

Modèles de Prédiction–Feuille d’Exercices 2. Modèles AR-MA

Exercice 1. Considérer le processus à moyenne mobile $\{X_t\}$ suivant :

$$X_t = \epsilon_t + \theta\epsilon_{t-2},$$

où $\{Z_t\} \leftrightarrow BBN(0, 1)$.

- Calculer les fonctions d’autocovariance et d’autocorrélation de ce processus.
- À l’aide de la fonction `arima.sim` de R, simuler 300 observations du processus ci-dessus avec $\theta = 0.8$. Calculer et tracer le corrélogramme du processus ainsi obtenu.
- Répéter la partie b) avec $\theta = -0.8$.
- Les corrélogrammes obtenus en b) et c) correspondent-ils à la fonction d’autocorrélation théorique calculée en a) ?
- On remarquera que la série en b) fluctue moins rapidement que celle en c). Expliquer cet état de fait à l’aide de la fonction d’autocorrélation.

Exercice 2. Ajuster un MA (1) à

$$X_t = -0.7X_{t-1} + \epsilon_t,$$

où ϵ_t est un bruit blanc gaussien de variance 4, c’est-à-dire un modèle incorrect. Effectuer un test montrant que ce modèle ne convient pas.

Exercice 3: MA(2).

On simule 200 valeurs d’une série temporelle X_t suivant le modèle moyenne mobile d’ordre 2 suivant :

$$X_t = \epsilon_t + 0.4\epsilon_{t-1} + 0.7\epsilon_{t-2}$$

où (ϵ_t) est un bruit blanc gaussien d’écart-type 0.1.

```
X=arima.sim(n=200,list(ma=c(0.4,0.7)),sd=.1)
plot(X,ylab="X",type="o",pch=20)
```

Lancez plusieurs simulations pour constater que les valeurs changent d’une exécution à l’autre.

Identifiez un modèle MA(2) sur les données générées, validez ce modèle, et estimez les paramètres.

Utiliser les fonctions suivantes :

```
-- Acf(X)
Pacf(X)
tsdisplay(X) # que l’on privilégiera
```

```

pour ACF et PACF de la chronique X (package forecast, plutôt que acf / pacf dans
stats) ;

# Attention, dans Acf, le décalage 0 (ACF(0) valant toujours 1) n'est pas représenté
(contrairement à acf)

-- output_X=Arima(X,order=c(0,0,2))
summary(output_X)

#pour estimer les paramètres d'un processus MA(2) (package forecast, plutôt que arima dans stat
tsdiag(output_X)

# pour l'étude des résidus du modèle ;
-- hist(residuals(output_X), freq = F, col = "grey")
curve(dnorm(x, mean = mean(residuals(output_X)),
sd = sd(residuals(output_X))), col = 2, add = TRUE)
qqnorm(residuals(output_X))
qqline(residuals(output_X),col=2)
pour histogramme et QQ-plot des résidus ;
-- coeftest(output_X)
coefci(output_X)

# pour tester la significativité des coefficients et afficher les intervalles de confiance,
une fois la gaussianité du modèle établie (dans le package lmtest).

```

Corrigé de l'exercice 3.

```

# Exercice 3: MA(2)
#####

X=arima.sim(n=200,list(ma=c(0.4,0.7)),sd=.1)
plot(X, ylab="X", type="o", pch =20)

# IDENTIFICATION
acf(X)
windows()
pacf(X)
# ou les deux sur le même graphe:

```

```

par(mfrow=c(2,1))
acf(X)
pacf(X)
par(mfrow=c(1,1)) # revenir à l'état antérieur (un graphique par fenêtre)

# ou alors, ce qu'on utilisera dans la suite:
# install.packages("forecast")
library(forecast)
tsdisplay(X)

# Explication Concernat PACF et ACF: .....
#Modél à retenir.....

# REMARQUE importante pour l'identification:
# attention: l'ACF est par définition égal à 1 pour h=0
# (la première barre monte donc toujours jusque 1 pour lag=0),
# le PACF n'est pas défini pour h=0 (la première barre est donc calculée pour lag=1)
# attention bis: Acf du package forecast commence à h=1 (donc pas de 1ère barre à 1)

# ESTIMATION des paramètres d'un MA(2):
# On va utiliser la procédure Arima du package "forecast" (à installer au préalable)
# (plus complet que la procédure arima standard)

# on va aussi utiliser Acf et Pacf (dans forecast) plutôt que acf et pacf
# Attention, dans Acf,
# le décalage 0 (ACF(0) valant toujours 1) n'est pas représenté
# (contrairement à acf)

output_X=Arima(X,order=c(0,0,2))
summary(output_X)
# on aura besoin de:
output_X$coef # les coefficients du modèle  $X_t = \text{mean} + \text{ma1} \text{eps}_{\{t-1\}} + \text{ma2} \text{eps}_{\{t-2\}}$ 
output_X$sigma2 # estimation variance
output_X$loglik # log-vraisemblance

```

```

output_X$aic      # critère d'information AIC
output_X$bic      # critère d'information BIC
output_X$residuals # les résidus

# REMARQUE1 : le coef "intercept" (dans coef) est bien la moyenne m, pas la constance c
# REMARQUE 2: les valeurs numériques des paramètres estimés
varient d'une génération aléatoire à l'autre

#EXAMINATION des RESISUS
# On examine les résidus pour vérifier l'hypothèse BBG.
plot(output_X$residuals, ylab="residus")
abline(0, 0)
Acf(residuals(output_X))
# ou plus simplement:
tsdiag(output_X)

#Comentaire.....

# REPATION GAUSSIENNE ?
hist(residuals(output_X), freq = F, col = "grey")
curve(dnorm(x, mean = mean(residuals(output_X)),
sd = sd(residuals(output_X))), col = 2, add = TRUE)
# QQ plot (cf droite de Henry):
qqnorm(residuals(output_X))
qqline(residuals(output_X))

# conclusion: .....

# SIGNIFICATIVITE
# install.packages("lmtest") # si pas installé
library(lmtest)
coeftest(output_X)
# la moyenne n'est pas significative
coefci(output_X) # intervalles de confiance (cf help(coefci) pour fixer un autre risque)

```

```

# on force constante à 0
output_X=Arima(X,order=c(0,0,2),include.mean=FALSE)

summary(output_X)

# On examine les résidus sur ce dernier modèle
(forcer la constante non-significative à 0 ne change pas grand chose)
tsdiag(output_X)

# QQ plot (cf droite de Henry)
qqnorm(residuals(output_X))
qqline(residuals(output_X))

hist(residuals(output_X), freq = F, col = "grey")
curve(dnorm(x, mean = mean(residuals(output_X)),
sd = sd(residuals(output_X))), col = 2, add = TRUE)

# le modèle est validé

# significativité
coefstest(output_X)

# intervalles de confiance (pour info)
coefci(output_X)

# On valide le modèle final (il faut savoir l'écrire)
# Modèle final: X_t = eps_t + 0.421 eps_{t-1} + 0.682 eps_{t-2}
# ou: X_t = (1 + 0.421 B + 0.682 B^2) eps_t
# avec \sigma estimé à 0.099 (RMSE)
# (les valeurs exactes dépendent de la simulatio

```

Exercice 4: AR(1)

On considère une série suivant un modèle autorégressif d'ordre 1 :

$$(1 - 0.8B)(X_t - 2) = \epsilon_t$$

générée par :

$$X = 2 + \text{arima.sim}(n = 200, \text{list}(\text{ar} = c(.8)), \text{sd} = .5)$$

Identifiez un modèle AR.

Exercice 5: AR(2) particulier

On considère une série suivant un modèle autorégressif d'ordre 2 :

$$(1 - 0.8B^2)(X_t + 5) = \epsilon_t$$

générée par :

$$X = -5 + \text{arima.sim}(n = 200, \text{list}(\text{ar} = c(0, .8)), \text{sd} = .8)$$

Identifiez un modèle ARMA.

Modèles ARMA-ARIMA

Exercice 6. Simuler une trajectoire de 200 observations du modèle ARMA(1,2) suivant

$$X_t = -10 + \frac{1 - 0.3B + 0.6B^2}{1 + 0.8B} \epsilon_t$$

où $\epsilon_t \leftrightarrow BBN(0, 1.5)$ et comparer les ACF et PACF théoriques et empiriques

Corrigé

```
#Simulation
```

```
set.seed(4123)
```

```
yc = arima.sim(n = 200, list(ar = -0.8, ma = c(-0.3, 0.6)), sd = sqrt(1.5))
```

```
## Calcule de l'ACF et PACF théoriques
```

```
acf.th = ARMAacf(ar = -0.8, ma = c(-0.3, 0.6), lag.max = 20, pacf = FALSE)
```

```
pacf.th = ARMAacf(ar = -0.8, ma = c(-0.3, 0.6), lag.max = 20, pacf = TRUE)
```

```
## Représentation de ces quatre fonctions sur un même graphique
```

```
plotacfthemp(yc, ar = -0.8, ma = c(-0.3, 0.6), lag.max = 20)
```

```
## L'estimation du paramètres du modèle, par exemple sur yc, s'obtient par
```

```
mod12 = Arima(yc, order=c(1, 0, 2))
```

```
summary(mod12)
```

```
## Remarque: Arima() note intercept, ce qui est ici, la moyenne de la série.
```

Annexe: Tableau des propriétés

	ACF	PACF
AR(p)	$\rho(h)$: décroissance exponentielle vers 0 ($h \rightarrow \infty$)	$r(h) = 0$ si $h > p$
MA(q)	$\rho(h) = 0$ si $h > q$	$r(h)$ décroissance exponentielle vers 0
ARMA(p, q)	décroissance exponentielle vers 0	décroissance exponentielle vers 0

Ces propriétés servent à identifier la nature des séries temporelles. Sous R , on utilisera les fonctions `acf` , `pacf` qui tracent, respectivement, les $\hat{\rho}(h)$ et $\hat{r}(h)$ (les auto-corrélations empiriques et les auto-corrélations partielles empiriques).

- Le logiciel trace en plus un niveau bleu horizontal en $y = m_\alpha$ tel que pour tout h ,

$$P(|\hat{\rho}(h)| \geq m_\alpha | \rho(h) = 0) = \alpha$$

(en général, le niveau α est fixé à 0,05) (la situation est la même pour les auto-corrélations partielles empiriques)

- On peut fixer α en ajoutant l'option : `acf(..., ci = alpha)`.

- Un $\hat{\rho}(h)$ sous la courbe bleue est donc non significatif (au niveau α). Le raisonnement est le suivant : on suppose $\rho(h) = 0$, si $|\hat{\rho}(h)| < m_\alpha$, on considère qu'il n'est pas nécessaire de revenir sur cette hypothèse de départ. La probabilité de rejeter à tort l'hypothèse "nulle" ($\rho(h) = 0$) est α .

- Si on appelle les fonctions `acf` ou `pacf` sans préciser le `lag.max` , le logiciel le fixe par défaut à $10 \log_{10}(n)$ (où n est la longueur de la série). La raison est que l'on ne veut pas prendre le `lag.max` trop grand parce que les moyennes empiriques ne convergent pas bien pour h grand. On conseille en général de prendre $n \geq 50$ et $h \leq n/4$.

Annexe B: Modélisation: Méthode de Box et Jenkins (1970)

Comment modéliser une chronique X_t par un ARMA(p,q) ?

- Identification des paramètres p et q . Utilisation des corrélogrammes.

Rappel

	ACF	PACF
AR(p)	$\rho(h)$: décroissance exponentielle vers 0	$r(h) = 0$ si $h > p$
MA(q)	$\rho(h) = 0$ si $h > q$	$r(h)$ décroissance exponentielle vers 0

Remarque pour l'identification:

- attention: l'ACF est par définition égal à 1 pour $h = 0$ (la première barre monte toujours jusque 1 pour `lag=0`),

- le PACF n'est pas défini pour $h=0$ (la première barre est donc calculée pour $\text{lag}=1$)
- attention bis: Acf du package forecast commence à $h=1$ (donc pas de 1ère barre à 1)
- Estimation des coefficients θ_j, ϕ_i , et σ^2
- Validation du modèle.
- Prévision du futur.