

Aide à l'utilisation du logiciel R pour l'étude des séries chronologiques

O. Roustant

« R » est un logiciel de statistique que l'on peut télécharger et utiliser gratuitement. Réalisé par des professionnels en statistique, il est révisé et complété très régulièrement. Voir :

<http://cran.us.r-project.org/>

Concernant les séries chronologiques, on peut utiliser deux « packages » (bibliothèques) :

- le package « stats », qui permet d'ajuster un modèle de type ARIMA, et de calculer les prévisions et les intervalles de prévision dans le cas gaussien ; on peut aussi réaliser des prévisions par les méthodes de lissage exponentiel (Holt-Winters).
- le package « tseries », qui permet d'ajuster des modèles de type ARIMA et GARCH¹. L'analyse de variance est bien plus détaillée. En revanche, il ne permet pas d'obtenir les intervalles de prévision.

Les deux bibliothèques permettent d'utiliser de façon complémentaire des procédures permettant l'analyse et/ou la prévision des séries chronologiques.

¹ Ce package doit être installé car il ne figure pas sur l'installation du noyau de R. On doit aussi installer le package « quadprog ». Ces installations s'effectuent très simplement par téléchargement des fichiers zip, puis en utilisant dans R le menu « package ».

Chargement de la bibliothèque « tseries »

Dans le menu « Packages », choisir la fonction « Load package », et sélectionner dans la liste « tseries ». Autre possibilité : entrer en ligne : > library(tseries)

Aide

Il est plus convivial d'utiliser l'aide sous format html, disponible avec la fonction « Html help », du menu « Help ». L'aide sur une fonction s'obtient également en ligne, avec la fonction help(nom de la fonction)

Importation d'une série chronologique

1. Enregistrer le fichier de données sous le répertoire de travail (par défaut : C:\Program Files\R\rw2000). Celui-ci peut être changé avec la fonction « Change dir » du menu « File ».
2. Il existe une fonction spécifique pour importer une série chronologique, qui permet d'obtenir un objet de type « Time Series ». L'avantage est de pouvoir gérer les dates et la saisonnalité. Par exemple, pour la série internationale du trafic aérien, constituée de données mensuelles de janvier 1949 à décembre 1960, on écrit :

```
air <- read.ts(file="airpass.dat", start=1949, frequency=12, skip=13)
```

avec :

```
start : date initiale  
frequency : période de la saisonnalité (ici 12 mois)  
skip : nombre de lignes qui figurent avant les données  
header : TRUE si la 1ère ligne contient le nom des colonnes  
sep : "." lorsque le séparateur decimal est un point, "," sinon
```

Méthodes de lissage exponentiel

1. Les **méthodes de lissage exponentiel** s'obtiennent avec la fonction : `HoltWinters`. Le tableau ci-dessous résume les fonctions à entrer pour chaque méthode :

| Nom de la méthode | Commande |
|----------------------------|---|
| SES | <code>xlisse <- HoltWinters(x, beta=0, gamma=0)</code> |
| Holt | <code>xlisse <- HoltWinters(x, gamma=0)</code> |
| Holt-Winters additif | <code>xlisse <- HoltWinters(x, seasonal= "add")</code> |
| Holt-Winters multiplicatif | <code>xlisse <- HoltWinters(x, seasonal= "mul")</code> |

Remarque : à la place de "add" et "mul", on peut entrer n'importe quel début des mots "additive" et "multiplicative" (par exemple "addit" et "multip")

2. **Affichage et visualisation des résultats :**

| Fonction | Effet |
|-----------------------------------|---|
| <code>xlisse</code> | Affichage des paramètres |
| <code>summary(xlisse)</code> | Description de l'objet obtenu par <code>HoltWinters</code> |
| <code>plot(xlisse)</code> | Représentation des valeurs observées + valeurs lissées |
| <code>plot(xlisse\$fitted)</code> | Représentation de l'estimation du niveau, de la pente et de la saisonnalité |

3. **Prévision.** Avec la fonction `predict`. Exemple, prévisions jusqu'à l'horizon h :

```
h <- 5 ; p <- predict(xlisse, n.ahead=h)
```

4. **Représentation des prévisions.** Voici une solution :

- D'abord créer une série chronologique obtenue en rallongeant la série de données des prévisions. Ceci se fait grâce à la fonction `ts`
- Représenter la série augmentée ainsi obtenue
- Représenter sur ce graphique les prévisions avec une autre couleur (fonction `lines`)

Cela donne (sur l'exemple du trafic aérien international) :

```
xlisse <- HoltWinters(air, seasonal="mul")
p <- predict(xlisse, n.ahead=12)
x_p <- ts(c(air,p), start=1949, frequency=12)
plot(x_p, type="o")
lines(p, col="red")
```

5. **Intervalles de prévision** : ne pas les utiliser sans étude probabiliste préalable !

Moyenne mobile, simulation

1. Le lissage par moyenne mobile peut s'effectuer avec la fonction : filter

| Objectif | Commande |
|-------------------|--|
| Lissage MA centré | <code>xlisssse <- filter(x, poids, sides=2, method="conv")</code> |
| Lissage MA passé | <code>xlisssse <- filter(x, poids, sides=1, method="conv")</code> |

Dans ce tableau, poids désigne un vecteur de poids, par exemple :
`poids <- rep(1,5)/5`

2. La fonction `filter` permet aussi de simuler des trajectoires de processus AR et MA, mais on préfèrera en général la fonction plus générale `arima.sim`

Méthodologie de Box et Jenkins

1. Pour **différencier** une série, utiliser simplement la fonction `diff`
2. **L'ACF et la PACF** s'obtiennent simplement avec les fonctions `acf`, `pacf`
3. Pour l'**estimation**, le mieux est d'utiliser la fonction `arma` :

```
fit <- arma(y, order = c(p,q))
```

La table d'analyse de variance s'obtient alors avec `summary(fit)`, les graphiques de validation avec `plot(fit)`, et les résidus avec : `r <- residuals(fit)`

4. Pour le contrôle graphique de la **normalité des résidus** : `qqnorm` et `qqline`.
Pour un test de normalité : `ks.test` (test de Kolmogorov-Smirnov)
5. **Prévision dans le cas gaussien** : utiliser la fonction `arima`.

```
fitarima <- arima(x, order=c(p,d,q))
tsdiag(fitarima)
predict(fitarima, n.ahead = h)
```

Remarque : `tsdiag` fournit des graphiques de validation complémentaire.