

Projet Séries Chronologiques

Alexis Buckens - 31150800 - Groupe 11 - xsales

26 novembre 2016

1 Observation des données

Dans un premier temps, il convient d'observer les données.

Les données semblent avoir une tendance et une saisonnalité, et par ailleurs, on dirait que la variabilité des données augmente avec le temps. Pour constater si une tendance est présente, le plus simple est de tracer la droite de la régression linéaire.

La pente de la droite semble bien indiquer que la série possède une tendance croissante, il faudra donc en tenir compte. Il faut maintenant s'intéresser aux saisonnalités de la série. Pour ce faire, on peut tracer l'évolution de la série sur un an pour chaque année, et comparer l'évolution de chaque année sur un seul graphique.

On voit sur le graphique qu'il semble y avoir une augmentation aux alentours de la même période chaque année et également une diminution à la même période chaque année. Le graphique laisse donc penser qu'il y a bien une saisonnalité tout les 12 mois. Une autre façon d'observer une saisonnalité consiste à calculer la moyenne annuelle et à comparer chaque mois avec la moyenne annuelle.

Année	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1965	+	-	-	-	-	-	-	+	+	+	+	+
1966	+	-	-	-	-	-	+	+	+	+	+	+
1967	+	-	-	-	-	-	-	+	+	+	+	+
1968	+	+	+	-	-	-	+	+	+	+	+	+
1969	+	+	+	-	-	-	+	+	+	+	+	+
1970	+	+	+	+	-	-	+	+	+	+	+	+

Ici, on constate bien aussi que les mois de Août à Janvier sont systématiquement au-dessus de la moyenne annuelle, tandis que les mois de Mai et Juin sont systématiquement en-dessous. Ce tableau vient confirmer la première impression du graphique précédent et laisse à penser qu'une saisonnalité doit être présente dans la série.

Maintenant que nous avons estimé que la série possède une tendance et une saisonnalité de longueur 12 approximativement, il nous reste à vérifier si la variabilité de la série est constante. Pour cela, on peut commencer par regarder si la variabilité de la série semble augmenter en observant le graphe de celle-ci lorsque la tendance est éliminée.

Comme la série semble être de plus en plus large, il semble, intuitivement, que la variabilité de celle-ci augmente avec le temps. Une autre façon de s'intéresser à la variabilité consiste à calculer l'étendue des valeurs de chaque année et à tracer le graphique de l'évolution de l'étendue année par année. Si les valeurs semblent augmenter, ce sera l'indice d'une augmentation de la variabilité.

On voit clairement sur le graphique que la variabilité semble augmenter au cours du temps. Il faut donc commencer par éliminer cette augmentation de la variabilité. Une méthode pour y arriver consiste à prendre le logarithme de la série. Si on transforme la série de cette manière et que l'on recalcule l'étendue année par année, on obtient cette fois le graphique suivant.

La variation d'amplitude est désormais bien moins importante, et on ne voit plus aucune augmentation claire. On peut donc considérer que prendre le logarithme suffit pour enlever l'augmentation de la variabilité. Comme le logarithme permet de passer d'une multiplication à une addition, on peut décomposer la série d'origine avec la fonction "decompose", en utilisant un modèle multiplicatif. Pour le reste du projet à l'exception de la prédiction, il sera plus simple de travailler sur la série transformée. La décomposition de la série donne les graphiques suivants.

Ces graphiques nous donnent des informations proches de celles dont on disposait déjà. On peut d'abord voir qu'il y a bien une tendance, et que par ailleurs, celle-ci semble plus ou moins linéaire, ce qui nous incitera à prendre une différence d'ordre 1 lorsque l'on voudra obtenir une série stationnaire. De plus, on voit aussi que la série présente bien des saisonnalités marquées.

Maintenant que nous avons ces informations sur la série et que nous avons supprimé l'augmentation de la variabilité, il faut élaborer le modèle en déterminant l'ordre de celui-ci et en calculant les coefficients.

2 Détermination du modèle

Afin de déterminer l'ordre du modèle qu'il faudra utiliser, il faut d'abord commencer par rendre la série stationnaire. Pour cela, il faut utiliser la série transformée en prenant le logarithme. Ensuite, puisque la tendance semblait être linéaire, il suffit de prendre la différence d'ordre 1 afin de supprimer la tendance. Pour vérifier si la tendance est bien éliminée, on peut à ce stade recalculer une régression linéaire et voir si la pente est toujours positive.

La pente a désormais bien l'air nulle, il n'y a donc plus de tendance et on peut maintenant supprimer les saisons. Pour éliminer celles-ci, il suffit de soustraire chaque valeur avec la valeur correspondante de la saison précédente. En d'autres termes, il faut prendre la différence de la série avec un décalage de 12. Une fois la série devenue stationnaire, on peut calculer ses séquences d'autocorrélations et d'autocorrélations partielles.

Les séquences permettent de déterminer approximativement les ordres maximums pour les différents paramètres. Ici, l'autocorrélation est significative jusqu'au lag 2, et celui-ci semble également significatif après une saison, mais ne l'est plus à partir de la seconde. On pourrait donc penser que lorsqu'il faudra choisir un modèle, pour la composante stationnaire en MA, q

ne sera pas plus grand que 2 (ou 3, qui sans être significatif, est proche de la limite), et que pour la composante saisonnière, Q ne sera pas plus grand que 1. Concernant l'autocorrélation partielle, celle-ci est significative seulement jusqu'au lag 1 et n'est déjà plus significative après une période. On peut donc supposer pour la composante AR que p ne sera pas plus grand que 1 et que P sera nul.

On peut donc utiliser la fonction « Comp.Sarima », afin de déterminer l'ordre idéal du modèle. Pour obtenir le meilleur modèle, on utilise les paramètres p.max=1, q.max=3, P.max=1 et Q.max=1. On obtient les résultats suivants :

```
0 1 3 x 0 1 1 saison : 12 : nb param : 4 AIC : 0
0 1 3 x 1 1 1 saison : 12 : nb param : 5 AIC : 1.014334
1 1 3 x 0 1 1 saison : 12 : nb param : 5 AIC : 1.285529
1 1 3 x 1 1 1 saison : 12 : nb param : 6 AIC : 2.589871
```

La fonction retourne les 10 % meilleurs modèles et l'AIC correspondant. Celui-ci est un critère d'information permettant de faire un compromis entre une vraisemblance élevée et un nombre de paramètres raisonnables. Plus l'AIC est petit, plus le compromis entre les deux est optimal. Néanmoins, comme l'on sait que l'AIC a parfois tendance à surévaluer le nombre de paramètres, on peut également réutiliser la même fonction, mais en fixant cette fois q.max à 2, ce qui semble cohérent avec les graphes de l'ACF/PACF.

```
1 1 0 x 0 1 1 saison : 12 : nb param : 2 AIC : 0
1 1 0 x 1 1 1 saison : 12 : nb param : 3 AIC : 0.6860883
1 1 1 x 0 1 1 saison : 12 : nb param : 3 AIC : 0.5652325
```

Ici, le meilleur modèle ne prend plus que 2 paramètres au lieu de 4. on comparera donc ces deux modèles, le premier ayant l'avantage de l'AIC, le second d'être plus parcimonieux.

On peut ensuite tenter de valider les modèles en effectuant un test sur les coefficients. Pour les 4 premiers modèles on obtient systématiquement au minimum un paramètre non significatif. Naturellement, il ne s'agit pas d'une bonne nouvelle et on peut essayer de s'intéresser à un modèle ayant un plus petit nombre de paramètres.

Pour le modèle 110X011 , on obtient : 0.9994 0.9916. Ce qui est nettement meilleur, et la fonction "OneAhead" indique que les prédictions d'un ou de l'autre modèle ne sont pas trop grandes, et que les deux sont très proches. On peut donc se focaliser sur ce modèle dont on peut observer les résidus.

Malheureusement, pour ce second modèle, les p-valeurs pour le test de Box-Pierce semblent proches de la limite, en particulier pour le lag 7 où elles semblent fort proches de 0,05.

Et par ailleurs, ceux-ci ne semblent pas normaux sur le qqplot. On a donc désormais des résidus qui ne semblent pas corrélés et dont la moyenne est nulle, mais qui ne sont pas obligatoirement IID.

Il semble donc malgré tout préférable de privilégier ce modèle. Comme le test de Box-Pierce donne un résultat limite, et que par ailleurs aucune structure de corrélation claire n'apparaît dans les graphes de l'autocorrélation et autocorrélation partielle, il peut sembler intéressant d'observer si les résidus au carré présentent une structure de corrélation similaire à celle d'un arima afin de pouvoir utiliser un modèle GARCH pour les modéliser. Cette intuition est renforcée par l'impression que donnent les résidus d'avoir une variance non constante.

Aucune structure de corrélation claire n'est visible, on peut donc considérer que les résidus sont bien un bruit blanc puisqu'ils ne semblent pas corrélés et que leur moyenne est nulle. Par ailleurs, la fonction `Comp.Sarima` calculée sur les résidus au carré renvoie bien $(000)x(000)$ comme modèle ayant le plus faible AIC, ce qui confirme bien le sentiment que les résidus ne sont plus qu'un bruit blanc.

3 Prédictions

On peut donc passer aux prédictions pour le modèle sélectionné sur une longueur de 10% de la série d'origine, avec l'intervalle de confiance à 0.95, dans le cas où les résidus sont normaux, ce qui peut être discutable étant donnée la forme un peu particulière du qqplot.

Sur ce graphique se trouve à titre purement indicatif l'intervalle de confiance pour le modèle ayant le plus petit AIC. L'intervalle du modèle ayant le plus petit AIC (mais pour lequel le test sur les coefficients posait problème) est en vert, et celui du modèle sélectionné est en rouge. On voit que comparativement, l'intervalle de confiance du second est légèrement plus large.

On peut comparer ces prédictions avec celles obtenues par lissage exponentiel.

Si on voit d'emblée que les deux prédictions sont légèrement différentes, il peut être intéressant d'intégrer les deux prédictions sur le même graphique.

On voit que les prédictions du modèle de Holt-Winters, en rouge, sont légèrement plus grande que celles du modèle SARIMA. Par ailleurs, l'intervalle de confiance du modèle de Holt-Winters semble légèrement plus élevé que pour le modèle SARIMA.

Néanmoins, cette conclusion paraît peu satisfaisante : les résidus ne semblent pas normaux, et leur variance a l'air bien peu constante (en particulier, on dirait qu'ils sont presque nuls entre 1965 et 1966). La série différenciée et transformée semble pourtant bien stationnaire, et il faudrait donc trouver en apparence un autre modèle. Néanmoins aucun de ceux proposés par la fonction `Comp.Sarima` ne possède des coefficients significatifs et des résidus normaux, et leurs résidus ne semblent pas pouvoir être modélisés en utilisant un modèle GARCH. Utiliser `BoxCox` au lieu du logarithme pour stabiliser la série ne résout aucun des problèmes. D'autres modèles, comme $(1,1,3)x(1,0,0)$ ne possèdent aucun problème de normalité ou de coefficients, mais ne semblent pas pouvoir prédire correctement la série comme le montrent les deux graphiques ci-dessous.

En particulier dans la prédiction avec ce modèle, il ne semble pas y avoir d'augmentation de l'amplitude des saisons, contrairement à ce qui se trouve dans la série d'origine.

4 Annexes

```
#### Chargement des donnees et fonctions
source("/run/media/alexis/6917a5e2-e4da-439f-ada9-c2d93a4db183/alexis
/stats/serieC/tp-3/FonctionsSeriesChrono.r")
source("/run/media/alexis/6917a5e2-e4da-439f-ada9-c2d93a4db183/alexis
/stats/serieC/tp-3/tsdiag2.txt")
setwd("/run/media/alexis/6917a5e2-e4da-439f-ada9-c2d93a4db183/alexis/
stats/serieC/")
library("fBasics")
library("forecast")
d<-ts(read.table("xsales.txt"), frequency = 12, start = 1965)

#### Premiere presentation des donnees
par(mfrow=c(3,1))
plot(d, ylab="sales", main="Monthly_sales_of_company_an_american_
company_X._Jan_1965_-_May_1971.")
acf(d, main="ACF", ylab="acf")
pacf(d, main="PACF", ylab="pacf")

#observation d'une tendance
par(mfrow=c(1,1))
plot(d, ylab="sales", main="Monthly_sales_of_company_an_american_
company_X._Jan_1965_-_May_1971.")
t=seq(1965+1/12,by=1/12,length=length(d))
mod=lm(d~t)
abline(reg=mod,col="red")

#observation d'une saisonalite avec graphique
plot(d[1:12],type="l", ylim=c(min(d),max(d)))
for(i in 2:12){
  lines(d[12*(i-1)+(1:12)],col=i)
  legend("topleft",legend =1965:1971, fill= 1:12)
}

#observation d'une saisonalite avec un tableau
# Calcul des moyennes annuelles
moyenne=NULL
for (i in 1:6){
  moyenne=c(moyenne,mean(d[6*(i-1)+(1:12)]))
}
moyenne
```

```

# Calculs de Signe
sais=rep(NA,length(d))
sais=ts(sais, start = 1965, frequency= 12)
for (y in 1:6){
  for (i in 1:12){
    sais[12*(y-1)+i]=ifelse(d[12*(y-1)+i]>moyenne[y], "+", "-")
  }
}
sais

#observation de la variabilite
plot(diff(d), xlab="temps (mois)", ylab="(1-B)Xt")

x=matrix(d[1:72], ncol=12, byrow=TRUE)
y=sapply(1:6, function(i)
  diff(range(x[i,])))
plot(1965:1970, y, type="b", xlab="Annee", ylab="Amplitude", main = "
  Variation_de_l' amplitude_au_cours_du_temps")

x=matrix(log(d[1:72]), ncol=12, byrow=TRUE)
y=sapply(1:6, function(i)
  diff(range(x[i,])))
plot(1965:1970, y, type="b", xlab="Annee", ylab="Amplitude", main = "
  Variation_de_l' amplitude_au_cours_du_temps")

x=matrix(BoxCox(d[1:72], BoxCox.lambda(d)), ncol=12, byrow=TRUE)
y=sapply(1:6, function(i)
  diff(range(x[i,])))
plot(1965:1970, y, type="b", xlab="Annee", ylab="Amplitude", main = "
  Variation_de_l' amplitude_au_cours_du_temps")

#decomposition
plot(decompose(d, "multiplicative"))

###Determination de l'ordre
#stabilisation de la serie en utilisant le log
d.s<-log(d)
plot(d.s)
d.sta<-diff(d.s, lag=1)

par(mfrow=c(1,1))
plot(d.sta, ylab="sales", main="Monthly_sales_of_company_an_american_
  company_X._Jan_1965_-_May_1971.")
t=seq(1965+1/12, by=1/12, length=length(d.sta))
mod=lm(d.sta~t)
abline(reg=mod, col="red")

d.sta<-diff(d.sta, lag=12)

```

```

par(mfrow=c(3,1))
plot(d.sta)
acf(d.sta, lag.max = 25)
pacf(d.sta, lag.max = 25)
#p.max=1, q.max=3, P.max=1, Q.max=1

#stabilisation de la serie en utilisant BoxCox
lambda<-BoxCox.lambda(d)
dc.s<-BoxCox(d, BoxCox.lambda(d))
plot(dc.s)
dc.sta<-diff(dc.s, lag=1)

par(mfrow=c(1,1))
plot(dc.sta, ylab="sales", main="Monthly_sales_of_company_an_american
      _company_X._Jan_1965_-_May_1971.")
t=seq(1965+1/12,by=1/12,length=length(dc.sta))
mod=lm(dc.sta~t)
abline(reg=mod,col="red")

dc.sta<-diff(dc.sta, lag=12)
par(mfrow=c(3,1))
plot(dc.sta)
acf(dc.sta, lag.max = 25)
pacf(dc.sta, lag.max = 25)
#comparaison de modeles
#Le suivant fait la comparaison la plus probables
Comp.Sarima(d.s, d=1, saison = 12, D=1, p.max = 1, q.max = 2, P.max =
      1, Q.max = 1)
Comp.Sarima(dc.s, d=1, saison = 12, D=1, p.max = 1, q.max = 2, P.max
      = 1, Q.max = 1)
Comp.Sarima(d.s, d=1, saison = 12, D=1, p.max = 1, q.max = 3, P.max =
      1, Q.max = 1)

###Estimation du modele
mod = arima(d.s, order = c( 0,1, 3), seasonal=list(order= c(0 ,1,1 )
      , period =12))
mod2 = arima(d.s, order = c( 1,1, 0), seasonal=list(order= c(0 ,1,1 )
      , period =12))
mod3 = arima(d.s, order = c( 0,1, 3), seasonal=list(order= c(1 ,1,1 )
      , period =12))
mod4 = arima(d.s, order = c( 1,1, 3), seasonal=list(order= c(0 ,1,1 )
      , period =12))
mod5 = arima(d.s, order = c( 1,1, 3), seasonal=list(order= c(1 ,1,1 )
      , period =12))
mod6 = arima(d.s, order = c( 1,1, 0), seasonal=list(order= c(1 ,1,1 )
      , period =12))
mod7 = arima(d.s, order = c( 1,1, 1), seasonal=list(order= c(0 ,1,1 )
      , period =12))
modc = Arima(dc.s, order = c( 1,1,0), seasonal=list(order= c(0 ,1,1 )

```



```
, period =12), lambda=lambda)
moda = arima(d.s, order = c( 1,1, 3), seasonal=list(order= c(1 ,0,0 )
, period =12))
```

###Validation des modeles

```
tsdiag2(mod,gof.lag=floor(sqrt(length(d.s))))
tsdiag2(mod2,gof.lag=floor(sqrt(length(d.s))))
tsdiag2(mod3,gof.lag=floor(sqrt(length(d.s))))
tsdiag2(mod4,gof.lag=floor(sqrt(length(d.s))))
tsdiag2(mod5,gof.lag=floor(sqrt(length(d.s))))
tsdiag2(mod6,gof.lag=floor(sqrt(length(d.s))))
tsdiag2(mod7,gof.lag=floor(sqrt(length(d.s))))
tsdiag2(modc,gof.lag=floor(sqrt(length(dc.s))))
tsdiag2(moda,gof.lag=floor(sqrt(length(d.s))))
```

```
round(coef.p(mod$coef[1:4],diag(mod$var.coef)[1:4]),4)
round(coef.p(mod2$coef[1:2],diag(mod2$var.coef)[1:2]),4)
mean(mod2$residuals)
round(coef.p(mod3$coef[1:5],diag(mod3$var.coef)[1:5]),4)
round(coef.p(mod4$coef[1:5],diag(mod4$var.coef)[1:5]),4)
round(coef.p(mod5$coef[1:6],diag(mod5$var.coef)[1:6]),4)
round(coef.p(mod6$coef[1:3],diag(mod5$var.coef)[1:3]),4)
round(coef.p(mod7$coef[1:3],diag(mod5$var.coef)[1:3]),4)
round(coef.p(modc$coef[1:2],diag(modc$var.coef)[1:2]),4)
round(coef.p(moda$coef[1:5],diag(moda$var.coef)[1:5]),4)
```

```
qqnorm(mod$residuals)
qqnorm(mod2$residuals)
qqnorm(mod3$residuals)
qqnorm(mod4$residuals)
qqnorm(mod5$residuals)
qqnorm(mod6$residuals)
qqnorm(mod7$residuals)
qqnorm(modc$residuals)
qqnorm(moda$residuals)
```

```
OneAhead(d.s, order = c( 0,1, 3), seasonal=list(order= c(0 ,1,1 ),
period =12))[[3]]
```

```
par(mfrow=c(1,1))
round(coef.p(mod2$coef[1:2],diag(mod2$var.coef)[1:2]),4)
tsdiag2(mod2,gof.lag=floor(sqrt(length(d.s))))
qqnorm(mod2$residuals)
dagoTest(mod2$residuals)
jarque.bera.test(mod2$residuals)
shapiro.test(mod2$residuals)
```

#Difference dans la prediction pour les differents modeles

```
OneAhead(d.s, order = c( 1,1, 0), seasonal=list(order= c(0 ,1,1 ),
```

```

    period =12)) [[3]]
OneAhead(d.s,   order = c( 0,1, 3), seasonal=list(order= c(0 ,1,1 ),
    period =12)) [[3]]
OneAhead(dc.s,   order = c( 1,1, 0), seasonal=list(order= c(0 ,1,1 ),
    period =12)) [[3]]
OneAhead(d.s,   order = c( 1,1, 3), seasonal=list(order= c(1 ,0,0 ),
    period =12)) [[3]]

```

#Etude des residus au carre

```

rescar<-mod2$residuals*mod2$residuals
par(mfrow=c(3,1))
plot(rescar)
acf(rescar)
pacf(rescar)
Comp.Sarima(rescar , d=0, saison = 12, D=0, p.max = 1, q.max =1 , P.
    max = 1, Q.max = 1)

```

###Prediction

```

par(mfrow=c(1,1))
pred.ts =predict(mod2, n.ahead = 0.1*length(d))
pred.ts2 =predict(mod, n.ahead = 0.1*length(d))
ts.tot = ts(c(d, exp(pred.ts$pred)), start= 1965, frequency=12)
plot ( ts.tot, type = "l",lty = 1, col ="blue", ylim=c(0,3000))
lines (exp(pred.ts$pred + 1.96*pred.ts$se), lty = 2, col = "blue")
lines (exp(pred.ts$pred - 1.96*pred.ts$se), lty = 2, col = "blue")
lines (exp(pred.ts2$pred + 1.96*pred.ts2$se), lty = 2, col = "green")
lines (exp(pred.ts2$pred - 1.96*pred.ts2$se), lty = 2, col = "green")
lines(exp(pred.ts2$pred), col="green")

```

#Holt-Winters

```

mod.HW.stab=HoltWinters(d.s)
pred.HW.stab=predict(mod.HW.stab ,n.ahead=0.1*length(d),prediction .
    interval=TRUE)
pred.HW=exp(pred.HW.stab)
ts.tot = ts(c(d,pred.HW[, "fit"]), start= 1965, frequency= 12)
plot ( ts.tot, type = "l",lty = 1, col ="blue",ylim=c(min(d,pred.HW),
    max(d,pred.HW)))
lines (pred.HW[, "upr"], lty = 2, col = "blue")
lines (pred.HW[, "lwr"], lty = 2, col = "blue")

```

#comparaison

```

par(mfrow=c(1,1))
pred.ts =predict(mod2, n.ahead = 0.1*length(d))
ts.tot = ts(c(d, exp(pred.ts$pred)), start= 1965, frequency=12)
plot ( ts.tot, type = "l",lty = 1, col ="blue", ylim=c(0,2500))
lines (exp(pred.ts$pred + 1.96*pred.ts$se), lty = 2, col = "blue")
lines (exp(pred.ts$pred - 1.96*pred.ts$se), lty = 2, col = "blue")
mod.HW.stab=HoltWinters(d.s)

```

```

pred.HW.stab=predict(mod.HW.stab,n.ahead=0.1*length(d),prediction.
  interval=TRUE)
pred.HW=exp(pred.HW.stab)
ts.tot = ts(c(d,pred.HW[, "fit"]), start= 1965, frequency= 12)
lines ( ts.tot, type = "l",lty = 1, col ="red",ylim=c(min(d,pred.HW),
  max(d,pred.HW)))
lines (pred.HW[, "upr"], lty = 2, col = "red")
lines (pred.HW[, "lwr"], lty = 2, col = "red")

#Comparaison entre moda, mod2, et HW
par(mfrow=c(1,1))
pred.ts =predict(mod2, n.ahead = 0.1*length(d))
ts.tot = ts(c(d, exp(pred.ts$pred)), start= 1965, frequency=12)
plot ( ts.tot, type = "l",lty = 1, col ="blue", ylim=c(0,2500), main
  = "Comparaison_entre_predictions_des_modeles_110x011_(Bleu),_113
  x110_(Vert)_et_HW_(Rouge)")
pred.HW.stab=predict(mod.HW.stab,n.ahead=0.1*length(d),prediction.
  interval=TRUE)
pred.HW=exp(pred.HW.stab)
ts.tot = ts(c(d,pred.HW[, "fit"]), start= 1965, frequency= 12)
lines ( ts.tot, type = "l",lty = 1, col ="red",ylim=c(min(d,pred.HW),
  max(d,pred.HW)))
par(mfrow=c(1,1))
pred2.ts =predict(moda, n.ahead = 0.1*length(d))
ts.tot2 = ts(c(d, exp(pred2.ts$pred)), start= 1965, frequency=12)
lines(ts.tot2, col="green")

#Entre moda, mod, mod2
plot(exp(predict(moda, n.ahead = 10)$pred ), ylim=c(300, 1200), col=
  'blue', main="comparaison_entre_113x110_(Bleu),_et_110x011_et_013
  x011_(noirs)")
lines(exp(predict(mod2, n.ahead = 10)$pred))
lines(exp(predict(mod, n.ahead = 10)$pred))

```