

TD 1 - Séries temporelles

Partie I : manipulation de séries temporelles sous le logiciel R

help.start()

Ouvrir la fenêtre d'aide

Fonctions de base sur la série temporelle

Les données concernent le trafic voyageur de la SNCF en classe (première et deuxième). Elles sont exprimées en millions de kilomètres voyageurs. Les observations portent sur la période 1963-1980 (source : "Séries temporelles et modèles dynamiques", Gouriéroux et Monfort, 1983).

Question 1 : ouvrir le logiciel RStudio et copier le fichier des données 'trafic.dat' dans votre répertoire de travail. Changer le répertoire courant à ce répertoire.

Question 2 : taper les commandes suivantes et observer ce qu'elles produisent.

data<-scan("trafic.dat")	Chargement du jeu de données
str(data)	Donner la nature de l'objet data et ses dimensions
is.ts(data)	Vérifier si data est un objet R de type time series
data<-as.ts(data)	Transformer data en un objet du type time series
is.ts(data)	Vérifier
data	Affichage de data
data<-ts(data,start=1963,frequency=12)	Renouveler la série des temps i.e. assigner un début et une fréquence
data	Vérifier
tsp(data)	Obtenir le début, la fin et la fréquence de la série
frequency(data)	Fréquence de la série des temps
?par	
par(mfrow = c(1,3))	
ts.plot(data)	Représentation graphique de la série
matrix(data,12,10)	Représentations annuelles de la série
matrix(data,10,12)	Comparer avec la sortie précédente
ts.plot(matrix(data,12,10))	Graphes
ts.plot(matrix(data,10,12))	Comparer avec la sortie précédente
Xdata=window(data,c(1970,4),1980)	Extraire une sous-série temporelle

Simulation d'un bruit blanc gaussien (définition page 8 du polycopié)

<code>eps<-rnorm(100)</code>	Simuler un bruit blanc (centré) gaussien
<code>??</code>	Déterminer la nature et la longueur de l'objet <i>eps</i>
<code>y<-sort(eps)</code>	Ordonner le vecteur <i>eps</i> par valeurs croissantes
<code>dev.off()</code>	
<code>par(mfrow = c(1,2))</code>	
<code>hist(eps,freq=FALSE)</code>	Tracer l'histogramme
<code>lines(y,dnorm(y), col="red")</code>	Comparer avec la densité d'une loi $N(0;1)$
<code>qqnorm(eps)</code>	Tracer <i>eps</i> en fonction des quantiles d'une loi $N(0;1)$
<code>qqline(eps,col="red")</code>	Tracer la droite de régression (linéaire des moindres carrés) sur les données
<code>abline(0,1)</code>	Tracer la première bissectrice et commenter.
<code>help(abline)</code>	Pour avoir un descriptif de la fonction <i>abline()</i>
<code>? abline</code>	Même chose
<code>dev.off()</code>	

Question 3 :

- A. Comment générer un vecteur de composantes qui suivent une loi normale de moyenne m et d'écart-type $\sigma=2$
 - à partir du vecteur *eps* ?
 - avec la fonction *rnorm* ? Faire **help(norm)** pour vous aider.
- B. Recommencer cette simulation mais avec un bruit blanc de longueur 1 000 puis 10 000.
- C. Vérifier que l'histogramme se rapproche de celui d'une loi normale.
- D. Vérifier que la droite de régression des moindres carrés ordinaires se rapproche de la première bissectrice.
- E. Pourquoi ?

Simulation d'une série temporelle (modèle déterministe du cours)

Question 4 : simuler une série temporelle Y mensuelle commençant au mois de mars 1980 et finissant au mois de décembre 2000 et telle que :

$$Y_t = Z_t + S_t + \varepsilon_t$$

où :

- la composante tendancielle Z_t est supposée linéaire : $Z_t = 0,2t + 2$;
- la composante saisonnière S_t est une fonction périodique de période 12 :
$$S_t = \cos(2\pi(t - 1980));$$
- l'erreur ε_t est un bruit blanc gaussien d'écart-type $\sigma=2$.

Question 5 : utiliser les fonctions rencontrées précédemment pour transformer Y en objet time series. Renouveler la série des temps.

Question 6 : représenter la série. Superposer les séries annuelles des années 1981 à 1998 et extraire une sous série commençant en janvier 1985 et finissant en juillet 1991.

Partie II : décomposition de séries temporelles

Elimination de la composante saisonnière

```
data<-scan("trafic.dat")
filt<-rep(1/12,11)           Définition des coefficients
filt<-(c(1/24,filt,1/24))    Construction de la moyenne mobile
length(filt)
Z<-filter(data,filter=filt,sides=2) Calcul de la série filtrée par moyenne
                                arithmétique d'ordre 13 modifiée aux
                                extrémités
Z<-ts(Z,start=1963,frequency=12) Renouveler la série des temps
ts.plot(data,Z)              Superposition de la série brute et de la série
                                filtrée
```

Estimation de la composante saisonnière

```
S<-data-Z                    S la saisonnalité
s<-tapply(S,cycle(S),mean,na.rm=T)
                                Estimation c des coefficients du saisonnier
s<-s-mean(s)                 Série corrigée des variations saisonnières
CVS<-matrix(1,18,12)
for (i in 1 :18) {for (j in 1 :12) {CVS[i,j]=t(matrix(X,12,18))[i,j]-s[j]}}
                                Calcul de XCV S = X - S
CVS=as.vector(t(CVS))
CVS=as.ts(CVS)
CVS=ts(CVS,start=1963,frequency=12)
ts.plot(data)
ts.plot(CVS)
```

Estimation de la tendance

```
y=time(CVS)
z=time(CVS)^2
CVS.lm=lm(CVS~ y+z)          Estimation de la tendance :  $f(t) = a + bt + ct^2$  .
CVS.lm$coefficients          Valeurs des coefficients a, b et c
ts.plot(CVS)
ts.plot(time(CVS),CVS.lm$fitted.values)
```

Prévision

```
X1=rep(1,12)  
for (i in 1 :12)  
{X1[i]=a+b*(1981+(i-1)/12)+c*(1981+(i-1)/12)^2 +s[i]}  
X2=c(as.vector(X),X1)  
X2=as.ts(X2)  
X2=ts(X2,start=1963,frequency=12)
```

Prévoir le trafic voyageur pour 1981
Juxtaposition de la série chronologique (jusqu'en 1980) et de la prévision pour 1981
Transformation en SC
Renouvellement des temps

Analyse des résidus

```
res=CVS-CVS.lm$fitted.values  
res=res/sqrt(var(res))  
acf(res)  
hist(res)  
qqnorm(res)  
abline(0,1)
```

Définition des résidus
Définition des résidus réduits
Corrélogramme des résidus
Histogramme des résidus
Comparaison des quantiles des résidus et des quantiles d'une loi normale

Utilisation de la fonction `decompose()` de R

La fonction `decompose()` de R permet directement de décomposer, comme son nom l'indique, une série temporelle selon le modèle additif (par défaut) ou le modèle multiplicatif :

```
data=as.ts(data)  
data=ts(data,start=1963,frequency=12)  
data.dcp= decompose(data,type="add")  
plot(X.dcp)
```

Transformer X en un objet du type time series
Renouveler la série des temps

L'option `type="add"` ou `"mult"` permet de spécifier si on souhaite utiliser un modèle additif ou multiplicatif. Sur la représentation graphique, quatre courbes sont représentées : de haut en bas, figurent la série initiale, la tendance, la composante saisonnière et la partie résiduelle. Ces quatre parties correspondent aux différentes composantes de l'objet ainsi créé. La décomposition repose sur l'application de moyennes mobiles dont on peut préciser le filtre éventuellement. Par défaut, une moyenne mobile symétrique est employée.

Exercice : à rendre pour le dimanche 7 mars 2021 à 23h30

1. Définir un modèle additif (voir le cours) avec :
 - une tendance quadratique (c'est-à-dire du type $at^2 + bt + c$)
 - une composante saisonnière fonction périodique de période 4
 - une erreur de type bruit blanc gaussienPour simuler une série temporelle trimestrielle débutant au deuxième trimestre 1986 et finissant au premier trimestre 2001.
Nous choisirons les différents paramètres du modèle : coefficients de la tendance (a, b et c), expression de la saisonnalité et écart-type du bruit blanc).
2. Afficher les différentes composantes. Tracer la tendance, la saisonnalité, le bruit ainsi que la série chronologique obtenue.
3. Superposer les séries annuelles des années 1987 à 2000.
4. Extraire une sous série commençant au deuxième trimestre 1989 et finissant au troisième trimestre 1991.

Exercice : à rendre pour le dimanche 7 mars 2021 à 23h30

Simuler une série chronologique $(Y_t)_{t=1, \dots, 100}$ suivant le modèle

$$Y_t = 0,01t + 1 + 2 \sin(2\pi t/5) + \varepsilon_t,$$

où $(\varepsilon_t)_t$ est un bruit blanc gaussien de variance 1/100.

1. Déterminer la tendance, la saisonnalité (période) de cette série chronologique. Les tracer.
2. Utiliser la méthode des moyennes mobiles ci-dessus pour éliminer la saisonnalité puis estimer les coefficients du saisonnier.
3. Utiliser une régression linéaire par moindres carrés pour estimer les coefficients de la tendance.
4. Comparer les estimateurs avec les vrais coefficients.
5. Proposer une prévision à l'horizon 3.
6. Analyser les résidus. Les représenter.
7. Appliquer la fonction `decompose` et comparer avec les vraies valeurs.