Modélisation ARIMA d'une série temporelle

ENSAE PARIS

Maha BOUJENDAR
maha.boujendar@ensae.fr

Marion CHABROL
marion.chabrol@ensae.fr

Année 2022 - 2023

Table des matières

I Partie I : Les données

I.1 Que représente la série choisie ?

L'indice de la production industrielle (IPI) est un indicateur économique qui mesure les variations de la production industrielle au cours du temps. Il permet de suivre l'évolution de la production dans un secteur industriel donné. La série que nous avons choisie représente l'indice de la production industrielle concernant la fabrication de charpentes et d'autres menuiseries. Il s'agit d'un indicateur économique qui peut fournir des informations sur l'activité économique dans le secteur de la construction et de l'industrie du bois en général.

En effet, les charpentes et les menuiseries étant des éléments fondamentaux dans la construction, leur production est souvent un indicateur clé de l'activité économique dans le secteur de la construction. De plus, la production de charpentes et de menuiseries est souvent représentative de la production du secteur de l'industrie du bois, industrie porteuse pour l'économie de nombreuses régions, notamment dans les zones rurales. Enfin, l'indice de production industrielle pour la fabrication de charpentes et de menuiseries peut également être utilisé par les entreprises pour évaluer leur propre performance.

Cet indicateur peut donc être utilisé par les décideurs économiques, les entreprises et les investisseurs pour évaluer la santé économique globale d'une région, suivre les tendances de la production et prendre des décisions en conséquence.

Notre série est CVS-CJP, c'est-à-dire corrigée des valeurs saisonnières et des jours ouvrés. Elle contient des données mensuelles et recense 398 observations.

I.2 Transformation de la série

Notons X_t la série initiale (non transformée). Elle comporte 398 observations étalées à fréquence mensuelle de février 1990 à octobre 2022.

Figure 1 – X_t

- Le graphique de la série laisse supposer une tendance linéaire décroissante, mais pas forcément de saisonnalité (ce qui était attendu puisque la série est déjà corrigée).

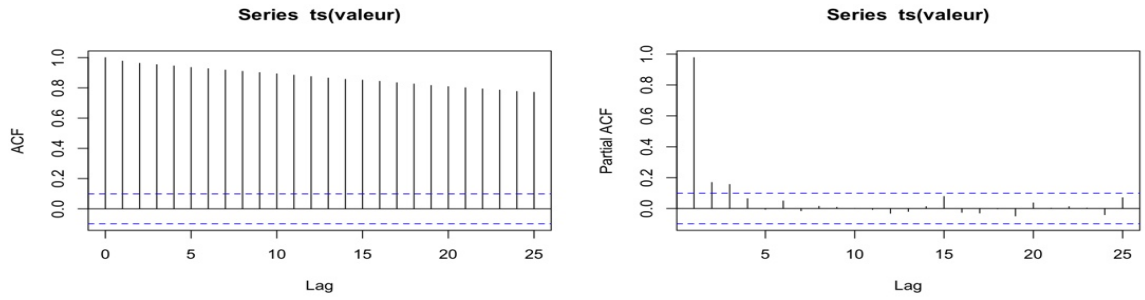


Figure 2 – ACF et PACF de X_t

A partir des fonctions d'autocorrélation et d'autocorrélation partielles, nous observons que :

- Les autocorrélations partielles (PACF) ne montrent pas de motif répété : la série ne semble pas présenter de saisonnalité.

- Les autocorrélations (ACF) diminuent très progressivement et l'autocorrélation partielle (PACF) d'ordre 1 est proche de 1 : la série ne semble pas stationnaire.

Nous allons maintenant procéder à un test de Dickey-Fuller augmenté (ADF) pour confirmer nos observations. La regression de la série X_t sur les dates (t) et une constante présente des coefficients significativement différents de 0 au niveau 5% pour t et pour la constante -cf TABLE 1-. On effectue donc le test ADF de type "ct" (avec constante et tendance) pour la série X_t. L'ADF exige aussi que les résidus ne soient pas autocorrélés. Pour cela, on utilise le test de Ljung Box qui augmente le nombre de retards tant que les résidus sont autocorrélés au niveau 5% (cf code).

Le test ADF suggère que la série X_t n'est pas stationnaire. La p_valeur étant supérieure à 5%, nous ne pouvons pas rejeter l'hypothèse nulle de racine unité (et donc de non stationnarité).

On différencie notre série dans le but de la rendre stationnaire : $dX_t = X_t - X_{t-1}$. En regressant la série dX_t sur les dates (t) et une constante, on obtient que ni t ni la constante ne sont significatives -cf TABLE 1-. On effectue donc le test ADF de type "nc" (sans constante ni tendance). La p_valeur du test ADF est cette fois inférieure au niveau 5% : on rejette l'hypothèse nulle de racine unitaire (non stationnarité). La série dX_t est donc bien stationnaire.

Xt	**t value**	**Pr(>|t|)**	**sign.**	**dXt**	**t value**	**Pr(>|t|)**	**sign.**
Intercept.	35.83	<2e-16	***	**Intercept.**	-0.011	0.991	
Dates	-35.12	<2e-16	***	**Dates**	0.007	0.994	

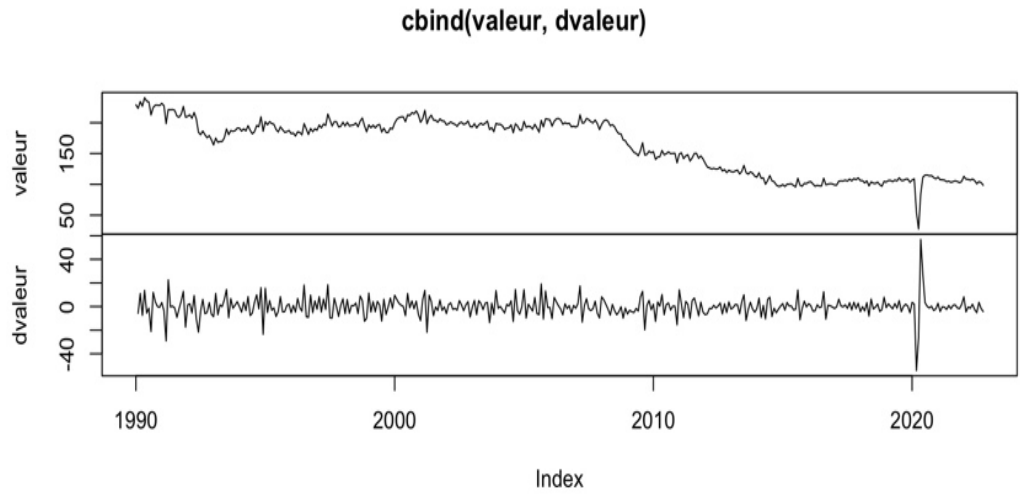
Table 1 – Regression sur une constante et sur les dates

Série	Lag	Stat	p_valeur
X_t	3	-2.123	0.5253
dX_t	2	-15.6666	< 0.01

Table 2 – Tests ADF

I.3 Comparaison graphique des séries

Les graphiques des 2 séries X_t et dX_t sont les suivants :

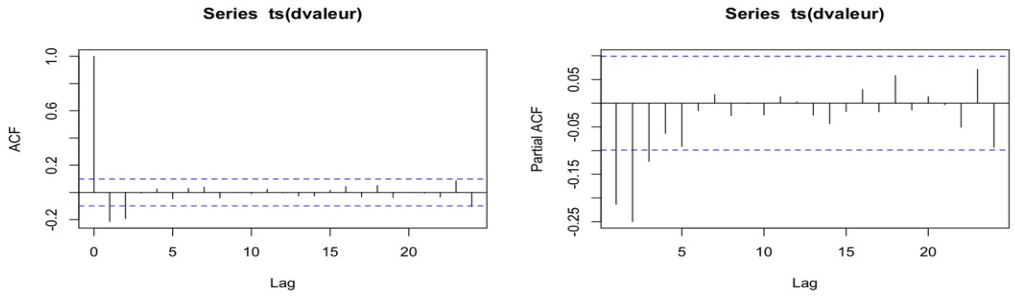


La série transformée dX_t semble corrigée de la tendance et est centrée en 0. On remarque un pic en 2020, qui correspond à la pandémie de covid.

II Partie II : Modèles ARMA

II.1 Choix du modèle ARMA(p,q)

Pour le choix du modèle ARMA(p,q), on considère la série différenciée dX_t. Nous allons identifier les ordres q_{max} et p_{max} à partir de l'autocorrélogramme et de l'autocorrélogramme partiel de dX_t.



Soit h $\in \mathbb{Z}$ l'horizon de temps. La dernière autocorrélation (ACF) significativement différente de 0 correspond à h = 2, on pose donc $q_{max} = 2$. De même, la dernière autocorrélation partielle (PACF) significative correspond à h = 3 (l'abcisse du graphe PACF commence à 1), on pose donc $p_{max} = 3$.

A présent, on va estimer tous les modèles ARMA(p,q) existants avec p $\leq p_{max}$ et q $\leq q_{max}$, et on va sélectionner tous les modèles valides et bien ajustés. On gardera ensuite le meilleur de ces modèles, meilleur au sens des critères d'information AIC et BIC.

Pour vérifier qu'un modèle ARMA(q,p) est bien ajusté, on estime ce modèle à l'aide de la fonction R *arima*, puis on calcule les p_valeurs des coefficients, avec la formule p_valeur $= \frac{coefficient}{standard\_error}$. Si les coefficients de plus haut ordre sont significativement différents de 0 (ie si p_valeur ≤ 0.05 pour ces coefficient), alors le modèle est bien ajusté.

Pour vérifier qu'un modèle ARMA(p,q) est valide, on effectue un test d'autocorrélation des résidus. Si l'absence d'autocorrélation des résidus n'est pas rejetée, le modèle est valide.

Suite à cette procédure, nous avons retenu 4 modèles : AR(3), ARMA(1,1), ARMA(2,1) et MA(2). Pour selectionner le meilleur de ces modèles, nous calculons l'AIC et le BIC de chaque modèle. Ces critères d'évaluation ont pour objectif de trouver le modèle qui décrit le mieux les données, tout en évitant de surajuster les données (overfitting). Ces critères reposent donc sur un compromis entre la qualité de l'ajustement et la complexité du modèle, en pénalisant les modèles ayant un grand nombre de paramètres, ce qui limite les effets du sur-ajustement. Le modèle minimisant à la fois l'AIC et la BIC est le MA(2) -cf TABLE 3-, c'est donc ce dernier que nous retenons pour la suite de l'analyse.

	AR(3)	ARMA(1,1)	ARMA(2,1)	MA(2)
AIC	2720.680	2719.519	2717.687	**2715.895**
BIC	2736.575	2731.441	2733.583	**2727.816**

Table 3 – Critères d'information

Nous obtenons à la fin de cette partie :

$$dX_t = \psi(B) * \epsilon_t \tag{1}$$

avec $\psi(B) = 1 - 0.305B - 0.204B^2$ *et B l'opérateur retard. Ces coefficients ont été obtenus grâce à la commande ma2$coef sur R.*

II.2 Expression du modèle ARIMA(p,d,q)

Avant de passer au modèle ARIMA, il faut vérifier que notre modèle ARMA est canonique. Notre modèle étant un MA(2), il est immédiatement causal. De plus, les racines du polynome $\psi(B)$ sont 1.59 et -3.08, de modules supérieurs à 1. Notre modèle est donc inversible. Le modèle étant un MA(2), nous n'avons pas de racine commune entre $\psi(B)$ et $\phi(B)$. Le modèle est donc bien canonique, et on peut interprêter ϵ_t comme une

innovation.
Pour obtenir la serie dX_t, nous avions différencié la série initiale X_t une fois. dX_t est un ARMA(0,2), X_t est donc un ARIMA(0,1,2).

On obtient finalement :

$$(1-B)^1 * X_t = \psi(B) * \epsilon_t \tag{2}$$

où $(1-B)^1 * X_t = dX_t$ *est un ARMA causal,* $\psi(B) = 1 - 0.305B - 0.204B^2$*, et B l'opérateur retard.*

III Partie III : Prévision

III.1 Région de confiance

Notre série différenciée dX_t suit un processus MA(2) :

$$dX_t = \epsilon_t - 0.305\epsilon_{t-1} - 0.204\epsilon_{t-2} \tag{3}$$

Notre processus est stationnaire.
Soit $\hat{dX_{t+h|t}}$ la meilleure prévision de dX_{t+h} sachant $dX_t, dX_{t-1}...$, avec h l'horizon de temps considéré.

Les erreurs de prévision en t+1 et en t+2 s'écrivent :
— $e_{t+1} = dX_{t+1} - \hat{dX_{t+1|t}} = dX_{t+1} - \mathbb{E}L[dX_{t+1}|dX_t, dX_{t-1}...]$
$= \epsilon_{t+1} + \psi_1\epsilon_t + \psi_2\epsilon_{t-1} - (\psi_1\epsilon_t + \psi_2\epsilon_{t-1}) - \mathbb{E}L[\epsilon_{t+1}|dX_t, dX_{t-1}...] = \epsilon_{t+1}$
car ϵ_t est une innovation donc $\mathbb{E}L[\epsilon_{t+1}|dX_t, dX_{t-1}...] = 0$.

— $e_{t+2} = dX_{t+2} - \hat{dX_{t+2|t}} = dX_{t+2} - \mathbb{E}[dX_{t+2}|dX_t, dX_{t-1}...]$
$= \epsilon_{t+2} + \psi_1\epsilon_{t+1} + \psi_2\epsilon_t - (\psi_2\epsilon_t) - \psi_1\mathbb{E}L[\epsilon_{t+1}|dX_t, dX_{t-1}...] - \mathbb{E}L[\epsilon_{t+2}|dX_t, dX_{t-1}...] = \epsilon_{t+2} + \psi_1\epsilon_{t+1}$
car ϵ_t est une innovation donc $\mathbb{E}L[\epsilon_{t+1}|dX_t, dX_{t-1}...] = \mathbb{E}L[\epsilon_{t+2}|dX_t, dX_{t-1}...] = 0$.

Or :
— $Var(e_{t+1}) = Var(\epsilon_{t+1}) = \sigma^2$
— $Var(e_{t+2}) = Var(\epsilon_{t+2} + \psi_1\epsilon_{t+1}) = (1+\psi_1^2)\sigma^2$
— $cov(e_{t+1}, e_{t+2}) = cov(\epsilon_{t+1}, \epsilon_{t+2} + \psi_1\epsilon_{t+1}) = \psi_1\sigma^2$ *avec* $\sigma^2 = \mathbb{V}(\epsilon_t)$

Selon certaines hypothèses détaillées dans la question suivante,

$$\begin{pmatrix} dX_{t+1} - \hat{dX_{t+1|t}} \\ dX_{t+2} - \hat{dX_{t+2|t}} \end{pmatrix} = \begin{pmatrix} \epsilon_{t+1} \\ \epsilon_{t+2} + \psi_1\epsilon_{t+1} \end{pmatrix} \sim \mathcal{N}(0, \Sigma), \text{ où } \Sigma = \begin{pmatrix} \sigma^2 & \psi_1\sigma^2 \\ \psi_1\sigma^2 & (1+\psi_1^2)\sigma^2 \end{pmatrix}$$

De plus, $det(\Sigma) = \sigma^4$ donc Σ est inversible si et seulement si $\sigma \neq 0$, ce que nous avons supposé vrai.
Notons $Y := \begin{pmatrix} dX_{t+1} \\ dX_{t+2} \end{pmatrix}$ et $\hat{Y} := \begin{pmatrix} \hat{dX_{t+1|t}} \\ \hat{dX_{t+2|t}} \end{pmatrix}$. Par le cours, nous savons que $(Y-\hat{Y})'\Sigma^{-1}(Y-\hat{Y}) \sim \chi^2(2)$.

La région de confiance bivariée de niveau α est donc :

$$\left\{ y \in \mathbf{R}^2 | (y-\hat{Y})'\Sigma^{-1}(y-\hat{Y}) \leq q_{1-\alpha}^{\chi^2(2)} \right\} \tag{4}$$

avec $q_{1-\alpha}^{\chi^2(2)}$ le quantile d'ordre $(1-\alpha)$ de la loi $\chi^2(2)$.

Par ailleurs, comme $e_{t+1} \sim \mathcal{N}(0, \sigma^2)$ et $e_{t+2} \sim \mathcal{N}(0, (1+\psi^2)\sigma^2)$, on peut déterminer les intervalles de confiances unidimensionnels de niveau α respectifs de dX_{t+1} et dX_{t+2} :

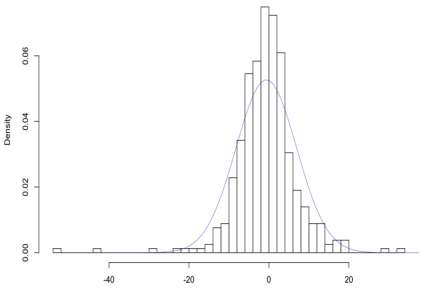
$$\left[\hat{X_{t+1|t}} - q_{1-\alpha/2}^{\mathcal{N}(0,1)} * \sqrt{\hat{\sigma}^2}, \hat{X_{t+1|t}} + q_{1-\alpha/2}^{\mathcal{N}(0,1)} * \sqrt{\hat{\sigma}^2} \right] \tag{5}$$

$$\left[\hat{X_{t+2|t}} - q_{1-\alpha/2}^{\mathcal{N}(0,1)} * \sqrt{(1+\psi^2)\hat{\sigma}^2}, \hat{X_{t+2|t}} + q_{1-\alpha/2}^{\mathcal{N}(0,1)} * \sqrt{(1+\psi^2)\hat{\sigma}^2}\right] \quad (6)$$

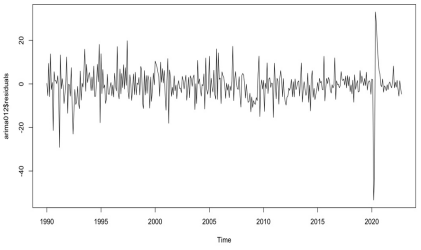
avec $q_{1-\alpha/2}^{\mathcal{N}(0,1)}$ le quantile d'ordre $(1-\alpha/2)$ de la loi $\mathcal{N}(0,1)$, et en considérant que $\hat{\sigma}^2$, l'estimateur de σ^2, est suffisamment proche de σ^2.

III.2 Hypothèses

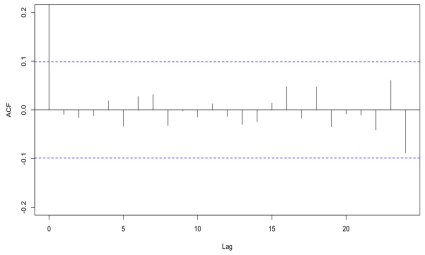
Afin d'écrire la région de confiance de niveau α ci-dessus, nous avons fait l'hypothèse que les innovations étaient des bruits blancs gaussiens. Nous allons à présent vérifier cette hypothèse. Dans le graphique (a) qui suit, nous avons superposé la densité d'une loi normale (de moyenne et de variance suivies par nos résidus) en bleu, à l'histogramme des résidus. L'hypothèse d'une répartition gaussienne semble cohérente vis à vis de l'histogramme. La courbe bleue et l'histogramme rendent en effet compte d'une répartition similaire, bien que l'histogramme présente une variance légèrement plus faible. D'après les graphes (b) et (c), les résidus semblent se comporter comme un bruit blanc.



(a) Histogramme des résidus



(b) Résidus



(c) ACF des résidus

De plus, nous avons fait l'hypothèse que notre modèle était bien spécifié : bien ajusté, avec les coefficients correctement estimés. D'après le graphe (a) ci-dessus, nous pouvons constater que le modèle gaussien surestime la variance de nos résidus.

Pour la construction de l'intervalle de confiance univarié à 95%, nous avons considéré que notre estimateur $\hat{\sigma}^2$ était suffisamment proche de σ^2.

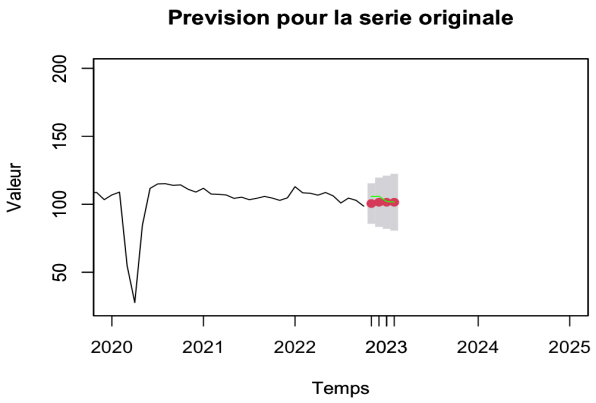
Notons par ailleurs que nous avons considéré les ϵ_t comme les innovations linéaires de dX_t, ce que nous avons justifié par le fait que notre modèle ARMA est canonique (notre modèle est un MA(2) avec les racines de ψ à l'extérieur du cercle unité, cf question II.2).

Nous avons enfin effectué un test d'autocorrélation des résidus (Portmanteau). Les résultats du tableau ci-dessous, obtenu grâce à la commande *Qtests(residus, 24, fitdf=2)*, montrent que l'absence d'autocorrélation des résidus n'est jamais rejetée : les p_valeurs sont toutes supérieurs à 0.60.

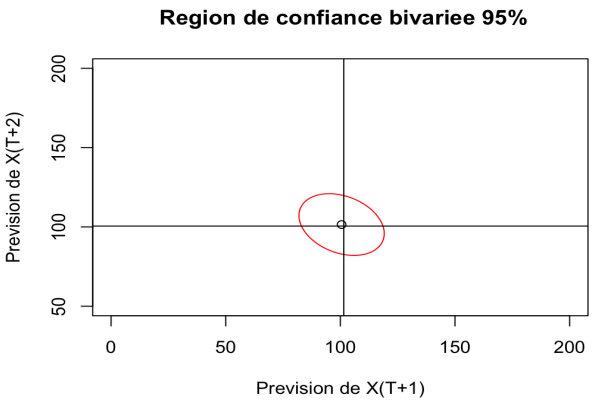
lag	pval	lag	pval
1	NA	13	0.9958283
2	NA	14	0.9971359
3	0.6586414	15	0.9985191
4	0.8451050	16	0.9968226
5	0.8529826	17	0.9981536
6	0.8957126	18	0.9967261
7	0.9147397	19	0.9968126
8	0.9282509	20	0.9982626
9	0.9648338	21	0.9990493
10	0.9810586	22	0.9987868
11	0.9904602	23	0.9965705
12	0.9952111	24	0.9762883

Table 4 – Test d'autocorrélation des résidus

III.3 Répresentation graphique



(a) Prévision pour la série originale



(b) Région de confiance bivariée à 95%

La figure a) montre, en rouge, la prévision pour les 4 prochaines valeurs de notre série X_t obtenues à partir du modèle ARIMA et grâce à la fonction forecast. La courbe verte correspond aux 4 valeurs de notre série initiale (retirées aux début du projet, pour constituer l'échantillon test). Le niveau de confiance des intervalles de prévision est ici défini à 95%. Nous pouvons remarquer que nos valeurs prédites sont différentes des valeurs réelles de la série initiale, bien qu'elles appartiennent à l'intervalle de confiance déterminé plus haut : cela pourrait être dû aux fortes fluctuations de la série durant l'année 2020.

La figure b) représente quant à elle la région de confiance bivariée à 95% de dX_{t+1} et dX_{t+2}. Elle estime la plage de valeurs probables pour nos deux variables aléatoires au niveau de confiance 95%. La région bivariée représente explicitement l'impact de notre erreur de prédiction à horizon 1 (e_{t+1}) sur celle à horizon 2 (e_{t+2}).

Valeurs observées	105.74	105.88	102.06	101.66
Valeurs prédites	100.5449	101.4958	101.4958	101.4958

Table 5 – Comparaisons des préditions aux valeurs observées grâce à l'échantillon test

III.4 Question ouverte

Nous voulons ici savoir si une série temporelle fournit une information utile pour prédire une autre série temporelle, ou dit autrement, si Y_t cause instantanément X_t au sens de Granger. Dans ce cas, la connaissance de Y_{t+1} permet d'améliorer la prévision de X_{t+1}.

Nous pouvons vérifier la validité de cette hypothèse grâce à un test de Granger : ce test compare les performances de deux modèles de prédiction de X_t, l'un utilisant l'information supplémentaire et l'autre non. Il est toutefois important de noter que ce test ne permet pas de conclure à une relation de cause à effet dans le sens traditionnel de la causalité (vu en économétrie) : il nous indique uniquement si l'utilisation de l'information supplémentaire permet de donner une meilleure prédiction.

IV ANNEXE : Code R

```r
rm(list = ls())

#install.packages("zoo")
#install.packages("tseries")
#install.packages("forecast")

library(zoo)
library(tseries)
library(fUnitRoots)

## importation des données

path <- "/Users/marionchabrol/Documents/ENSAE/serie temp/"
setwd(path) #definit l'espace de travail (working directory ou "wd")
getwd() #affiche le wd
list.files() #liste les elements du wd

datafile <- "projet_charpente.csv"
data <- read.csv(datafile,sep=";")

dates_char <- as.character(data$date)
dates_char[1] #date début
tail(dates_char,1) #date fin
dates <- as.yearmon(seq(from=1990+0/12, to=2023+1/12, by=1/12)) #définition des dates
valeur <- zoo(data$valeur, order.by=dates)

T <- length(valeur)

test <- valeur[(T-3):T]
valeur <- valeur[1:(T-4)] #supprime les 5 dernieres valeurs

dates <- dates[1:(T-4)]

monthplot(valeur)
plot(valeur)

#2
##différenciation

dvaleur <- diff(valeur,1)

par(mfrow=c(1,2))
plot(valeur)
plot(dvaleur)

acf(ts(valeur))
acf(ts(dvaleur))

pacf(ts(valeur))
pacf(ts(dvaleur))
```

```r
#Test de stationnarité ADF

Qtests <- function(series, k, fitdf=0) {
  pvals <- apply(matrix(1:k), 1, FUN=function(l) {
    pval <- if (l<=fitdf) NA else Box.test(series, lag=l, type="Ljung-Box",
    ↪  fitdf=fitdf)$p.value
    return(c("lag"=l,"pval"=pval))
  })
  return(t(pvals))
}

#tests ADF jusqu'à des residus non autocorrelés
adfTest_valid <- function(series, kmax, adftype){
  k <- 0
  noautocorr <- 0
  while (noautocorr==0){
    cat(paste0("ADF with ",k," lags: residuals OK? "))
    adf <- adfTest(series, lags=k, type=adftype)
    pvals <- Qtests(adf@test$lm$residuals, 24, fitdf =
    ↪  length(adf@test$lm$coefficients))[,2]
    if (sum(pvals<0.05,na.rm=T)==0) { #il faut avoir des pvals > 0.05
      noautocorr <- 1; cat("OK \n")
    } else cat("nope \n")
    k <- k+1
  }
  return(adf)
}

summary(lm(valeur ~ dates))
adf <- adfTest_valid(valeur,24,adftype="ct")
adf
#p_valeur > 0.05, on ne rejette pas l'hypothèse nulle de non stationnarité pour adf
kpss.test (valeur , null ="Trend")
#p_valeur < 0.05, on rejette l'hypothèse nulle de stationnarité pour kpss


summary(lm(dvaleur ~ dates[-1]))
#
adf <- adfTest_valid(dvaleur,24,"nc") #ok pour la série différenciée
adf
#p_valeur < 0.05, on rejette l'hypothèse nulle de non stationnarité
kpss.test(dvaleur , null ="Level")
#p_valeur > 0.05, on ne rejette pas l'hypothèse nulle de stationnarité

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

#3
#visualisation des données
plot(cbind(valeur,dvaleur))


###Partie 2
##4
```

```r
#on a retenu dvaleur
par(mfrow=c(1,2))
acf(ts(dvaleur), 24);pacf(ts(dvaleur), 24) #on regarde jusqu'à deux ans de retard
#
# estimation des ordres maximum pmax et qmax
pmax=3;qmax=2

signif <- function(estim){ #fonction de test des significations individuelles des
↪  coefficients
  coef <- estim$coef #recup coeff
  se <- sqrt(diag(estim$var.coef)) #calcul standard errors
  t <- coef/se #test de student
  pval <- (1-pnorm(abs(t)))*2
  return(rbind(coef,se,pval))
}

Qtests <- function(series, k, fitdf=0) { #series = ARMA, k=horizon temporel avec 24 (2
↪  ans, données mensuelles)
  pvals <- apply(matrix(1:k), 1, FUN=function(l) {
    pval <- if (l<=fitdf) NA else Box.test(series, lag=l, type="Ljung-Box",
    ↪  fitdf=fitdf)$p.value
    return(c("lag"=l,"pval"=pval))
  })
  return(t(pvals))
}

arma32 <- arima(dvaleur,c(3,0,2), include.mean=F)
Qtests(arma32$residuals, 24, 5) #5 = p+q = 3+2


arimafit <- function(estim){ #estim = modèle en tant que paramètres
  adjust <- round(signif(estim),3) #enlève les p_valeurs avec mille chiffres
  pvals <- Qtests(estim$residuals,24,length(estim$coef)-1)
  pvals <- matrix(apply(matrix(1:24,nrow=6),2,function(c) round(pvals[c,],3)),nrow=6)
  colnames(pvals) <- rep(c("lag", "pval"),4)
  cat("tests de nullite des coefficients :\n")
  print(adjust)
  cat("\n tests d'absence d'autocorrelation des residus : \n")
  print(pvals)
}

estim <- arima(dvaleur,c(3,0,2), include.mean=F); arimafit(estim)
#pas bien ajusté, valide

estim <- arima(dvaleur,c(3,0,1), include.mean=F); arimafit(estim)
#pas bien ajusté, valide

estim <- arima(dvaleur,c(3,0,0), include.mean=F); arimafit(estim)
#bien ajusté, valide
ar3 <- arima(dvaleur,c(3,0,0), include.mean=F)

estim <- arima(dvaleur,c(2,0,1), include.mean=F); arimafit(estim)
#bien ajusté, valide
arma21 <- arima(dvaleur,c(2,0,1), include.mean=F)
```

```r
estim <- arima(dvaleur,c(2,0,0), include.mean=F); arimafit(estim)
#bien ajusté, pas valide

estim <- arima(dvaleur,c(1,0,2), include.mean=F); arimafit(estim)
#pas bien ajusté, valide

estim <- arima(dvaleur,c(1,0,1), include.mean=F); arimafit(estim)
#bien ajusté, valide
arma11 <- arima(dvaleur,c(1,0,1), include.mean=F)

estim <- arima(dvaleur,c(1,0,0), include.mean=F); arimafit(estim)
#bien ajusté, pas valide

estim <- arima(dvaleur,c(0,0,2), include.mean=F); arimafit(estim)
#bien ajusté, valide
ma2 <- arima(dvaleur,c(0,0,2), include.mean=F)

estim <- arima(dvaleur,c(0,0,1), include.mean=F); arimafit(estim)
#bien ajusté, pas valide


models <- c("ar3", "arma11", "arma21", "ma2"); names(models) <- models
apply(as.matrix(models),1, function(m) c("AIC"=AIC(get(m)), "BIC"=BIC(get(m))))

#meilleur modèle MA(2)

#5

#il faut regarder si les racines des modules du polynome sont supérieures à 1

ma2$coef
racine_ma2 <- polyroot(c(1, ma2$coef[1], ma2$coef[2]))

Mod(racine_ma2[1])
Mod(racine_ma2[2])

#Le modèle est bien causal (c'est un MA), on peut donc passer à l'ARIMA.
#Comme on a différencié une fois, d = 1. Le modèle est inversible.
#On obtient donc un ARIMA(0,1,2).

arima012 <- arima (valeur, c(0,1,2), include.mean=F)


# Partie 3 : Previsions
#6
tsdiag(arima012)
qqnorm(arima012$residuals )

plot(density(arima012$residuals ,lwd=0.5),xlim=c(-10,10), main="Densite des residus",
     xlab="Valeurs prises")

hist(arima012$residuals,40, freq = F)
mu<-mean(arima012$residuals)
sigma<-sd(arima012$residuals)
x<-seq(-40,40)
```

```r
y<-dnorm(x,mu,sigma)
lines(x,y,lwd=0.5,col="blue")



par(mfrow=c(1,1))
plot(arima012$residuals)
acf(ts(arima012$residuals),24, ylim = c(-0.2, 0.2))

#Extraction des coefs du modele et de la variance des residus
arima012$coef
phi_1 <- as.numeric(arima012$coef[1])
phi_2 <- as.numeric(arima012$coef[2])
sigma2 <- as.numeric(arima012$sigma2)
phi_1
phi_2
sigma2

# Question 8
XT1 = predict(arima012, n.ahead=2)$pred[1]
XT2 = predict (arima012, n.ahead=2)$pred[2]
XT1
XT2


# On cherche d'abord a tracer le region de confiance univariee pour la serie originale a
↪  95%.

install.packages("forecast")
library(forecast)

fore = forecast(arima012, h=5, level=95)

par(mfrow=c(1 ,1))
plot(fore ,col=1,fcol=2,shaded=TRUE,xlab="Temps",ylab="Valeur", main="Prevision pour la
↪  serie originale")

#Ensuite , on represente la region de confiance bivariee a 95%.

library(ellipse)
arma = arima0(x, order=c(0,1,2))
Sigma <- matrix (c(sigma2,phi_1*sigma2,phi_1*sigma2,(phi_1)^2*sigma2 + sigma2),ncol =2)
inv_Sigma <- solve(Sigma)
plot ( XT1,XT2 , xlim =c(0, 200) , ylim =c(50,200) , xlab =" Prevision de X(T+1) ",
       ylab ="Prevision de X(T+2)" , main =" Region de confiance bivariee 95%")

lines(ellipse(Sigma, centre=c(XT1,XT2)), type="l", col="red", xlab="Xt+1",ylab="Xt+2",
      main="Ellipse de confiance pour (Xt+1,Xt+2)")

abline(h=XT1, v=XT2)
```

Références

[1] Code : INSEE, statistiques et études, *Moyenne annuelle de la production industrielle (base 2015) - Fabrication de charpentes et d'autres menuiseries (NAF rév. 2, niveau classe, poste 16.23)*

[2] INSEE, *Enquête mensuelle de conjoncture dans l'industrie*, 2023

[3] FRANCQ Christian, *Linear Time Series* Course, 2023