-0.3126	-0.0351	-0.134	-0.8341
s.e.	0.0764	0.0812	0.075	0.0888

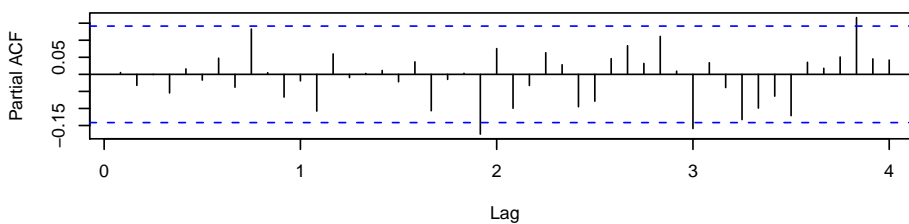
sigma^2 estimated as 0.09835: log likelihood = -53.65, aic = 117.29



Series res



Series res



On conserve ce modèle. Essayons de le simplifier par minimum d'AIC.

```
> armaic <- function(x, M = 0, d = 0, ...) {
+   AIC <- matrix(NA, M + 1, M + 1)
+   colnames(AIC) <- seq(0, M)
+   rownames(AIC) <- seq(0, M)
+   aic <- Inf
+   popt <- 0
+   qopt <- 0
+   for (p in 0:M) {
+     for (q in 0:(M - p)) {
+       outtemp <- arima(x, order = c(p, d, q), optim.control = list(maxit = 600),
+         ...)
+       if (aic > outtemp$aic) {
+         out <- outtemp
+         aic <- outtemp$aic
+         popt <- p
+         qopt <- q
+       }
+       AIC[p + 1, q + 1] <- outtemp$aic
+     }
+   }
+   res <- list(model = out, AIC = AIC, popt = popt, qopt = qopt)
+ }
> out.armaic <- armaic(x, M = 3, d = 1, seasonal = list(order = c(0,
```



```

+      0, 1), period = 12))
> out.armaic

$model

Call:
arima(x = x, order = c(p, d, q), seasonal = ..1, optim.control = list(maxit = 600))

Coefficients:
          ar1          ar2          ma1          sma1
      1.4499   -0.7392   -0.8460    0.4149
s.e.  0.0514    0.0499    0.0319    0.0543

sigma^2 estimated as 0.3298:  log likelihood = -167.09,  aic = 344.18

$AIC
      0          1          2          3
0 460.8149 414.2856 382.2743 379.6187
1 397.9033 396.3119 381.5724      NA
2 390.7097 344.1769      NA      NA
3 364.6553      NA      NA      NA

$popt
[1] 2

$qopt
[1] 1

Examinons de plus près le modèle suggéré :

> out <- arima(x, order = c(2, 1, 1), seasonal = list(order = c(0,
+      1, 1), period = 12))
> ts.diag(out, lag.max = 48)

      Box-Ljung test

data:  res
X-squared = 53.5378, df = 44, p-value = 0.1536

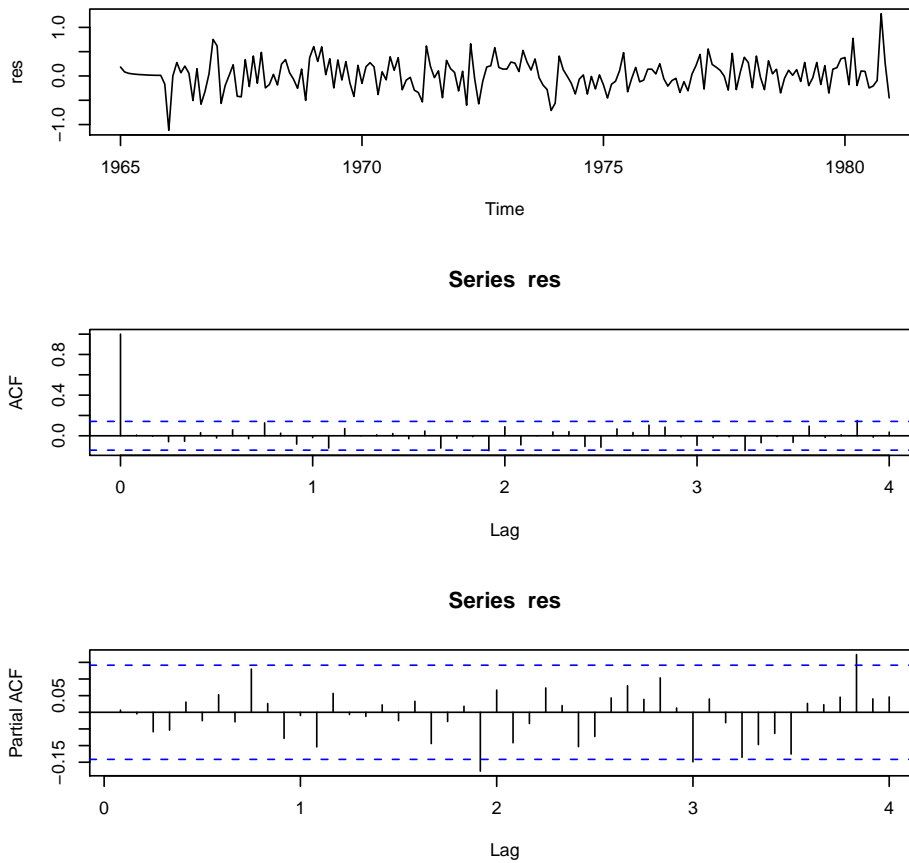
> out

Call:
arima(x = x, order = c(2, 1, 1), seasonal = list(order = c(0, 1, 1), period = 12))

Coefficients:
          ar1          ar2          ma1          sma1
      0.4219    0.0677   -0.7387   -0.8395
s.e.  0.1731    0.1015    0.1531    0.0886

sigma^2 estimated as 0.09884:  log likelihood = -54.26,  aic = 118.53

```

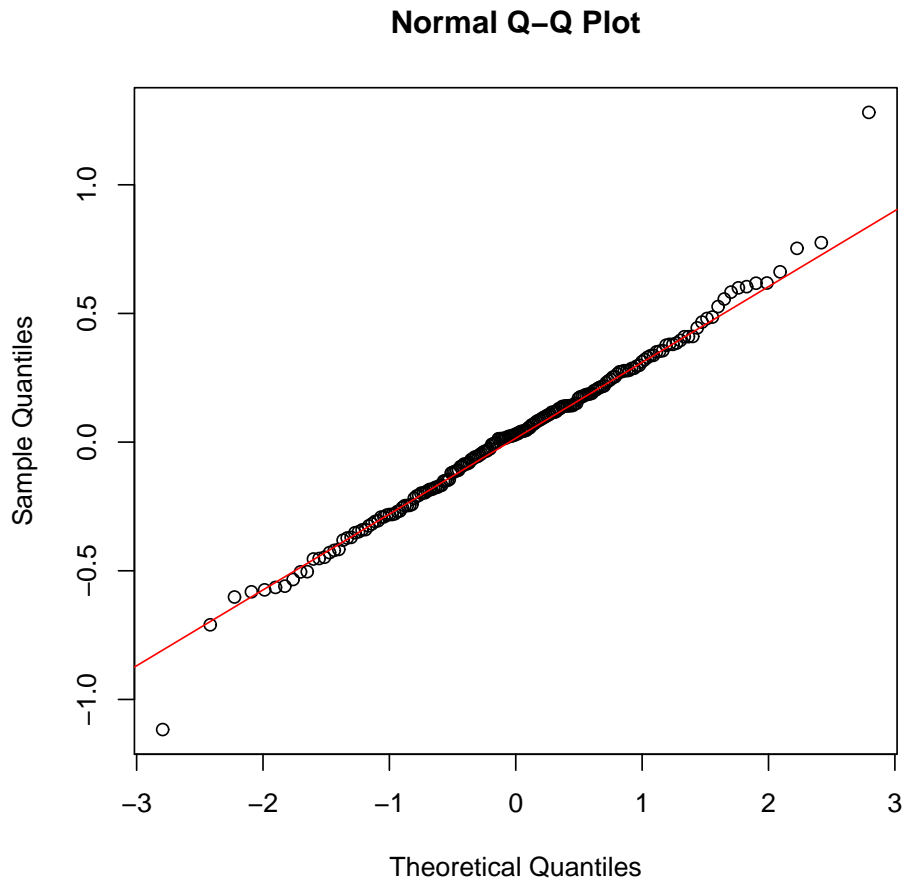


On conserve ce dernier modèle,  $SARIMA(2, 1, 1) \times (0, 1, 1)_{12}$ , l'AIC étant minimal et la stationnarité de la série  $(1 - B)(1 - B^{12})X$  étant plus claire.

3. Prédire la concentration pour l'année 1981, assortie d'intervalles de confiance individuels de coefficient de sécurité 95%.

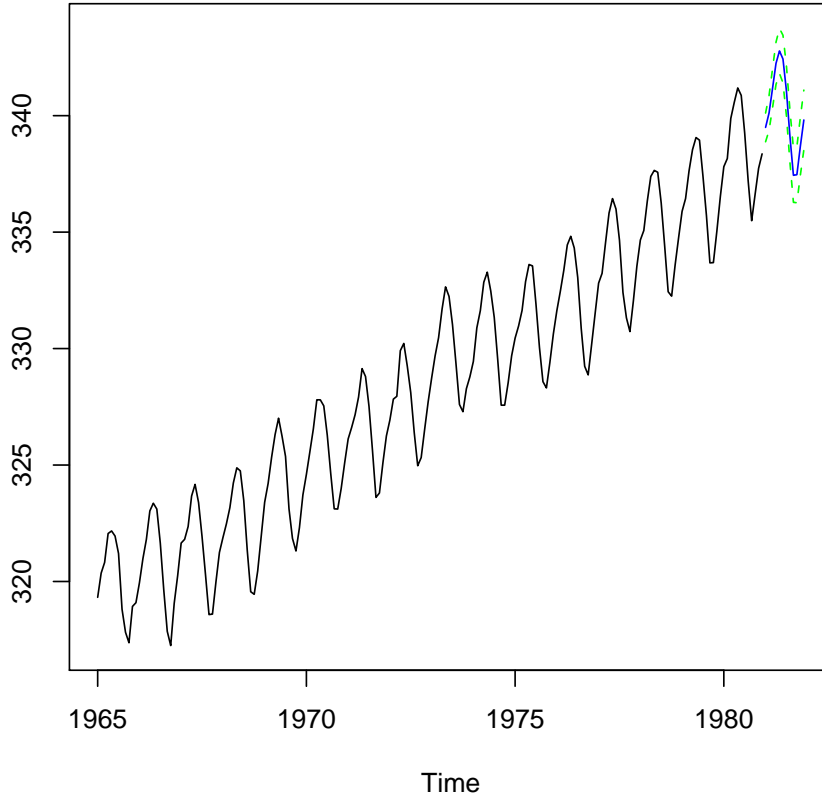
**Solution.** Vérifions tout d'abord le caractère gaussien des résidus.

```
> res <- residuals(out)
> qqnorm(res)
> qqline(res, col = "red")
```



Le résultat est très correct, on conserve l'hypothèse de résidus gaussiens.

```
> outpred <- predict(out, n.ahead = 12)
> pred <- outpred$pred
> predup <- pred + qnorm(0.975) * outpred$se
> predlow <- pred + qnorm(0.025) * outpred$se
> ts.plot(x, pred, predup, predlow, col = c("black", "blue", "green",
+     "green"), lty = c(1, 1, 2, 2))
```



**Exercice 3** On considère un  $ARMA(1, 1)$   $X$  :

$$\forall t \in \mathbb{Z}, X_t = \phi X_{t-1} + Z_t + \theta Z_{t-1},$$

où  $Z$  est un bruit blanc de variance  $\sigma^2$  et  $|\phi| < 1$ . On note  $Y$  la série différenciée :

$$\forall t \in \mathbb{Z}, Y_t = X_t - X_{t-1}.$$

1. Montrer que  $Y$  est un  $ARMA$  piloté par  $Z$  et déterminer ses paramètres.

**Solution.** Tout d'abord,  $Y$  est stationnaire puisque c'est l'image par un filtre linéaire du processus stationnaire  $X$ . Puis, comme  $(1 - \phi B)X = (1 + \theta B)Z$ ,

$$(1 - \phi B)Y = (1 - \phi B)(1 - B)X = (1 - B)(1 - \phi B)X = (1 - B)(1 + \theta B)Z.$$

Donc  $Y$  est un  $ARMA$  piloté par  $Z$  (dont la variance est  $\sigma^2$ ) et :

$$\forall t \in \mathbb{Z}, Y_t = \phi Y_{t-1} + Z_t + (\theta - 1)Z_{t-1} - \theta Z_{t-2}.$$

$Y$  est donc un  $ARMA(1, 2)$ .

2.  $Y$  est-il causal par rapport à  $Z$ ? Est-il inversible par rapport à  $Z$ ?

**Solution.**  $Y$  est causal par rapport à  $Z$  car le polynôme  $z \mapsto (1 - \phi z)$  a ses racines en dehors du cercle unité (puisque  $|\phi| < 1$ ). Par contre, il n'est pas inversible par rapport à  $Z$  puisque le polynôme  $z \mapsto (1 - z)(1 + \theta z)$  a 1 pour racine.

3. On suppose désormais que  $\phi = -1/2$  et  $\theta = 1/4$ . A l'aide d'un ordinateur, calculer la matrice de covariance de  $(Y_1, \dots, Y_4)$ .

**Solution.** Comme  $Y$  est stationnaire, il suffit de connaître sa fonction d'autocovariance aux lags  $0, \dots, 3$ . Pour cela, on peut poser  $U$  l'AR(1) tel que :

$$\forall t \in \mathbb{Z}, U_t = \phi U_{t-1} + Z_t.$$

De sorte que  $Y = (1 - B)(1 + \theta B)U$ , d'où :

$$Y_t = U_t + (\theta - 1)U_{t-1} - \theta U_{t-2}.$$

On a donc :

$$\begin{pmatrix} Y_0 \\ Y_1 \\ Y_2 \\ Y_3 \end{pmatrix} = A \begin{pmatrix} U_{-2} \\ U_{-1} \\ U_0 \\ U_1 \\ U_2 \\ U_3 \end{pmatrix},$$

avec :

$$A = \begin{pmatrix} -\theta & \theta - 1 & 1 & 0 & 0 & 0 \\ 0 & -\theta & \theta - 1 & 1 & 0 & 0 \\ 0 & 0 & -\theta & \theta - 1 & 1 & 0 \\ 0 & 0 & 0 & -\theta & \theta - 1 & 1 \end{pmatrix}.$$

Soient  $\Gamma_U$  la matrice de covariance de  $(U_{-2}U_{-1}U_0U_1U_2U_3)^T$  et  $\Gamma_Y$  celle de  $(Y_0Y_1Y_2Y_3)^T$ . On sait que :

$$\Gamma_U(i, j) = \frac{\sigma^2}{1 - \phi^2} \phi^{|i-j|}.$$

Ceci permet de calculer :

$$\Gamma_Y = A\Gamma_U A^T.$$

```
> sigma2 <- 4
> phi <- -1/2
> theta <- 1/4
> GammaU <- matrix(0, 6, 6)
> for (i in 1:6) {
+   for (j in 1:6) GammaU[i, j] <- sigma2/(1 - phi^2) * phi^(abs(i -
+     j))
+ }
> A <- matrix(0, 4, 6)
> A[upper.tri(A, diag = TRUE)] <- (toeplitz(c(-theta, theta - 1,
+   1, 0, 0, 0))[1:4, ])[upper.tri(A, diag = TRUE)]
> GammaY <- A %%% GammaU %%% t(A)
> GammaY
```

```
      [,1] [,2] [,3] [,4]
[1,] 11.0000 -7.250 2.625 -1.3125
[2,] -7.2500 11.000 -7.250 2.6250
[3,] 2.6250 -7.250 11.000 -7.2500
[4,] -1.3125 2.625 -7.250 11.0000
```

4. A l'aide d'un ordinateur, calculer les prédictions linéaires de  $Y_3$  et  $Y_4$  sachant  $(Y_1, Y_2)$  ainsi que les variances des erreurs de prédiction.

**Solution.**

```

> Gamma <- GammaY[1:2, 1:2]
> g3 <- GammaY[1:2, 3]
> g4 <- GammaY[1:2, 4]
> alpha3 <- solve(Gamma, g3)
> v3 <- GammaY[3, 3] - t(alpha3) %*% g3
> alpha4 <- solve(Gamma, g4)
> v4 <- GammaY[4, 4] - t(alpha4) %*% g4
> alpha3

[1] -0.3461187 -0.8872146

> v3

      [,1]
[1,] 5.476256

> alpha4

[1] 0.06712329 0.28287671

> v4

      [,1]
[1,] 10.34555

```

Donc :

$$P(Y_3|Y_1, Y_2) = -0.3461Y_1 + -0.8872Y_2 ,$$

avec une variance d'erreur de prédiction valant 5.476. Puis :

$$P(Y_4|Y_1, Y_2) = 0.06712Y_1 + 0.2829Y_2 .$$

avec une variance d'erreur de prédiction valant 10.35.