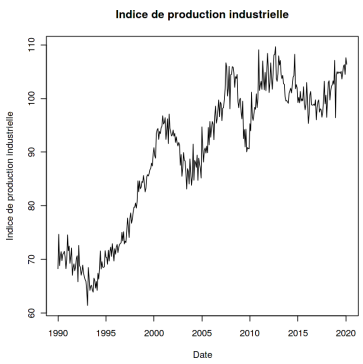
Projet de séries temporelles linéaires

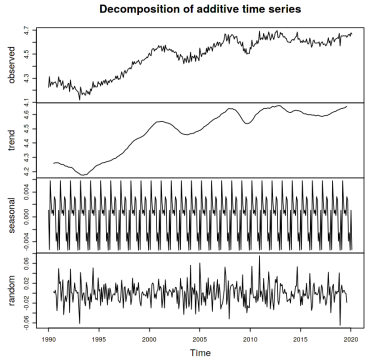
Fadi SAHBANI et Hamdi BEL HADJ HASSINE

Partie I : Les données

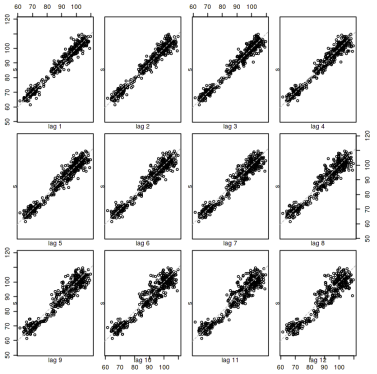
1. Notre série étudiée est l'Indice de la production industrielle lié à la Réparation et installation de machines et d'équipements (NAF rév. 2, niveau division, poste 33) de base 100 en 2015, disponible sur le site de l'INSEE. Ces données sont calculées par l'INSEE à partir des enquêtes mensuelles auprès d'un échantillon d'entreprises.
   Cette série est corrigée des variations saisonnières (CVS) et des jours ouvrés (CJO). L'estimation de ces effets est effectuée avec la méthode X13-Arima.
   La série étudiée comporte 362 observations à une fréquence mensuelle et couvre la période comprise entre janvier 1990 et février 2020.
2. On commence par représenter graphiquement la série sur la figure 1 ci-dessous.



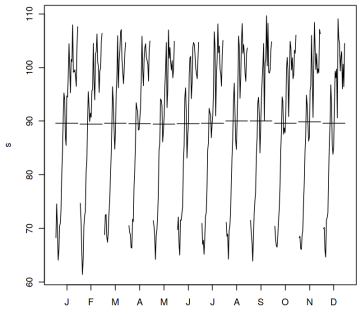
(a) Graphique de la série



(b) Décomposition de la série



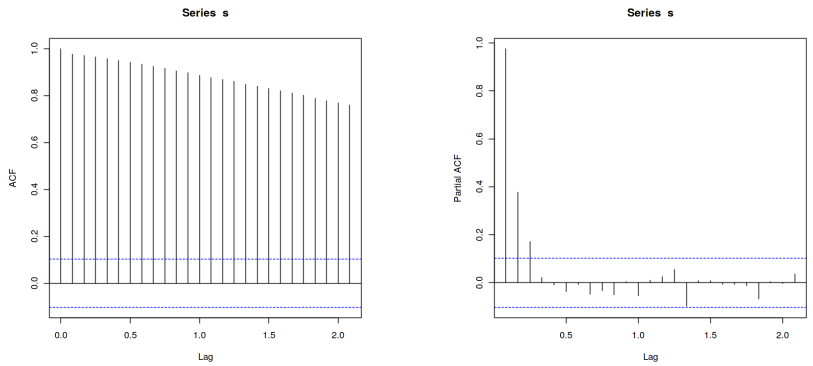
(c) Le lagplot de la série



(d) Le monthplot de la série

FIGURE 1 – Visualisation de la série

On constate d'abord qu'il semble y avoir une tendance mais pas de saisonnalité ; cela est confirmé par le monthplot puisque les 12 chronogrammes mensuels sont à peu près identiques. La décomposition de la série permet aussi d'identifier une tendance croissante qu'on testera ultérieurement par une régression linéaire.
On examine ensuite la fonction d'auto-corrélation et d'auto-corrélation partielle :



(a) (b)

FIGURE 2 – ACF et PACF de la série

La saisonnalité : On vérifie qu'il n'existe pas de motif répétitif dans la fonction d'auto-corrélation, ce qui indique l'absence de saisonnalité et confirme le monthplot.

La tendance : l'ACF montre aussi que la série n'est pas stationnaire et qu'elle présente une tendance ce qu'on peut aussi voir dans le lagplot. Pour vérifier la présence d'une tendance on fait une régression linéaire sur t. la p-valeur du coefficient de t est de l'ordre de 10^{-16} d'où sa significativité statistique. Dans la suite on effectuera donc les tests de stationnarité tenant compte de la tendance.

La stationnarité : La figure 2 soulève le soupçon que la série n'est pas stationnaire. Pour le vérifier la littérature propose plusieurs tests parmi lesquels nous choisirons deux :
On commence par appliquer un test de Kwiatkowski, Phillips, Schmidt et Shin (ou ***KPSS***). Il s'agit d'un test de stationnarité dans le cas d'une série comportant une tendance, dont l'hypothèse nulle H_0 est que la série avec tendance est stationnaire.
On peut aussi utiliser le Test de ***Dickey-Fuller augmenté*** dont l'hypothèse nulle est l'existence d'une racine unité (impliquant la non-stationnarité de la série). Pour appliquer ce test correctement il faut vérifier sa condition de validité, à savoir la non auto-corrélation des résidus. Pour cela on commence par le test ADF à $k = 0$ retards et on effectue des tests d'autocorrélation de Ljung-Box jusqu'à l'ordre 24 (2 ans) sur les résidus. Tant que l'un des tests de Ljung-Box rejette la nullité des autocorrélations, on augmente k, jusqu'à obtenir des résidus non auto-corrélés. Le test ADF à k retards est alors valide.
Les résultats des deux tests sont regroupés dans les tables 1 et 2 ci-dessous.
Le test KPSS rejette l'hypothèse de stationnarité de la série au seuil de 5%. Et la p-valeur du test ADF vaut 0.564, donc on ne peut pas rejeter l'hypothèse d'existence d'une racine unitaire au seuil 5%. Ces deux tests montrent que notre série est intégrée et qu'elle est au moins I(1).

Table 1 – KPSS

KPSS stat	lag	p-valeur
0.79116	5	<0.01

Table 2 – Augmented Dickey-Fuller Test

DF stat	lag	p-valeur
-2.031	21	0.5638

Pour stationnariser la série on applique une différence première qu'on notera $X_t = \Delta S_t$ avec S_t la série originale. On reconduit ensuite les tests pour vérifier si la série X_t est stationnaire.

Table 3 – KPSS

KPSS stat	lag	p-value
0.040988	5	>0.1

Table 4 – Augmented Dickey-Fuller Test

DF stat	lag	p-value
-16.1929	2	0.01

On voit que la p-valeur du test KPSS est supérieure à 0.1 ce qui implique le non rejet de la stationnarité au seuil de 5% et que la p-valeur de l'ADF est de 0.01 ce qui permet de rejeter l'hypothèse nulle de l'existence d'une racine unitaire au seuil 5%. Les deux tests suggèrent donc que la série différenciée est stationnaire.

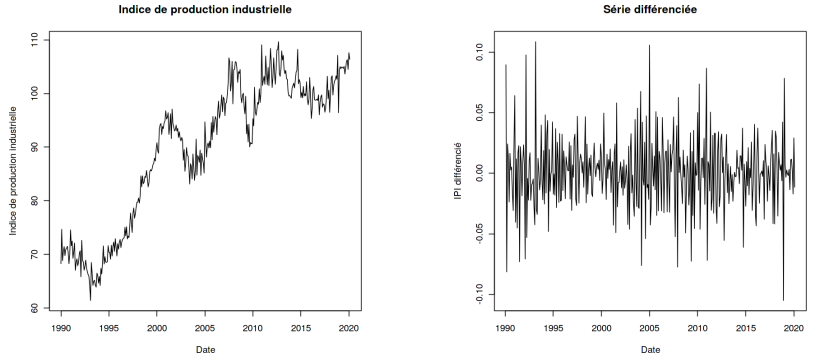
On teste maintenant la présence d'une tendance : on fait une régression linéaire sur t. la p-valeur du coefficient de t est de l'ordre de 0.81 ce qui confirme que cette série ne comporte pas de tendance.

Test d'hétéroscédasticité : La série différenciée (fig. 3) semble stationnaire, à part une légère augmentation de la volatilité entre 2005 et 2010. Il faudrait donc effectuer un test d'hétéroscédasticité pour vérifier si une transformation de type Box-Cox est nécessaire pour réduire l'hétéroscédasticité.

On a appliqué un test Breusch-Pagan dont l'hypothèse nulle est l'homoscédasticité de la série différenciée. Le test ne rejette pas l'hypothèse nulle à l'ordre 5%, donc aucune transformation n'est nécessaire.

On conclut que seule la différentiation de la série S_t est nécessaire pour la rendre stationnaire et qu'elle est I(1).

3. Représentons la série choisie avant et après la transformation. Graphiquement la série différenciée semble stationnaire au second ordre.

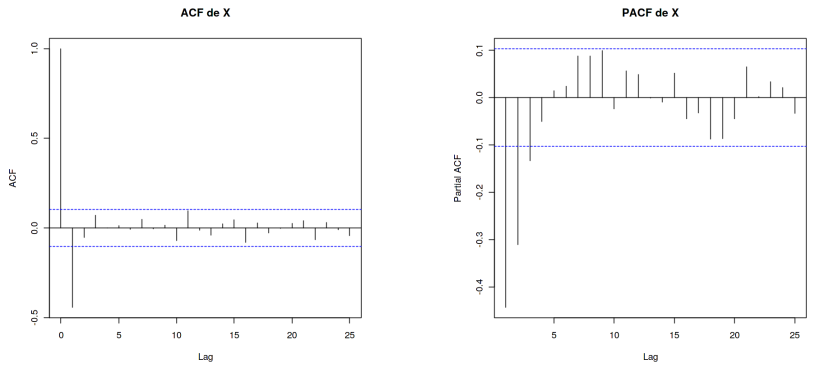


(a) La série avant transformation (b) la série après transformation

Figure 3 – Graphique de la série

Partie II : Modèles ARMA

4. La série différenciée X_t étant stationnaire, on la modélise par un modèle ARMA(p,q). En première étape on trace les autocorrélogrammes de X_t pour déterminer les ordres maximaux p_{max} et q_{max} :



(a) (b)

FIGURE 4 – ACF et PACF de la série différenciée X_t

Les autocorrélations ne sont plus significatives au niveau 5% à partir du premier retard, donc $q_{max} = 1$. De manière similaire les autocorrélations partielles indiquent $p_{max} = 3$. Huit modèles sont donc possibles.

Pour choisir un modèle, on considère d'abord les critères suivants :

1. Critère de validité : On se sert du test Ljung-Box, dont l'hypothèse nulle est la nullité jointe des autocorrélations d'une série jusqu'à un ordre k donné, pour tester la nullité des autocorrélations des résidus du modèle pour tout k jusqu'à 24. On considère le modèle valide si l'hypothèse nulle n'est rejetée au niveau 5% pour aucune valeur de k.

2. Critère d'ajustement : On vérifie que le dernier coefficient AR et le dernier coefficient MA du modèle sont significatifs à l'ordre 5%.

Ces deux critères permettent de restreindre le nombre des modèles valides et ajustés à 3. Afin de s'assurer que les modèles sont bien valides, on considère un troisième critère :

3. Nullité des autocorrélations des résidus : On élimine les modèles dont les autocorrélations des résidus dépassent les bandes de l'intervalle de confiance à 95%.

Ce troisième critère s'avère néanmoins trop restrictif et élimine tous les modèles. Pour qu'il soit plus informatif, on considère les intervalles de confiance à 96.5%. Cela permet effectivement d'éliminer l'un des trois modèles précédemment retenus et de garder uniquement les modèles ARMA(0,1) et ARMA(2,1).

TABLE 5 – Validité

	q = 0	**q = 1**
p = 0	Invalide	Valide
p = 1	Invalide	Valide
p = 2	Invalide	Valide
p = 3	Valide	Valide

TABLE 6 – Ajustement

	q = 0	**q = 1**
p = 0	Invalide	Valide
p = 1	Valide	Invalide
p = 2	Valide	Valide
p = 3	Valide	Invalide

TABLE 7 – Nullité des autocorrélations

	q = 0	**q = 1**
p = 0	Invalide	Valide
p = 1	Invalide	Valide
p = 2	Invalide	Valide
p = 3	Invalide	Invalide

Après ce processus automatisé, une vérification manuelle des coefficients, résidus et autocorrélogrammes des deux modèles retenus (Annexe fig. 6 et 7) confirme que tous les deux sont valides.

Afin de les comparer, plusieurs méthodes sont envisageables :

•**Critères d'information AIC et BIC** (Annexe tab. 8 et 9) : Parmi tous les modèles initiaux, le critère AIC est minimisé par le modèle ARMA(2,1) et le critère BIC est minimisé par le modèle ARMA(0,1), qui correspondent aux deux modèles retenus à l'étape précédente. Cela renforce la validité de notre démarche mais ne permet pas de trancher entre ces deux modèles.

•**Normalité des résidus :** Les tests de normalité Shapiro-Wilk et Jarque-Bera rejettent la normalité des résidus pour les deux modèles au seuil 1%. Le tracé des QQ-plots (Annexe fig. 8) montre que les résidus suivent une loi relativement proche de la loi normale mais présentant plus de valeurs extrêmes.

•**Coefficients de détermination :** On calcule les R^2 ajustés pour tenir compte du nombre de paramètres. On obtient $R^2_{aj} = 0.272$ pour le modèle ARMA(2,1) et $R^2_{aj} = 0.266$ pour ARMA(0,1).

•**Prévisions hors-échantillon** d'horizon 1 (Annexe fig. 9) : On divise la série S en données d'entraînement (340 observations) et de test (22 observations), on calcule les coefficients d'un ARIMA(0,1,1) et un ARIMA(2,1,1) sur les données d'entrainement, puis on prévoit la valeur de la série à chaque date $t \in [341, 362]$ étant donné les valeurs $(S_i)_{1\leq i\leq t-1}$. La comparaison des erreurs quadratiques moyennes donne $\text{MSE}_{ARIMA(2,1,1)} = 126.0 < \text{MSE}_{ARIMA(2,1,1)} = 129.3$.

$\Rightarrow$ Au vu de ces résultats, en particulier les deux derniers critères, il paraît que le modèle **ARMA(2,1)** est le modèle optimal.

5. Le modèle retenu ARMA(2,1) de la série différenciée X_t correspond à un modèle ARIMA(2,1,1) de la série originelle S_t et s'exprime comme suit :

$$X_t = \alpha_0 + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \epsilon_t + \psi_1 \epsilon_{t-1}$$

$$\Rightarrow (1 - \phi_1 B - \phi_2 B^2) X_t = \alpha_0 + (1 + \psi_1 B)\epsilon_t$$

$$\Rightarrow (1 - \phi_1 B - \phi_2 B^2)(1 - B) S_t = \alpha_0 + (1 + \psi_1 B)\epsilon_t$$

avec $\alpha_0 = 0.100$, $\phi_1 = -0.231$, $\phi_2 = -0.159$ et $\psi_1 = -0.380$.

Partie III : Prévision

6. Dans cette partie on suppose la normalité des résidus de la série. On note $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$.

Selon notre modèle ARMA(2,1), X_{T+1} et X_{T+1} s'écrivent :

$X_{T+1} = \alpha_0 + \phi_1 X_T + \phi_2 X_{T-1} + \epsilon_{T+1} + \psi_1 \epsilon_T$

$X_{T+2} = \alpha_0 + \phi_1 X_{T+1} + \phi_2 X_T + \epsilon_{T+2} + \psi_1 \epsilon_{T+1}$

Notons les valeurs à prédire $X^* = \binom{X_{T+1}}{X_{T+2}}$ et les prévisions $\hat{X}^* = \binom{\hat{X}_{T+1}}{\hat{X}_{T+2}}$.

Puisque $\mathbb{E}(\epsilon_{T+1}|X_{1..T}) = \mathbb{E}(\epsilon_{T+1}) = 0$ et $\mathbb{E}(\epsilon_{T+2}|X_{1..T}) = \mathbb{E}(\epsilon_{T+2}) = 0$ alors :

$$\hat{X}^* = \mathbb{E}(X^*|X_{1,..,T}) = \begin{pmatrix} \alpha_0 + \phi_1 X_T + \phi_2 X_{T-1} + \psi_1 \epsilon_T \\ \alpha_0 + \phi_1 \hat{X}_{T+1} + \phi_2 X_T \end{pmatrix} \text{ et } X^* - \hat{X}^* = \begin{pmatrix} \epsilon_{T+1} \\ (\phi_1 + \psi_1)\epsilon_{T+1} + \epsilon_{T+2} \end{pmatrix} = A\epsilon^*$$

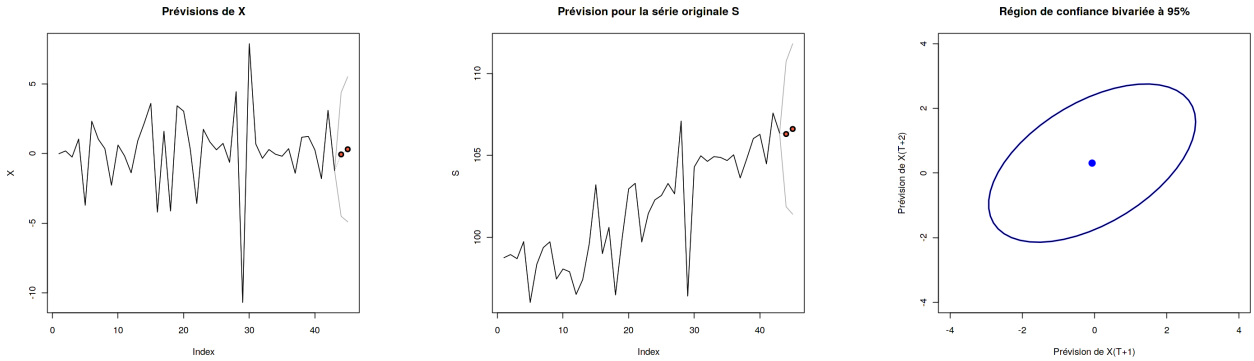
$$\text{avec } A = \begin{pmatrix} 1 & 0 \\ (\phi_1 + \psi_1) & 1 \end{pmatrix} \text{ et } \epsilon^* = \begin{pmatrix} \epsilon_{T+1} \\ \epsilon_{T+2} \end{pmatrix} \sim \mathcal{N}\big(0_2, \left(\begin{smallmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{smallmatrix}\right)\big)$$

On en déduit (en utilisant $\mathbb{V}(A\epsilon^*) = A\mathbb{V}(\epsilon^*)A^T$) :

$$X^* - \hat{X}^* \sim \mathcal{N}\big(0_2, \Sigma\big) \quad \text{avec } \Sigma = \sigma^2 \begin{pmatrix} 1 & \phi_1+\psi_1 \\ \phi_1+\psi_1 & 1+(\phi_1+\psi_1)^2 \end{pmatrix}$$

Par la suite $(X - \hat{X})^T \Sigma^{-1} (X - \hat{X}) \sim \chi^2(2)$ (la matrice Σ étant toujours inversible).
D'où la région de confiance bivariée de niveau 95% ($\alpha = 0.05$) est définie par :
$\text{RC}_\alpha = \left\{ X \in \mathbb{R}^2, (X - \hat{X})^T \Sigma^{-1} (X - \hat{X}) \leq q_{1-\alpha}^{\chi^2(2)} \right\}$ correspondant à une ellipse.
La région de confiance univariée est donnée par :
$\hat{X}_{T+1} - X_{T+1} \sim \mathcal{N}(0, \sigma^2) \Rightarrow \text{IC}_\alpha^{T+1} = [\hat{X}_{T+1} \pm q_{1-\alpha/2}^{\mathcal{N}(0,1)} \sigma]$
$\hat{X}_{T+2} - X_{T+2} \sim \mathcal{N}(0, \sigma^2(1 + (\phi_1 + \psi_1)^2)) \Rightarrow \text{IC}_\alpha^{T+2} = [\hat{X}_{T+2} \pm q_{1-\alpha/2}^{\mathcal{N}(0,1)} \sigma \sqrt{1 + (\phi_1 + \psi_1)^2}]$

7. Pour que la région de confiance calculée ci-dessus soit correcte plusieurs hypothèses sont nécessaires : La spécification du modèle est correcte ($X_t = \alpha_0 + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \epsilon_t + \psi_1 \epsilon_{t-1} \forall t$), les paramètres sont correctement estimés, les résidus sont gaussiens ($\epsilon_t \sim \mathcal{N}(0, \sigma^2)$) et $\sigma^2 > 0$ est connue. En effet, pour pouvoir prédire les valeurs X_{t+1} et X_{t+2} il faut que ϵ soit l'innovation linéaire d'un ARMA ne présentant pas de racines communes ou unitaires (Annexe fig. 10).
   En réalité l'estimation des paramètres et de la variance inconnue des résidus est une source d'incertitude et rend la région de confiance moins précise. L'hypothèse de la normalité des résidus reste aussi douteuse parce que les tests de Shapiro-Wilk et Jarque-Bera dans la deuxième partie la rejettent.
8. Les valeurs prédites par notre modèle de X_{t+1} et X_{t+2} sont respectivement -0.0615 et 0.309 qui correspondent aux deux dates de mars et avril 2020. La figure 5 montre les régions de confiance univariée et bivariée de ces deux prédictions. On constate qu'elles sont relativement larges.



(a) Prévision de (X_{T+1}, X_{T+2}) avec ARMA(2,1) et $IC_{5\%}$ univarié

(b) Prévision de (S_{T+1}, S_{T+2}) avec ARIMA(2,1,1) et $IC_{5\%}$ univarié

(c) Région de confiance bivariée à 95% de (X_{T+1}, X_{T+2})

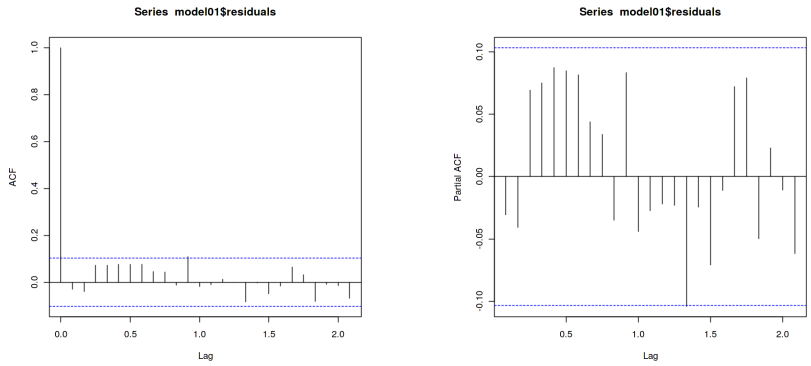
FIGURE 5 – Régions de confiance

9. La connsaissance de Y_{T+1} peut améliorer la prédiction de X_{T+1} si Y_t cause instantanément X_t au sens de Granger, c'est-à-dire :

$$\hat{X}_{T+1|\{X_u, Y_u, u \leq t\} \cup \{Y_{T+1}\}} \neq \hat{X}_{T+1|\{X_u, Y_u, u \leq t\}}$$

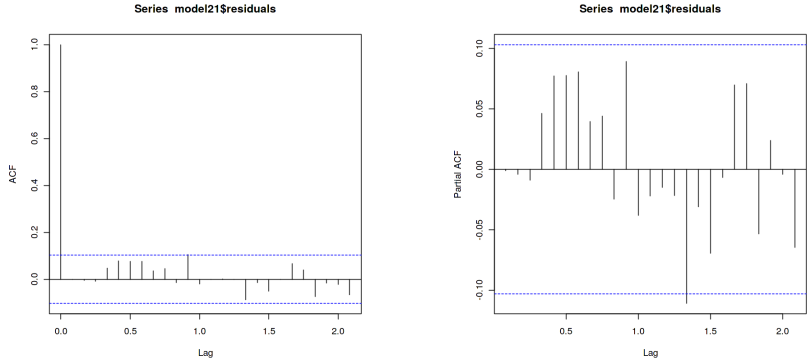
Cette condition est caractérisée par la corrélation entre les deux résidus d'un modèle VAR de (X,Y), et est testable par un test de Wald.

Annexes



(a) (b)

FIGURE 6 – ACF et PACF des résidus du modèle ARMA(0,1)



(a) (b)

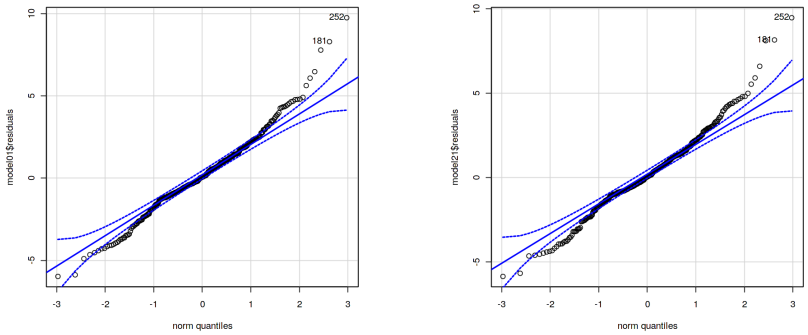
FIGURE 7 – ACF et PACF des résidus du modèle ARMA(2,1)

TABLE 8 – AIC

	q = 0	**q = 1**
p = 0	1732.262	1622.174
p = 1	1658.130	1623.255
p = 2	1626.593	1621.269
p = 3	1621.454	1622.671

TABLE 9 – BIC

	q = 0	**q = 1**
p = 0	1740.040	1633.841
p = 1	1669.797	1638.810
p = 2	1642.148	1640.714
p = 3	1640.899	1646.005



(a) ARIMA(0,0,1) (b) ARIMA(2,0,1)

FIGURE 8 – QQ-plot des résidus des modèles ARMA(2,1) et ARMA(0,1)

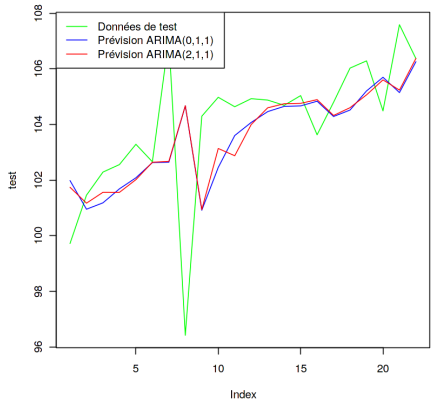


FIGURE 9 – Prévision en utilisant ARIMA(0,1,1) et ARIMA(2,1,1)

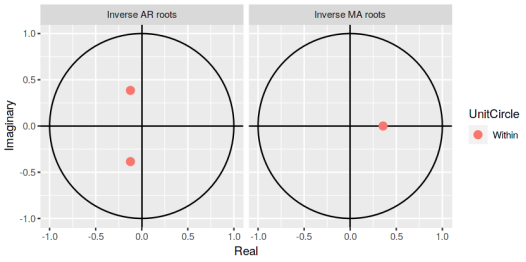


FIGURE 10 – Les racines d'ARMA(2,1)

Script R

```r

# Packages nécessaires:
library(tseries)
library(forecast)
library(fUnitRoots)
library(car)


#####--   Partie 1 : Les données   --#####

### Question 1 ###
path <- dirname(rstudioapi::getSourceEditorContext()$path)
setwd(path)

# Extraction et mise en forme des données :
data <- as.data.frame(read.csv("valeurs_mensuelles.csv", fileEncoding = "UTF-8", sep = ";"))
donnees <- as.matrix(data[-c(1, 2),c(1,2)])
colnames(donnees) <- c("Date","Valeurs")
donnees <- apply(donnees, 2, rev)
rownames(donnees) <- 1:dim(donnees)[1]
head(donnees)
s <- ts(as.numeric(donnees[,2]),start=1990,frequency=12)
n <- length(s)
plot(s,xlab='Date',ylab="Indice de production industrielle", main="Indice de production industrielle")

### Question 2 ###
monthplot(s)
#les 12 chronogrammes mensuels sont à peu près identiques, ce qui confirme l'absence de saisonalité

lag.plot(s,lags=12,layout=c(3,4),do.lines=FALSE)
# Le Lagplot montre une corrélation forte.

fit1 <- decompose(s)
plot(fit1)
#on peut voir que l'erreur ne semble pas trop varier au cours du temps ce qui indique l'adéquation du modele additif
#Visualisation ACF et PACF
acf(s)
pacf(s)
# L'ACF montre que la série n'est pas stationnaire et qu'elle présente une tendance, mais ne montre pas de saisonnalité.

#Test de la tendance
summary(lm(s ~ seq(1,n)))
# Le coefficient de la tendance linéaire est significatif, donc on effectue les tests de stationnarité avec tendance

#Test KPSS de la stationnarité
kpss.test(s,null="Trend")
```

```r
# Le test KPSS rejette au niveau 1% l'hypothèse de stationnarité de la série.

#Test ADF de la stationnarité
#   Tests de LjungBox combinés pour verifier l'autorrélation des résidus
#   jusqu'à l'ordre k
LjungBoxtest <- function(X, k, fitdf=0) {
  pvalues <- apply(matrix(1:k), 1, FUN=function(l) {
    pval <- if (l<=fitdf) NA else Box.test(X, lag=l, type="Ljung-Box", fitdf=fitdf)$p.
    value
    return(c("lag"=l,"pval"=pval))
  })
  return(t(pvalues))
}
#Fonction pour effectuer un test Dickey-Fuller augmenté valide
ValidADF <- function(X,kmax,type){ # Tests ADF jusqu'à avoir des résidus non autocorrélés
  k <- 0
  pasautcor <- 0
  while (pasautcor==0){
    cat(paste0("ADF avec ",k, " lags: "))
    adf <- adfTest(X,lags=k,type=type)
    pvalues <- LjungBoxtest(adf@test$lm$residuals,24,fitdf=length(adf@test$lm$
    coefficients))[,2]
    if (sum(pvalues<0.05,na.rm=T) == 0) {
      pasautcor <- 1;          pasautcor <- 1; cat("Les résidus ne sont pas auto-corrélés.
     Test ADF valide.\n")}
    else cat("Les résidus sont auto-corrélés \n")
    k <- k + 1
  }
  return(adf)
}
adf <- ValidADF(s,24,"ct")
adf
# Le test ADF ne rejette pas au seuil 5% la présence d'une racine unitaire.
# La serie est donc intégrée. On la différencie à l'ordre 1 :
x <- diff(s)
plot(x,xlab='Date',ylab="IPI différencié", main="Série différenciée")

# La série différenciée ne semble pas présenter de constante ou tendance.
#On le vérifie avec une régression linéaire :
summary(lm(x ~ seq(1,length(x))))
# Les coefficients associés à la constante et à la tendance sont non
# significatifs au seuil 5%

#On fait un test KPSS:
kpss.test(x,null="Level")
# Le test KPSS ne rejette pas la stationnarité de la série différenciée

# et maintenant on verifie avec le test ADF
suppressWarnings(
  adf <- ValidADF(x,50,"nc"))
adf
# Le test ADF rejette au seuil 1% la présence d'une racine unitaire.

# On peut aussi le vérifier par la fonction lagplot
lag.plot(x,lags=12,layout=c(3,4),do.lines=FALSE)
```

```r
# La serie différenciée est donc stationnaire,
#c'est-à-dire que la série originale s est I(1)

#Finalement on effectue un test pour vérifier l'homoscédasticité

# Test d'homoscédasticité de Breusch-Pagan
lmtest::bptest(lm(x ~ seq(1,length(x))))
# On ne rejette pas l'homoscédasticité de X au niveau 5%
### Question 3 ###

plot(s,xlab='Date',ylab="Indice de production industrielle", main="Indice de production industrielle")
plot(x,xlab='Date',ylab="IPI différencié", main="Série différenciée")

#####--   Partie 2 : Modèles ARMA   --#####

### Question 4 ###

acf(as.numeric(x), main="ACF de X")
# L'ACF est seulement significatif à l'ordre 1. On note un léger dépassement au lag 11 mais on suppose qu'il est dû au hasard
# On a donc q_max=1
pacf(as.numeric(x),main="PACF de X")
# Le PACF est significatif jusqu'à l'ordre 3, donc p_max=3

# On vérifie ensuite la validité de tous les modèles
# Tests de validité des modèles :
pmax <- 3
qmax <- 1
valide <- matrix(nrow=pmax+1,ncol=qmax+1)
for (p in 0:pmax){
  for (q in 0:qmax){
    model <- arima(x, order = c(p,0,q))
    valide[p+1,q+1] <- all(LjungBoxtest(model$residuals,24,fitdf=p+q)[-c(1:(p+q)),2]>0.05)
  }
}
rownames(valide) <- paste("p=",0:pmax)
colnames(valide) <- paste("q=",0:qmax)

cat("Validité des modèles :")
valide


# Tests de sigificativité des coefficients :
significatif <- matrix(nrow=pmax+1,ncol=qmax+1)
for (p in 0:pmax){
  for (q in 0:qmax){
    model <- arima(x, order = c(p,0,q))
    df <- length(x)-p-q-1
    coef <- model$coef
    se <- sqrt(diag(model$var.coef))
    t <- coef/se
    pval <- ((1-pt(abs(t),df))*2)
    if (p==0 & q==0) {significatif[p+1,q+1] <- (pval[1] < 0.05)}
```

```r
    else if (q==0 | p==0) {significatif[p+1,q+1] <- (pval[p+q] < 0.05)}
    else {significatif[p+1,q+1] <- ( (pval[p] < 0.05) & (pval[q] < 0.05)) }
  }
}
rownames(significatif) <- paste("p=",0:pmax)
colnames(significatif) <- paste("q=",0:qmax)

cat("Significativité des modèles :")
significatif

# Tests de nullité des autocorrélations des résidus :
autocorr_res <- matrix(nrow=pmax+1,ncol=qmax+1)
for (p in 0:pmax){
  for (q in 0:qmax){
    model <- arima(x, order = c(p,0,q))
    acf_res <- acf(model$residuals, plot=FALSE)
    pacf_res <- pacf(model$residuals, plot=FALSE)
    autocorr_res[p+1,q+1] <- all(abs(acf_res$acf)[-1] < qnorm((1 + 0.95)/2)/sqrt(n-1)) &
    all(abs(pacf_res$acf)[-1] < qnorm((1 + 0.95)/2)/sqrt(n-1))
  }
}
rownames(autocorr_res) <- paste("p=",0:pmax)
colnames(autocorr_res) <- paste("q=",0:qmax)

cat("Tests de nullité des autocorrélations des résidus :")
autocorr_res

# Ce dernier critère étant restrictif, il n'a validé aucun modèle
# Alors on augmente l'intervalle de confiance à 96.5% au lieu de 95% pour tester la
    nullité des autocorrélations des résidus
autocorr_res <- matrix(nrow=pmax+1,ncol=qmax+1)
for (p in 0:pmax){
  for (q in 0:qmax){
    model <- arima(x, order = c(p,0,q))
    acf_res <- acf(model$residuals, plot=FALSE)
    pacf_res <- pacf(model$residuals, plot=FALSE)
    autocorr_res[p+1,q+1] <- all(abs(acf_res$acf)[-1] < qnorm((1 + 0.965)/2)/sqrt(n-1)) &
     all(abs(pacf_res$acf)[-1] < qnorm((1 + 0.965)/2)/sqrt(n-1))
  }
}
rownames(autocorr_res) <- paste("p=",0:pmax)
colnames(autocorr_res) <- paste("q=",0:qmax)

cat("Tests de nullité des autocorrélations des résidus :")
autocorr_res

#Le test a séléctionné alors 3 modèles

# On résume les résultats de tous les tests de validité et de significativité :
cat("Tous les tests :")
valide & significatif & autocorr_res

# Cela a permis de sélectionner deux modèles valides et ajustés : Les modèles ARMA(0,0,1)
    et ARIMA(2,0,1)
# Vérifions ces modèles
```

```
202
203 model01 <- arima(x, order=c(0,0,1))
204 LjungBoxtest(model01$residuals,24,fitdf=1)
205 df <- length(x)-1
206 coef <- model01$coef
207 se <- sqrt(diag(model01$var.coef))
208 t <- coef/se
209 pval <- ((1-pt(abs(t),df))*2)
210 cat("Coefficients : \n")
211 coef
212 cat("P-valeurs : \n")
213 pval
214 plot(model01$residuals)
215 acf(model01$residuals)
216 pacf(model01$residuals)
217 # On constate un très léger dépassement sur les autocorrélogrammes, c'est pourquoi ce
    modèle ne passait pas le test à 95% mais passait le test à 96.5%.
218 #Autrement, modèle est effectivement valide et bien ajusté et les résidus ressemblent à
    un bruit blanc.
219
220 # On vérifie le modèle ARMA(2,1) :
221 model21 <- arima(x, order=c(2,0,1))
222 LjungBoxtest(model21$residuals,24,fitdf=3)
223 df <- length(x)-3
224 coef <- model21$coef
225 se <- sqrt(diag(model21$var.coef))
226 t <- coef/se
227 pval <- ((1-pt(abs(t),df))*2)
228 cat("Coefficients : \n")
229 coef
230 cat("P-valeurs : \n")
231 pval
232 plot(model21$residuals)
233 acf(model21$residuals)
234 pacf(model21$residuals)
235 # Il y a également un léger dépassement de l'intervalle
236 #de confiance mais sinon le modèle semble valide et bien ajusté.
237
238 # On compare les deux modèles maintenant par les criteres d'information AIC et BIC
239 pmax <- 3
240 qmax <- 1
241 liste_aic <- matrix(nrow=pmax+1,ncol=qmax+1)
242 liste_bic <- matrix(nrow=pmax+1,ncol=qmax+1)
243 for (p in 0:pmax){
244   for (q in 0:qmax){
245     model <- arima(x, order = c(p,0,q))
246     liste_aic[p+1,q+1] <- AIC(model)
247     liste_bic[p+1,q+1] <- AIC(model, k = log(length(x)))
248   }
249 }
250 rownames(liste_aic) <- paste("p=",0:pmax)
251 colnames(liste_aic) <- paste("q=",0:qmax)
252 rownames(liste_bic) <- paste("p=",0:pmax)
253 colnames(liste_bic) <- paste("q=",0:qmax)
254
```

```r
cat("AIC :")
liste_aic  # le meilleur modèle est ARMA(2,1)
cat("BIC :")
liste_bic  # le meilleur modèles est ARMA(0,1)

# Les critères d'information sont divergents, on regarde la normalité des résidus
model01 <- arima(x, order = c(0,0,1))
model21 <- arima(x, order = c(2,0,1))
cat("Modèle ARMA(0,1) :")
#On realise un test de Jarque Bera et de Shapiro
jarque.bera.test(model01$residuals)
shapiro.test(model01$residuals)
# H0: Normalité
# La normalité est rejetée
#QQ-plot:
qqPlot(model01$residuals)

cat("Modèle ARMA(2,1) :")
jarque.bera.test(model21$residuals)
shapiro.test(model21$residuals)
# La normalité est rejetée
#QQ-plot:
qqPlot(model21$residuals)

# Les 2 modèles donnent des résultats proches


# On s'intéresse aux R2 ajustés des deux modèles :
adjr2 <- function(model){
  ss_res <- sum(model$residuals^2)
  p <- length(model$model$phi)
  q <- length(model$model$theta[model$model$theta!=0])
  ss_tot <- sum((x - mean(x))^2)
  n <- length(x)
  adj_r2 <- 1-(ss_res/(n-p-q-1))/(ss_tot/(n-1)) #r2 ajusté
  return(adj_r2)
}
cat("Modèle ARMA(0,1) : R2 = ",adjr2(model01),"\n")
cat("Modèle ARMA(2,1) : R2 = ",adjr2(model21))

# Le modèle ARMA(2,1) a un meilleur R2 ajusté

# Prévision sur un échantillon de test
n <- length(s)
train <- s[1:340]
test <- s[341:n]
model01 <- arima(train, order=c(0,1,1))
model21 <- arima(train, order=c(2,1,1))
testfit01 <- Arima(c(train,test), model=model01)
forecast01 <- fitted(testfit01)[341:n]
testfit21 <- Arima(c(train,test), model=model21)
forecast21 <- fitted(testfit21)[341:n]
cat("Modèle ARMA(0,1) : RMSE = ",sum((forecast01-test)^2),"\n")
cat("Modèle ARMA(2,1) : RMSE = ",sum((forecast21-test)^2))

```

```r
plot(test, col="green",type="l")
lines(forecast01, col="blue")
lines(forecast21, col="red")
legend("topleft", legend = c("Données de test", "Prévision ARIMA(0,1,1)", "Prévision ARIMA(2,1,1)"),
       col = c("green", "blue","red"), lty = 1)

# Le modèle ARMA(2,1) donne une erreur inférieure au modèle ARMA(0,1) sur l'échantillon de test

#Chacun des deux modèles minimise un des deux criteres AIC et BIC, mais les critères des prevision
#et R2 ajusté ont montré que ARMA(2,1) est légèrement meilleur. On gardera donc ce modèle.

# Visualisation du modèle ARMA retenu
model_x <- Arima(x,order=c(2,0,1),include.constant=TRUE)
cat("Coefficients : \n")
model_x$coef # Coefficients du modèle
plot(model_x$x,col="red",type="l", main="Série différenciée et modèle ajusté", ylab="X")
lines(fitted(model_x),col="blue")
legend("topleft", legend = c("Série différenciée", "Modèle"),
       col = c("red", "blue"), lty = 1)
# On remarque que le modèle n'explique pas une grande partie de la variance (ce qui était prévisible par le R2 ~ 0.27), mais cela reste acceptable pour un modèle ARIMA série économétrique.

# Tracé de la série originale et du modèle
plot(s,col="red", type="l", main="Série originale et modèle ajusté")
model <- Arima(s,order=c(2,1,1), include.constant=TRUE)
lines(fitted(model),col="blue")
legend("topleft", legend = c("Série", "Modèle"),
       col = c("red", "blue"), lty = 1)

#On peut aussi vérifier les racines du modèle choisi :
autoplot(model)
# Elles sont bien à l'intérieur du cercle unité

#####--   Partie 3 : Prévision   --#####
# On suppose dans cette partie que les résidus sont gaussiens

### Question 8 ###

model <- Arima(x,order=c(2,0,1),include.constant=TRUE)

#Extraction des coefs du modèle et de la variance des résius
model$coef
alpha_0 <- as.numeric(model$coef[4])
phi_1 <- as.numeric(model$coef[1])
phi_2 <- as.numeric(model$coef[2])
psi_1 <- as.numeric(model$coef[3])
sigma2 <- as.numeric(model$sigma2)

# Prévisions de X_{T+1} et X_{T+2}
prev_T1 <- alpha_0 +phi_1*x[n-1] +phi_2*x[n-2]+ psi_1*as.numeric(model$residuals[n-1])
```

```
prev_T2 <- alpha_0 +phi_1*prev_T1 +phi_2*x[n-1]
cat("X(T+1) : ",prev_T1,"\n")
cat("X(T+2) : ",prev_T2)


# Intervalle de confiance univarié pour X_{T+1}
borne_sup_T1<- prev_T1 +1.96*sqrt(sigma2)
borne_inf_T1<- prev_T1-1.96*sqrt(sigma2)

# Intervalle de confiance univarié pour X_{T+2}
borne_sup_T2<- prev_T2 +1.96*sqrt(sigma2*(1+(phi_1+psi_1)^2))
borne_inf_T2<- prev_T2 -1.96*sqrt(sigma2*(1+(phi_1+psi_1)^2))
cat("IC(T+1) : [",borne_inf_T1,borne_sup_T1,"] \n")
cat("IC(T+2) : [",borne_inf_T2,borne_sup_T2,"] \n")


#Représentation graphique de l'intervalle de confiance (univarié) de X :
IC_X <- function(){
ts.plot(ts(c(x[320:n-1],borne_sup_T1, borne_sup_T2),start=2016+7/12,end=2020+3/12,
    frequency=12),type="l", col="darkgrey",main="Prévision de X", ylab="X")
lines(ts(c(x[320:n-1],borne_inf_T1, borne_inf_T2),start=2016+7/12,end=2020+3/12,frequency
    =12),type="l", col="darkgrey")
lines(ts(c(x[320:n-1],prev_T1, prev_T2),start=2016+7/12,end=2020+3/12,frequency=12),lty="
    dotted", col="grey")
lines(ts(x[320:n-1],start=2016+7/12,end=2020+1/12,frequency=12),type="l")
points(2020+2/12, prev_T1,bg='tomato2', pch=21, cex=1, lwd=2)
points(2020+3/12, prev_T2,bg='tomato2', pch=21, cex=1, lwd=2)
}
IC_X()

# On compare avec l'IC généré automatiquement par la librairie
autoplot(forecast(model,h=2),xlim=c(2015,2020.5))
cat("IC(T+1) : [",forecast(model,h=1)$lower[2], forecast(model,h=1)$upper[2],"] \n")
cat("IC(T+2) : [",forecast(model,h=2)$lower[4], forecast(model,h=2)$upper[4],"] \n")
# Résultats semblables à nos prévisions théoriques

#Représentation graphique de l'intervalle de confiance pour S :
IC_S <- function(){
prev_s_T1 <- s[n]+prev_T1
prev_s_T2 <- prev_s_T1+prev_T2
plot(c(s[320:n],borne_sup_T1+s[n], borne_sup_T2+prev_s_T1),type="l", col="darkgrey", main
    ="Prévision pour la série originale S", ylab="S")
lines(c(s[320:n],borne_inf_T1+s[n], borne_inf_T2+prev_s_T1),type="l", col="darkgrey")
lines(c(s[320:n],prev_s_T1, prev_s_T2),lty="dotted", col="grey")
lines(s[320:n],type="l")
points(n-320+2, prev_s_T1,bg='tomato2', pch=21, cex=1, lwd=2)
points(n-320+3, prev_s_T2,bg='tomato2', pch=21, cex=1, lwd=2)
}
IC_S()

# Intervalle de confiance bivarié :
Sigma <- matrix(c(1, phi_1+psi_1,phi_1+psi_1,1+(phi_1+psi_1)^2), ncol=2)
inv_Sigma <- solve(Sigma)
plot(prev_T1,prev_T2,xlim=c(-4,4),ylim=c(-4,4), xlab="Prévision de X(T+1)", ylab="
    Prévision de X(T+2)", main="Région de confiance bivariée à 95%")
```

```r
lines(car::ellipse(center = c(prev_T1,prev_T2), shape= inv_Sigma, radius=sqrt(qchisq
    (0.95,df=2)) ))
```