

TP1Bis: R pour les séries temporelles.

Exercice 1. (Fonction `ts()`, chronogramme). Une série temporelle est un vecteur où à chaque élément est associée une date. Un exemple est donnée par le taux de chômage (pourcentage de chômeurs sur l'ensemble des forces de travail) en France pour les années 2000 -2013 :

2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
8.6	7.8	7.9	8.6	8.9	8.9	8.8	8.0	7.4	9.1	9.3	9.2	9.8	10.

1. On veut afficher la courbe de chômage : stocker les taux de chômage dans un vecteur et le représenter graphiquement à l'aide de `plot()`.
2. Le vecteur considéré contient seulement la variable d'intérêt : nous avons perdu trace des dates. Comme première approche pour gérer ce problème, créer une matrice dont la première colonne contient les années allant de 2000 jusqu'à 2013 et la deuxième colonne contient les taux de chômage correspondants. Afficher cette matrice à l'aide de `plot()`.
3. R permet de gérer de façon efficace une structure de série temporelle à l'aide de la fonction `ts()`. Cette fonction requiert la spécification de trois paramètres importants :
 - `start` : la date de début de la série (la première unité de temps)
 - `frequency` : le nombre d'observations par unité de temps
 - `end` : la date de fin.

Dans notre exemple, nous avons une observation par unité de temps (c'est à dire par année) et donc `frequency=1`. Créer une série temporelle à partir du vecteur des taux de chômage précédemment créé. Afficher la série temporelle (le diagramme ainsi obtenu est appelé chronogramme).

Exercice 2. (Série mensuelle). On fabrique par simulation une série temporelle de février 2013 jusqu'en décembre 2014.

1. Simuler 23 observations indépendantes suivant une loi normale de moyenne 1 et variance 0.5 à l'aide de la fonction `rnorm()`.
2. Quelle est l'unité de temps ? Combien d'observations y a-t-il par unité de temps ? A partir des observations simulées créer et afficher une série temporelle allant de février 2013 jusqu'en décembre 2014.
3. La fonction `time()` donne le "temps" de chaque observation dans une série temporelle. Extraire le temps dans la série temporelle considérée et réfléchir à la façon dont celui-ci est mesuré.

Exercice 3 (Série quotidienne). On fabrique maintenant une série temporelle du 9 juillet 2006 au 31 décembre 2013 avec le nombre d'appels reçu par le service clients d'une banque.

1. La notation standard d'une date dans le format "caractère" est 'YYYY-MM-DD'. La fonction `strftime()` permet de convertir une date en ce format en d'autres format. Utiliser cette fonction avec le paramètre `format= '%j'` pour extraire le nombre du jour de l'année correspondant au 9 juillet

2006. Ce nombre est stocké comme un "character", utiliser `as.numeric()` pour le convertir en une valeur numérique.

2. Combien de jours y a-t-il dans la période considérée ? On peut calculer le nombre total de jours dans une période en calculant la longueur du vecteur de toutes les dates comprises entre le début et la fin de la période. Ce vecteur s'obtient avec

```
seq(from=as.Date('2006-07-09'),to=as.Date('2006-12-31'),by='day')
```

3. Simuler indépendamment une observation par jour suivant une loi de Poisson de paramètre $\lambda = 40$ à l'aide de la fonction `rpois()`.

4. Créer et afficher la série temporelle correspondante.

Exercice 4 (Données manquantes à intervalles régulières, sous-série). Dans certaines séries temporelles, des valeurs manquent régulièrement à certaines dates. Par exemple il n'y a pas de cotations pendant le week-end à la Bourse. Dans ces cas, il faut adopter le business time, une échelle de temps qui ne montre pas de manquants les samedis et dimanches. Dans cet exercice on travaille avec la série temporelle multiple `EuStockMarkets` fournie avec R (cours de 4 titres, stockés un par colonne).

1. Explorer la structure de `EuStockMarkets` à l'aide de la fonction `str(EuStockMarkets)`: Quelle est la période considérée ? Quelle est l'unité de temps ? Et le nombre d'observations dans l'unité de temps ? Comment interprétez-vous ce nombre ? Vous pouvez retrouver ces mêmes informations à l'aide des fonctions `start()`, `end()` et `frequency()`.

2. On considère le titre `DAX`, stocké dans la première colonne de la série multiple `EuStockMarkets`. Créer une variable `DAX` à partir de la première colonne de `EuStockMarkets` et vérifier qu'elle a la structure de série temporelle.

3. Extraire la sous série du titre `DAX` dont le début est le 130e jour ouvrable de 1991 et dont la fin est le 140e jour ouvrable de 1991 à l'aide de la fonction `window()`.

Modélisation

Exercice 5 (Modélisation série réelle). On considère la série temporelle `sunspot.year` avec le nombre de taches solaires observées de 1700 à 1988 ($n = 289$).

1. Charger la série $(x_t)_{t=1,\dots,n}$ à l'aide de `data()` et afficher sa trajectoire.

2. On préfère étudier la racine carré des données (cette transformation rends la variance plus constante). Calculer la nouvelle série $(z_t = \sqrt{x_t})_{t=1,\dots,n}$

3. Calculer les fonctions empiriques d'autocorrélation et d'autocorrélation partielle de $(z_t)_{t=1,\dots,n}$ et afficher les autocorrélogrammes empiriques correspondants.

4. Le quel entre les modèles `AR(1)`, `AR(2)`, `MA(1)`, `MA(2)` vous paraît-il plus approprié pour $(Z_t)_{t=1,\dots,n}$? Répondre en se basant sur les autocorrélogrammes empiriques affichés au point précédent.

5. Estimer les paramètres du modèle choisi à l'aide de la fonction `arma()`. Écrire le modèle estimé ($\hat{Z}_{t=1,\dots,n}$)
6. Tester la blancheur du processus résiduel $Z_t - \hat{Z}_t$.
7. En utilisant le modèle estimé, prédire le nombre de taches solaires pour les années 1989-1993.
8. Le nombre de taches solaires réellement observées dans les années 1989-1993 est donné dans le tableau suivant

1989	1990	1991	1992	1993
157.60	142.60	145.70	94.30	54.60

Afficher dans un même graphique toutes les données et les prévisions faites au point précédent.